SIAM J. Comput.
Vol. 32, No. 3, pp. 754–792

THE RECONSTRUCTION OF DOUBLED GENOMES*

NADIA EL-MABROUK[†] AND DAVID SANKOFF[‡]

Abstract. The genome can be modeled as a set of strings (chromosomes) of distinguished elements called genes. Genome duplication is an important source of new gene functions and novel physiological pathways. Originally (ancestrally), a duplicated genome contains two identical copies of each chromosome, but through the genomic rearrangement mutational processes of reciprocal translocation (prefix and/or suffix exchanges between chromosomes) and substring reversals, this simple doubled structure is disrupted. At the time of observation, each of the chromosomes resulting from the accumulation of rearrangements can be decomposed into a succession of conserved segments, such that each segment appears exactly twice in the genome. We present exact algorithms for reconstructing the ancestral doubled genome in linear time, minimizing the number of rearrangement mutations required to derive the observed order of genes along the present-day chromosomes. Somewhat different techniques are required for a translocations-only model, a translocations/reversals model, both of these in the multichromosomal context (eukaryotic nuclear genomes), and a reversals-only model for single chromosome prokaryotic and organellar genomes. We apply these methods to the yeast genome, which is thought to have doubled, and to the liverwort mitochondrial genome, whose duplicate genes are unlikely to have arisen by genome doubling.

Key words. genome duplication, genome rearrangement, signed genes, reversal, translocation, Hannenhalli–Pevzner graph, exact polynomial algorithms

AMS subject classifications. 68Q25, 68Q17, 05C38, 05C62, 05C85

PII. S0097539700377177

1. Introduction.

In almost all the genomes which have been studied, there are some genes that are present in two or more copies. These copies may be identical or may have some differences, and they may be adjacent on a single chromosome or dispersed on different chromosomes throughout the genome. There are a number of different ways in which duplicate genes can arise; perhaps the most spectacular mechanism is the simultaneous doubling of the entire genome. Normally a lethal accident of meiosis or other reproductive step, if genome doubling can be resolved in the organism and eventually fixed as a normalized diploid state in a population, simultaneous doubling constitutes a duplication of the entire genetic complement. It transcends other mechanisms for gene duplication in that not only is one copy of each gene free to evolve its own function (or to lose function, becoming a pseudogene and mutating randomly, eventually beyond recognition), but it can evolve in concert with any subset of the thousands of other extra gene copies (cf. [14] for accounts of gene family coevolution). Whole new physiological pathways may emerge, involving novel functions for many of these genes. Genome duplication is thus a likely source of rapid and far-reaching evolutionary progress. Its rarity does not detract from its

[*] Received by the editors August 29, 2000; accepted for publication (in revised form) December 13, 2002; published electronically April 23, 2003. This work was supported in part by grants from the Natural Sciences and Engineering Research Council of Canada. Various parts of this paper were presented at the RECOMB conference in Lyons in April 1999, the Genome Informatics Workshop (Tokyo) in December 1999, the JOBIM conference in Montpellier in May 2000, and at the DCAF conference in Montreal in September 2000 and appear in preliminary form in the proceedings of these conferences. The present version contains important corrections and improvements.

http://www.siam.org/journals/sicomp/32-3/37717.html

[†] Département d'Informatique et de Recherche Opérationnelle, Université de Montréal, CP 6128 Succursale Centre-ville, Montréal, QB H3C 3J7, Canada (mabrouk@iro.umontreal.ca).

[‡] Centre de Recherches Mathématiques, Université de Montréal, CP 6128 Succursale Centre-ville, Montréal, QB H3C 3J7, Canada (sankoff@crm.umontreal.ca).

importance.

For some genomes, recent polyploidy is easily detected due to the presence of a complete set of duplicated chromosomes. However, in most cases, all we can observe are duplicated chromosomal segments scattered throughout the genome.

Evidence for the effects of genome duplication has shown up across the eukaryote spectrum. More than two hundred million years ago, the vertebrate genome may have undergone two duplications [4, 20, 31], though at least one of these remains controversial [38, 21, 25, 8, 13]. Although numerous reversals and reciprocal translocations have subsequently occurred, the number of such chromosome rearrangements has been sufficiently modest that hundreds of conserved paralogous segments can be detected in the human genome since the ancient duplications; similar observations hold for the mouse genome [28, 29] and for less intensively mapped vertebrate genomes. More recent genome duplications are known to have occurred in some vertebrate lines, such as the frogs [40], the salmoniform fish [31], and the zebrafish [33].

Another example is given by the comparison of chromatin-eliminating Ascaridae with other nematodes. This comparison suggests that somatic cells of these worms have discarded a good proportion of the genes present in germ cells, possibly because these are redundant duplicates arising through genomic doubling some 200 million years ago [27].

Genome duplication is particularly prevalent in plants. Comparison of the well-studied rice [1], oats (wild and domestic), corn [1, 15], and wheat [26] genomes indicate several occurrences in the cereal lineage. Soybeans [36], Arabidopsis [24, 3], rapeseed [34], and other cultivars have genome duplications in their ancestry. Paterson et al. have presented convincing evidence that one or more genome duplications also occurred much earlier in plant evolution [32].

Following the complete sequencing of all Saccharomyces cerevisiae chromosomes, the prevalence of gene duplication has led to the hypothesis that this yeast genome is also the product of an ancient doubling [35, 39].

What of bacteria and other prokaryotes? In 1985, Herdman [19], observing that bacterial genome sizes clustered around multiples of 0.8Mb (i.e., 1.6Mb, 3.2Mb, etc.), suggested that the larger ones are the product of ancient duplications. The gene order of modern-day bacteria is not strong evidence for or against such duplication. There are often pairs of regions which are similar in gene content and order, but these are too rare and scattered to be convincing proof of a genome-wide duplication. If this event did occur, it has since been almost totally obscured by loss or divergence (in sequence and function) of one or both of the copies of most gene pairs, by lateral transfer of genes among related and unrelated organisms and by extensive rearrangement of the gene order. Nevertheless, prokaryotic genome duplication remains a possibility and often crops up in the literature, e.g., [23]. In contrast to plants, fungi, animals, and other eukaryotes which have a multiple-chromosome genome in their nucleii, prokaryotes tend to have a single, often circular, chromosome, so that translocation is not a possibility. They do not have meiosis, so genome duplication cannot arise as a result of a defect in this mechanism. It could, however, result from a fusion of two sister genomes. Reversal of long or short chromosomal segments is often cited as one of the predominant mechanisms for gene order rearrangement in unichromosomal genomes.

The prevalence and evolutionary importance of genome duplication, together with the fragmented nature of its present-day remnants, usually greatly obscured by subsequent developments at the sequence and chromosomal levels, lead to the question addressed in this paper: How can we reconstruct some or most of the original gene order at the time of genome duplication, based on traces conserved in the ordering of those duplicate genes still identifiable? Solving this would allow us key insights into the mechanisms and consequences of this dramatic evolutionary event. A similar question can also be considered in the case of duplication of fragments of chromosomes [9].

Originally a duplicated genome contains two identical copies of each chromosome, but through intrachromosomal movements and reciprocal translocations, this simple doubled structure is disrupted. The problem considered here is therefore as follows: given a present-day genome modeled by a set of strings (chromosomes) of distinguished elements (genes), each gene appearing exactly twice in the genome, how to recover an ancestral duplicated genome by performing a minimal number of reversals and/or reciprocal translocations? We assume that a sign + or − is associated to each gene, representing its transcriptional orientation. Our method makes use of a formula of Hannenhalli and Pevzner (HP) for the classical problem of signed genome rearrangement.

In a series of papers published in 1995, HP solved the problems of calculating the minimum number of rearrangements necessary to transform one signed genome G into another signed genome H, with rearrangement models based on

- reversals only [17],
- translocations only [16], and
- both reversals and translocations [18].

Though the minimizing formulae and their derivations are different in each case, the frameworks for the three models are similar. They are based on a graph called the *breakpoint graph*, in which each vertex is incident to one black and one gray edge, black edges corresponding to genome G, and gray edges to genome H. This graph decomposes naturally into a set of color-alternating cycles. The number of cycles is the dominant term in the minimizing formulae. The other terms depend on overlap relationships among these cycles, and on their clustering into "good" and "bad" components.

The 1995 papers also presented exact polynomial algorithms for actually constructing a series of rearrangements satisfying the minimality criterion. Subsequently, many alternate versions have been proposed to make various parts of the algorithms more efficient [22, 6, 5, 37]. However, our results on duplicated genomes do not depend on these algorithms. After deriving an ancestral genome by our new methods, an efficient version of the HP algorithm can simply be applied to the present-day genome to convert it to the ancestral genome we obtain.

Our approach in this paper is, given a present-day genome G, to estimate its ancestral polyploid genome by one whose comparison with G minimizes the HP formulae. As the ancestral genome H is unknown, we can start only with the *partial graph* of black edges, and we must complete this graph with an optimal set of gray edges. Though the three evolutionary models described above have different aspects related to the particular kind of genome (multichromosomal or circular) and operation (translocations and/or reversals) considered, the key concepts are the same for the three models.

The first step of the general method is to complete the graph with "valid" gray edges, i.e., gray edges representing a duplicated genome, so as to maximize the number of cycles of the resulting graph. The key idea is to subdivide the graph into a set of disjoint subgraphs, called *natural and supernatural graphs*, that can be solved independently. This is detailed in section 5. These graphs first provide an upper bound for the number of cycles. This bound is presented in section 6. Section 7 then describes a linear algorithm, called *dedouble*, for constructing a completed graph, where the number of cycles actually attains the upper bound. The main characteristic of *dedouble* is that any gray edge constructed links two vertices of the same supernatural graph. The second step of the general method consists in modifying *dedouble* in order to minimize the number of bad components. Though the concept of bad components is different for each of the three models, they are all related to the notion of *subpermutations* (SPs). Section 8 describes the general approach and the major modification to algorithm *dedouble*. Sections 9, 10, and 11 are then dedicated to the models with translocations only, translocations and reversals, and reversals only, respectively. Developments specific to each model are detailed in these sections. Finally, section 12 gives an application of our algorithm to the multichromosomal yeast genome, and section 13 gives another application to a circular mitochondrial genome.

We begin by formalizing the problem in the next section. We then introduce the HP graph and formulae in section 3 and introduce our notation and main definitions in section 4.

2. Formalizing the problem.

We consider three models: translocations-only, both translocations and reversals, and reversals-only. The first two pertain to the multichromosomal context (eukaryotic nuclear genomes), while the third is relevant to single chromosome prokaryotic and organellar genomes.

A *string* is a sequence of signed ($+$ or $-$) terms (*genes*) from a set $\mathcal{B}$. A *multichromosomal genome* is a collection of at least two nonnull strings (*chromosomes*). For a string $X = x_1 x_2 \cdots x_r$, denote by $-X$ the *reverse* string $-x_r\ -x_{r-1} \cdots -x_1$.

In the models with translocations, a *rearranged duplicated genome* G is a multichromosomal genome containing an even number of chromosomes, such that each gene in $\mathcal{B}$ is present exactly twice, i.e., once in each of two different chromosomes, or twice in a single chromosome.

Example 1. Let $\mathcal{B} = \{a, b, c, d, e, f, g, h\}$ be a set of 8 genes, and let G be a genome consisting of four chromosomes:

$$1\colon +a\ +b\ -c\ +b\ -d; \quad 2\colon -c\ -a\ +f;$$
$$3\colon -e\ +g\ -f\ -d; \quad 4\colon +h\ +e\ -g\ +h.$$

G is a rearranged duplicated genome. Each gene appears exactly twice in the set of chromosomes; e.g., gene b appears twice in chromosome 1. Signs represent gene orientation.

A circular chromosome is a string $x_1 x_2 \cdots x_r$, where x_1 is considered to follow x_r. As most single chromosome genomes contain a circular chromosome, in this paper only these circular genomes are considered. However, the application of all the results to genomes with single noncircular chromosomes is straightforward.

In the reversals-only model, a rearranged duplicated genome consists of a single circular genome G containing each gene in $\mathcal{B}$ exactly twice.

Example 2. Let $G = +a\ +b\ -c\ +b\ -d\ -e\ +a\ +c\ -d\ -e$. G is a rearranged duplicated genome on the set of genes $\mathcal{B} = \{a, b, c, d, e\}$. That G is a circular genome means that vertex $+a$ is considered to follow vertex $-e$.

The problem is to calculate the minimum number of rearrangement operations required to transform a given rearranged duplicated genome G into some *perfect duplicated genome* H (or simply *duplicated genome*) to be found. We call this problem the *genome halving problem.* In the case of a multichromosomal genome, H consists of chromosomes $C_1, \ldots, C_{2N}$, where for each $i \in \{1, \ldots, 2N\}$, we have $C_i = C_j$ for exactly one $j \in \{1, \ldots, 2N\} \backslash \{i\}$. In the case of a circular genome, H is of the form $C\ C$ or $C\ -C$, where C is a string containing exactly one occurrence of each gene of $\mathcal{B}$.

Each of the three models permits a different combination of the rearrangement operations reversal and translocation. A *reversal* transforms some proper substring of a genome into its reverse. Let X_1, X_2, Y_1, and Y_2 be nonnull strings. A *reciprocal translocation* between two chromosomes $X = X_1X_2$ and $Y = Y_1Y_2$ is of the form $X_1X_2, Y_1Y_2 \longrightarrow X_1Y_2, Y_1X_2$ (prefix-prefix) or of the form $X_1X_2, Y_1Y_2 \longrightarrow X_1 - Y_1, -Y_2X_2$ (prefix-suffix) (see Figure 2.1).

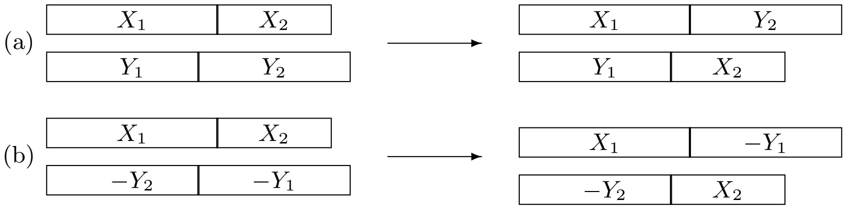

FIG. 2.1. *Reciprocal translocation between two chromosomes X_1X_2 and Y_1Y_2.* (a) *Prefix-prefix translocation.* (b) *Prefix-suffix translocation.*

3. The HP theory. Given two genomes H_1 and H_2 containing the same gene set $\mathcal{B}$, where each gene appears exactly once in each genome, the *genome rearrangement problem* is to find the minimum number of rearrangement operations necessary to transform H_1 into H_2 (or H_2 into H_1). HP designed polynomial algorithms for the reversals-only version of the problem (in the case of single chromosome genomes) [17], the translocations-only version [16], and the version with both reversals and translocations [18] (the latter two for multichromosomal genomes).

The algorithms all depend on a bicolored graph $\mathcal{G}_{12}$ constructed from H_1 and H_2. The details of this construction vary from model to model, due to the different ways chromosomal endpoints must be handled, but the general character of the graph is the same and may be summarized as follows.

Graph $\mathcal{G}_{12}$. If gene x of H_1 has positive sign, replace it by the pair x^tx^h, and if it is negative, replace it by x^hx^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 some chromosome in H_1, other than x^t and x^h deriving from the same x, are connected by a black edge (thick lines in figures), and any two adjacent in H_2 are connected by a gray edge (thin lines). In the case of a single chromosome, the black edges may be displayed linearly according to the order of the genes in the chromosome (Figure 3.1). For a genome containing N chromosomes, N such linear orders are required (Figure 3.2), and the genes at either end of the chromosome must be treated somewhat differently.

Now, each vertex is incident to exactly one black and one gray edge so that there is a unique decomposition of $\mathcal{G}_{12}$ into c_{12} disjoint cycles of alternating edge colors. By the *size of a cycle* we mean the number of black edges it contains. Note that $c_{21} = c_{12} = c$ is maximized when $H_1 = H_2$, in which case each cycle has one black edge and one gray edge.

A rearrangement operation ρ, either a reversal or a translocation, is determined by the two points where it "cuts" the current genome which correspond to two black edges e and f. We say that *ρ is determined by the two black edges e* and *f*. Rearrangement operations may change the number of cycles of the graph so that minimizing the number of operations can be seen in terms of increasing the number of cycles as fast as possible. Let $\Delta(c)$ be the difference between the number of cycles before and after applying the rearrangement operation ρ. HP showed that $\Delta(c)$ may take on values

1, 0, or −1, in which cases they are called *ρ proper*, *improper*, or *bad*, respectively. Roughly speaking, an operation determined by 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, all of whose black edges are related by the same linear order (i.e., are on the same line), and 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 crossing cycles, excluding the case of a cycle of size 1 (see Figures 3.1 and 3.2). A component is termed *good* if it can be transformed to a set of cycles of size 1 by a series of proper operations, and *bad* otherwise. Bad components are called *subpermutations* in the translocations-only model, *hurdles* in the reversals-only model, and *knots* in the combined model. More details on bad components and how to solve them will be given in the sections dedicated to each of the three evolutionary models.

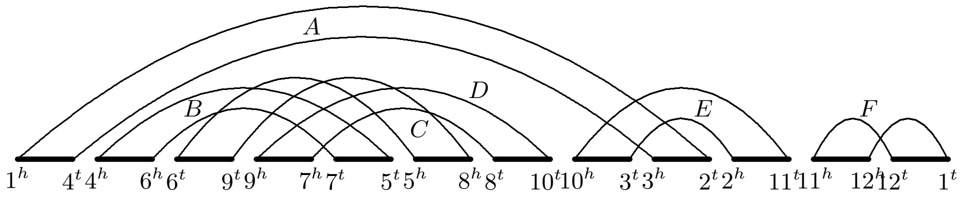

FIG. 3.1. *Graph $\mathcal{G}_{12}$ corresponding to circular genomes (i.e., the first gene is adjacent to the last gene) $H_1 = +1 +4 -6 +9 -7 +5 -8 +10 +3 +2 +11 -12$ (black edges) and $H_2 = +1 +2 +3 \cdots +12$ (gray edges). A, B, C, D, E, and F are the six cycles of $\mathcal{G}_{12}$. $\{A, E\}$, $\{B, C, D\}$, and $\{F\}$ are the three components of $\mathcal{G}_{12}$.*

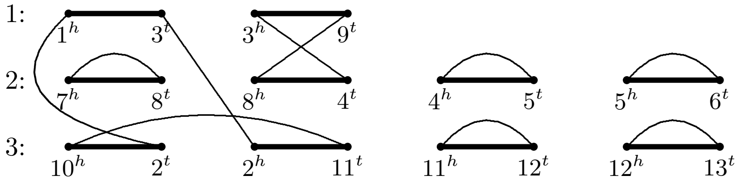

FIG. 3.2. *Graph $\mathcal{G}_{12}$ corresponding to genomes H_1, H_2, both with three chromosomes, where $H_1 = \{1 : 1\ 3\ 9\ ;\ 2 : 7\ 8\ 4\ 5\ 6\ ;\ 3 : 10\ 2\ 11\ 12\ 13\}$ and $H_2 = \{1 : 1\ 2\ 3\ 4\ 5\ 6\ ;\ 2 : 7\ 8\ 9\ ;\ 3 : 10\ 11\ 12\ 13\}$. All genes are signed "+." The edges, which are on the same horizontal row of the graph, correspond to a chromosome of H_1. There are seven cycles. As no cycle of size > 1 is contained on one row, $\mathcal{G}_{12}$ does not contain any component. Both genomes have the same set of endpoints, so we can omit the extremal vertices (x^t for initial genes and x^h for terminal genes) as discussed in section 4.*

The HP formulae for all three models may be summarized as follows:

$$\textbf{HP1}: \quad RO(H_1, H_2) = b(\mathcal{G}_{12}) - c(\mathcal{G}_{12}) + m(\mathcal{G}_{12}) + f(\mathcal{G}_{12}),$$

where $RO(G, H)$ is the minimum number of rearrangement operations (reversals and/or translocations), $b(\mathcal{G}_{12})$ is the number of black edges, $c(\mathcal{G}_{12})$ is the number of cycles, $m(\mathcal{G}_{12})$ is the number of bad components of $\mathcal{G}_{12}$, and $f(\mathcal{G}_{12})$ is a correction of size 0, 1, or 2 depending on the set of bad components.

Generally speaking, bad components are rare, so the number of cycles of $\mathcal{G}_{12}$ is the dominant parameter in the **HP1** formula if $b(\mathcal{G}_{12})$ is considered as a constant. In other words, the more cycles there are, the fewer reversals we need to transform H_1 into H_2.

4. Preliminaries. To make use of the HP graph structure for our genome halving problem, we first introduce, arbitrarily, a distinction within each pair of identical genes in the rearranged duplicated genome G, labeling one occurrence x_1 and the other x_2 for each x in $\mathcal{B}$.

In the case of linear chromosomes (noncircular), the HP method requires that the two genomes being compared share the same set of chromosomal endpoints. To ensure this constraint for linear multichromosomal genomes, we add a new initial term O_{i1} and a new final term O_{i2} to each chromosome C_i. This also ensures that all translocations, including those which reduce (by *fusion*, e.g., null X_1Y_2, Figure 2.1) or augment (by *fission*, e.g., null X_1X_2, Figure 2.1) the number of chromosomes in the genome, can be treated as reciprocal translocations. This also allows us to consider genomes with an odd number $2N-1$ of chromosomes by adding a dummy chromosome consisting of just one initial and one final O, to obtain $2N$ chromosomes.

In each chromosome, each x_j (except the O_{ij}) is replaced by x_j^t and x_j^h as in the HP construction. Define

$$O = \{O_{i1}, O_{i2}\}_{i=1,\cdots,2N}, \; V = \{x_j^s\}_{\substack{x\in\mathcal{B}\\ j=1,2}}^{s\in\{h,t\}}, \; \mathbf{V} = O \cup V.$$

In the case of a circular genome, endpoints are irrelevant, and thus the set O is empty, and $\mathbf{V} = V$. We use the notation $\overline{1} = 2$, $\overline{2} = 1$, $\widetilde{t} = h$, $\widetilde{h} = t$. For $u = x_j^s \in V$, its *counterpart*, denoted $\overline{u}$, is $x_{\overline{j}}^s$ (the corresponding vertex in the paralogous gene), and its *obverse*, denoted $\widetilde{u}$, is $x_j^{\widetilde{s}}$ (the vertex corresponding to the other "end" of the gene). Note that $\overline{\overline{u}} = \widetilde{\widetilde{u}} = u$.

The *partial graph* $\mathcal{G}(\mathbf{V}, A)$ associated with G has the edge set A of black edges linking adjacent terms (other than the obverse) in G. The partial graph associated with the genome G of Example 1 is shown in Figure 4.1. To differentiate the two occurrences of each gene x, one is subscripted "1," and its counterpart is "2."

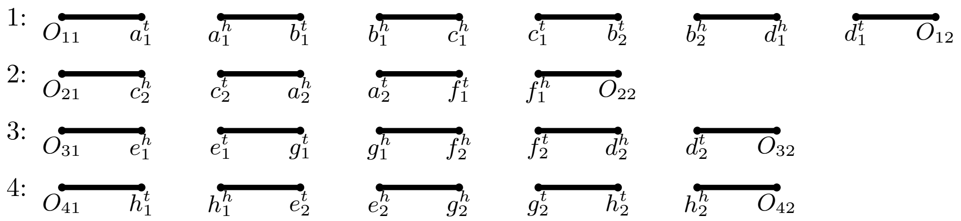
1:
O_{11} a_1^t a_1^h b_1^t b_1^h c_1^h c_1^t b_2^t b_2^h d_1^h d_1^t O_{12}
2:
O_{21} c_2^h c_2^t a_2^h a_2^t f_1^t f_1^h O_{22}
3:
O_{31} e_1^h e_1^t g_1^t g_1^h f_2^h f_2^t d_2^h d_2^t O_{32}
4:
O_{41} h_1^t h_1^h e_2^t e_2^h g_2^h g_2^t h_2^t h_2^h O_{42}

FIG. 4.1. *The partial graph $\mathcal{G}(\mathbf{V}, A)$ corresponding to Example* 1.

We are required to add to this partial graph a set Γ of gray edges so that every vertex in $\mathbf{V}$ is incident to exactly one black edge and one gray edge and so that the resulting genome is a perfectly duplicated one. A set Γ of gray edges giving rise to a duplicated genome is said to be *valid*. In the case of a multichromosomal genome, a chromosome of a perfectly duplicated genome should begin and end with two elements of O. The graph $\mathcal{G}_\Gamma(\mathbf{V}, A, \Gamma)$ obtained by adding a valid set Γ of gray edges is called a *completed graph* of $\mathcal{G}(\mathbf{V}, A)$. Lemmas 4.1 and 4.2 give the constraints that Γ should satisfy to be valid in the cases of multichromosomal and circular genomes, respectively.

LEMMA 4.1. *For multichromosomal genomes,* Γ *is valid if and only if the following conditions are satisfied:*

1. Γ *contains no edge of form* $(x, \overline{x})$ *for any* $x \in V$.
2. *Suppose* $(x, y) \in \Gamma$ *and* $y \in V$. *If* $x \in V$, *then* $(\overline{x}, \overline{y})$ *is also in* Γ. *Otherwise* $(x \in O)$, $\overline{y}$ *is also linked by a gray edge to an element of* O.
3. *The resulting genome does not contain any circular chromosome.*

Proof. Clearly, a duplicated genome must satisfy all three conditions. Suppose now that Γ is a set of gray edges so that every vertex of $\mathbf{V}$ is incident to exactly one gray edge, and Γ satisfies the three conditions. Then, from condition 3, as no circular fragment is present, and as the only "genes" with only one end are the elements of O, each chromosome of the resulting genome H has its two endpoints in O. From condition 1, the two copies of the same gene cannot be adjacent in H, and from condition 2, if two genes are adjacent in H, then their homologs are also adjacent in H in the same order. This ensures that each permutation (string) is present exactly twice in H. Therefore, H is a perfectly duplicated genome. □

LEMMA 4.2. *For circular genomes,* Γ *is valid if and only if the following conditions are satisfied:*

1. Γ *contains exactly zero or two edges of form* $(x, \overline{x})$.
2. *If* $(x, y) \in \Gamma$, *then* $(\overline{x}, \overline{y}) \in \Gamma$.
3. *The resulting genome consists of a single circular chromosome.*

Proof. The proof follows from the definition of a circular duplicated genome. □

To find a duplicated genome that gives rise to the minimal number of rearrangement operations, we have to construct a valid set of gray edges that minimizes the formula **HP1** (section 3). The key idea is to decompose the partial graph into a set of subgraphs that can be completed independently. We describe such a decomposition in the next section.

5. Decomposition into subgraphs. We define the set $\mathcal{NG}$ of *natural graphs* of $\mathcal{G}(\mathbf{V}, A)$ as follows.

DEFINITION 5.1. *Let* $e = (x, y) \in A$. *Define* A_e *recursively by* $(x, y) \in A_e$, *and if* $(x, y) \in A_e$, *then both the edge of* A *adjacent to* $\overline{x}$ *and the edge of* A *adjacent to* $\overline{y}$ *are also in* A_e.

Let $\mathbf{V}_e$ *be the subset of* $\mathbf{V}$ *made up of vertices incident to the edges in* A_e. *Then* $\mathcal{G}_e(\mathbf{V}_e, A_e)$ *is the* natural graph *(of* size $|A_e|$*) of* $\mathcal{G}(\mathbf{V}, A)$ *generated by* e. *Note that if* $f \in A_e$, *then* $A_f = A_e$.

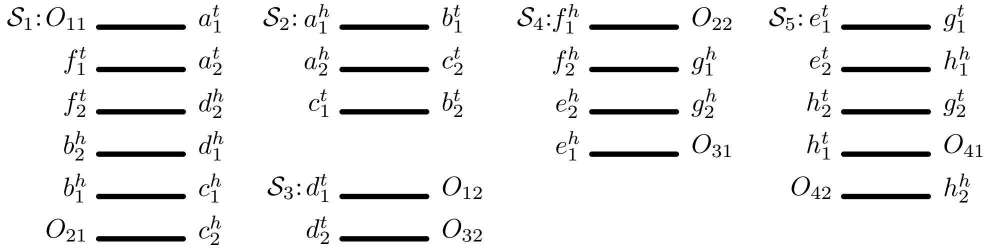

FIG. 5.1. *The natural graphs of the partial graph* $\mathcal{G}(\mathbf{V}, A)$ *of Figure* 4.1.

As an illustration, the decomposition of the partial graph of Figure 4.1 into natural graphs is given in Figure 5.1.

Let $\mathcal{G}_\alpha$ be a subgraph of $\mathcal{G}(\mathbf{V}, A)$. $\mathcal{G}_\alpha$ represents a set of fragments of the chromosomes of G. The subgraph $\mathcal{G}_\alpha$ is said to be *completable* if we can find a set of gray edges linking the vertices of $\mathcal{G}_\alpha$ that gives rise to a set of fragments of a potential duplicated genome. Not every natural subgraph is completable. In the case of multichromosomal genomes, we proved in [10] that a natural graph is completable if and only if it is of even size or it contains vertices in O. Similarly, for circular genomes, all natural graphs of even size are completable. Moreover, as we can have at most two gray edges of form $(u, \overline{u})$, then at most two natural graphs of odd size are completable.

The underlying idea of the subdivision and amalgamating procedure is to form completable graphs. First, $\mathcal{NG}$ is subdivided into the following subsets:

- $\mathcal{NE}$ is the subset of $\mathcal{NG}$ containing the natural graphs of even size.
- $\mathcal{NO}$ is the subset of $\mathcal{NG}$ containing the natural graphs of odd size. We further subdivide $\mathcal{NO}$ into $\mathcal{NO}_+$ and $\mathcal{NO}_-$ according to whether the natural graphs include vertices in O or not. Note that $\mathcal{NO}_+$ may contain a natural graph formed by a single edge linking two vertices in O.

The set A contains $2(|\mathcal{B}| + N)$ edges in the case of multichromosomal genomes, and $2|\mathcal{B}|$ edges in the case of circular genomes. Moreover, the graphs of $\mathcal{NE}$ contain an even number of edges. Therefore, $\mathcal{NO}$ must also contain an even number of edges and thus an even number of graphs. We can then pair off all the graphs in $\mathcal{NO}$ as follows:

- Arbitrarily choose pairs of graphs in $\mathcal{NO}_+$ to amalgamate. The set of larger graphs thus formed is denoted $\mathcal{SO}_+$.
- Arbitrarily choose pairs of the remaining graphs in $\mathcal{NO}$ to amalgamate. This includes graphs in $\mathcal{NO}_-$ plus, if applicable, the remaining one in $\mathcal{NO}_+$. The set of graphs thus formed is denoted $\mathcal{SO}$.

We denote $\mathcal{SE} = \mathcal{NE} \cup \mathcal{SO}_+$, and we call the graphs of $\mathcal{SN} = \mathcal{SE} \cup \mathcal{SO}$ *supernatural.*

In the example of Figure 5.1, $\mathcal{NE} = \{\mathcal{S}_1,\ \mathcal{S}_3,\ \mathcal{S}_4\}$, $\mathcal{NO}_- = \{\mathcal{S}_2\}$, and $\mathcal{NO}_+ = \{\mathcal{S}_5\}$. Moreover, $\mathcal{SE} = \mathcal{NE}$, and if $\mathcal{S}_{25}$ is the supernatural graph obtained by amalgamating $\mathcal{S}_2$ and $\mathcal{S}_5$, then $\mathcal{SO} = \{\mathcal{S}_{25}\}$. The set $\{\mathcal{S}_1, \mathcal{S}_{25}, \mathcal{S}_3, \mathcal{S}_4\}$ is a decomposition of $\mathcal{G}(\mathbf{V}, A)$ into supernatural graphs.

Note that for circular genomes, $\mathcal{NO}_+$ is empty, and thus $\mathcal{NO}$ graphs are arbitrarily amalgamated. In that case, $\mathcal{SO}_+$ is empty and $\mathcal{SE} = \mathcal{NE}$.

Notation 1. In a supernatural graph $\mathcal{G}_\alpha(\mathbf{V}_\alpha, A_\alpha)$ of $\mathcal{NE} \cup \mathcal{SO}$, if a vertex $u \in \mathbf{V}_\alpha \cap O$ exists, then we denote by $\overline{u}$ the (only) other vertex in $\mathbf{V}_\alpha \cap O$. For example, in Figure 5.1, $\overline{0_{11}} = 0_{21}$, and $\overline{0_{41}} = 0_{42}$.

In a supernatural graph $\mathcal{G}_\alpha(\mathbf{V}_\alpha, A_\alpha)$ of $\mathcal{SO}_+$ made up of two natural graphs $\mathcal{G}_1(\mathbf{V}_1, A_1)$ and $\mathcal{G}_2(\mathbf{V}_2, A_2)$ of $\mathcal{NO}_+$, if $u \in \mathbf{V}_1 \cap O$, then we arbitrarily choose one of the two vertices of $\mathbf{V}_2 \cap O$ to be $\overline{u}$.

5.1. Ordering the edges of the natural subgraphs. To simplify the ensuing development, we use a particular representation of each supernatural graph $\mathcal{G}_\alpha(\mathbf{V}_\alpha, A_\alpha)$ of size $2n$, where $n > 1$. Relabeling the vertices in $\mathbf{V}_\alpha$ allows us to define a *suitable order* for the edges in A_α (cf. Figure 5.2).

1. If $\mathcal{G}_\alpha \in \mathcal{CE}$, $A_\alpha = \{e_1, e'_1, \ldots, e_n, e'_n\}$ such that the following hold:
   - $e_1 = (a_1, b_1)$; $e'_1 = (\overline{a_1}, b_2)$.
   - For each i, $1 < i < n$, $e_i = (a_i, \overline{b_{i-1}})$ and $e'_i = (\overline{a_i}, b_{i+1})$.
   - $e_n = (a_n, \overline{b_{n-1}})$; $e'_n = (\overline{a_n}, \overline{b_n})$.
2. If $\mathcal{G}_\alpha \in \mathcal{SO}$, let $\mathcal{G}_1(A_1)$ and $\mathcal{G}_2(A_2)$ be its two component natural subgraphs, where $A_\alpha = A_1 \cup A_2$. Then $A_1 = \{e_1, e'_1, \ldots, e_{n_1-1}, e'_{n_1-1}, e_{n_1}\}$, where e_i and e'_i are defined as above except that $e_{n_1} = (\overline{b_{n_1}}, \overline{b_{n_1-1}})$. Similarly, $A_2 = \{e_{n_1+1}, e'_{n_1+1}, \ldots, e_{n-1}, e'_{n-1}, e_n\}$ with $e_n = (\overline{b_n}, \overline{b_{n-1}})$.

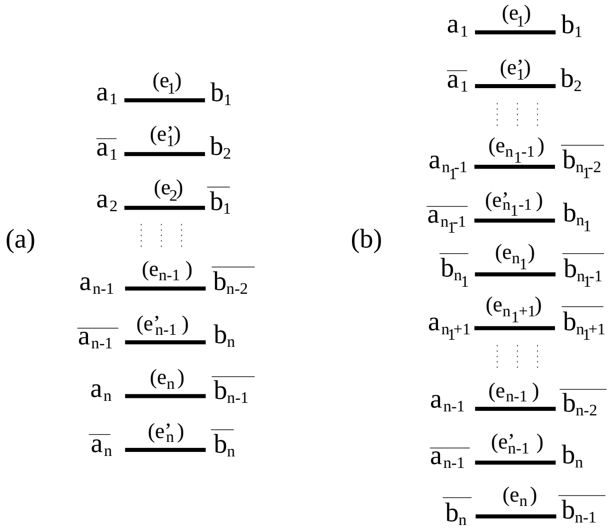

FIG. 5.2. *A suitable order for the edges of supernatural graphs.* (a) *A supernatural graph in $\mathcal{SE}$.* (b) *A supernatural graph in $\mathcal{SO}$.*

For an illustration, consider the supernatural graphs $\{\mathcal{S}_1, \mathcal{S}_{25}, \mathcal{S}_3, \mathcal{S}_4\}$ of our running example. By means of a relabeling of the vertices (a vertex x_1 could be relabeled as x_2, or vice-versa), one possible suitable order for the edges of the graphs is considered in Figure 7.5.

In the ensuing discussion, we start with any decomposition of $\mathcal{G}(\mathbf{V}, A)$ into a set $\mathcal{SN}$ of supernatural graphs in the suitable order.

As the dominant parameter in the **HP1** formula is the number of cycles, we begin by considering a set of valid gray edges maximizing the number of cycles of a completed graph. In the next section, we provide an upper bound on the number of cycles, and in section 7, we describe an algorithm for constructing a completed graph that allows us to reach this bound.

6. Upper bound on the number of cycles.

We need a preliminary definition.

DEFINITION 6.1. *Let $\mathcal{G}_\alpha(\mathbf{V}_\alpha, A_\alpha)$ be a supernatural graph of size $2n$. Consider the ordering of A_α described in the last section. Then $V_l = \bigcup_{1\leq i\leq n}\{a_i, \overline{a_i}\}$ is the set of* left vertices *of $\mathbf{V}_\alpha$, and $V_r = \bigcup_{1\leq i\leq n}\{b_i, \overline{b_i}\}$ is the set of* right vertices *of $\mathbf{V}_\alpha$.*

Note that from the definition, a natural subgraph of $\mathcal{SO}$ has four more right vertices than left vertices.

The set $\mathbf{V}$ is partitioned into subsets of left and right vertices: x is a left vertex in $\mathbf{V}$ if it is a left vertex of a graph of $\mathcal{SN}$. Otherwise, it is a right vertex.

Let $\mathcal{G}_\Gamma(\mathbf{V}, A, \Gamma)$ be a completed graph of $\mathcal{G}(\mathbf{V}, A)$, and let C be a particular cycle of size r of the graph with vertex set $\mathbf{V}_C$ and black and gray edge sets A_C and Γ_C, respectively. We define the *signature* S_C of C to be the subset of $\mathbf{V}_C$ derived as follows: For every left vertex x in $\mathbf{V}_C$, if $\overline{x}$ is not already in S_C, then add x to S_C.

Let $\mathcal{S}$ be the set of signatures of all the cycles of $\mathcal{G}_\Gamma$. Define the *signature graph* with the set of nodes $\mathcal{S}$ and the set of edges E as follows: for all $S_1, S_2 \in \mathcal{S}$, S_1 and S_2 are linked by an edge in E if and only if there is a vertex x such that $x \in S_1$ and $\overline{x} \in S_2$.

In Figure 6.1, a completed graph is given on the left. It represents a completed

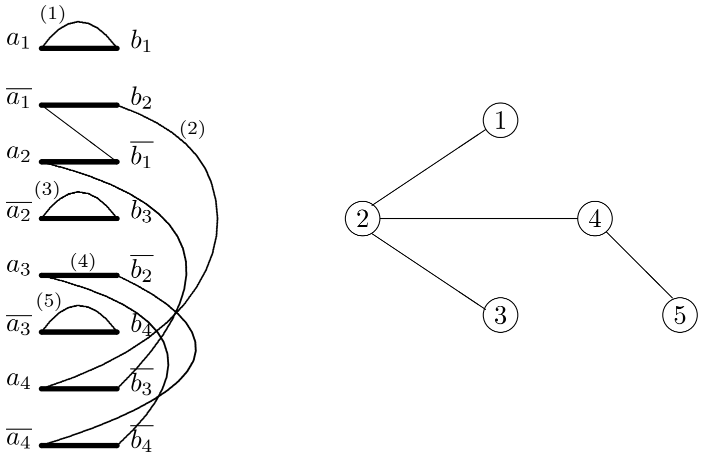

Fig. 6.1. *Example of a signature graph.*

supernatural graph of $\mathcal{SE}$. The completed graph is made up of five cycles, whose signatures are as follows:

$$1.\ \{a_1\};\quad 2.\ \{\overline{a_1}, a_2, a_4\};\quad 3.\ \{\overline{a_2}\};\quad 4.\ \{a_3, \overline{a_4}\};\quad 5.\ \{\overline{a_3}\}.$$

The graph on the right of Figure 6.1 is the signature graph derived from the graph on the left.

Lemma 6.2. *In the case of multichromosomal genomes, the signature graph of any completed supernatural graph is connected.*

Proof. The proof is deduced from the fact that $\mathcal{SN}$ is a set of smallest completable graphs: for any supernatural graph $\mathcal{G}_\alpha$, there does not exist any subgraph of $\mathcal{G}_\alpha$ that is also completable. □

For a node S_C of $\mathcal{S}$, denote by $t(S_C)$ the number of vertices in S_C and by $\delta(S_C)$ the number of outgoing edges.

Lemma 6.3. *For a multichromosomal genome, let $\mathcal{G}_e(\mathbf{V}_e, A_e)$ be a supernatural graph of size $2n$, where $n > 0$. Let $\mathcal{G}_e(\mathbf{V}_e, A_e, \Gamma_e)$ be a completed graph, and let c_e be its number of cycles. If $\mathcal{G}_e \in \mathcal{SE}$, then $c_e \leq n+1$. Otherwise ($\mathcal{G}_e \in \mathcal{SO}$), $c_e \leq n$.*

Proof. Let $\mathcal{S}$ be the set of vertices and E the set of edges of the signature graph of $\mathcal{G}_e(\mathbf{V}_e, A_e, \Gamma_e)$. Then $c_e = |\mathcal{S}|$.

For every $S_C \in \mathcal{S}$, $\delta(S_C) \leq t(S_C)$. Now $\sum_{S_C \in \mathcal{S}} t(S_C) \leq 2n$ so that

$$|E| = \frac{1}{2} \sum_{S_C \in \mathcal{S}} \delta(S_C) \leq \frac{1}{2} \sum_{S_C \in \mathcal{S}} t(S_C) \leq n.$$

From Lemma 6.2, a signature graph is connected so that $|\mathcal{S}| \leq |E| + 1 \leq n+1$.

For the case $\mathcal{G}_e(\mathbf{V}_e, A_e) \in \mathcal{SO}$, $\sum_{S_c \in \mathcal{S}} t(S_C) \leq 2n-2$. By the same argument as above,

$$|\mathcal{S}| \leq |E| + 1 = \frac{1}{2} \sum_{S_C \in \mathcal{S}} \delta(S_C) + 1 \leq \frac{1}{2} \sum_{S_C \in \mathcal{S}} t(S_C) + 1 \leq n. \qquad □$$

Results are slightly different for circular genomes.

Lemma 6.4. *In the case of circular genomes, the signature graph of any completed supernatural graph of $\mathcal{SE}$ is connected. On the other hand, at most one completed* *supernatural graph of $\mathcal{SO}$ has a signature graph with two connected components. The signature graph corresponding to any other graph of $\mathcal{SO}$ is connected.*

Proof. In the case of circular genomes, at most two natural graphs among all natural graphs of odd size are completable. This follows from the fact that a circular genome contains at most two adjacencies of form $(x, \overline{x})$. Therefore, as a supernatural graph of $\mathcal{SO}$ is obtained by concatenating two natural graphs of odd size, at most one completable supernatural graph of $\mathcal{SO}$ has a signature graph with two connected components. Any other graph of $\mathcal{SN}$ cannot be subdivided into smallest completable graphs and thus have signature graphs reduced to one connected component. □

Lemma 6.5. *For a circular genome, let $\mathcal{G}_e(\mathbf{V}_e, A_e, \Gamma_e)$ be a completed supernatural graph of size $2n$, and let c_e be its number of cycles. If $\mathcal{G}_e \in \mathcal{SE}$, then $c_e \leq n+1$. Moreover, there is at most one supernatural graph $\mathcal{G}_e$ of $\mathcal{SO}$ such that $c_e = n+1$. For all the other supernatural graphs of $\mathcal{SO}$, $c_e \leq n$.*

Proof. The proof is similar to that of Lemma 6.3 but uses the result of Lemma 6.4. □

Notation 2. We denote by $\boldsymbol{\gamma(G)}$ the number of "good" supernatural graphs:

- In the case of a multichromosomal genome G, $\gamma(G) = |\mathcal{SE}|$.
- In the case of a circular genome G, if $\mathcal{SO}$ is empty, then $\gamma(G) = |\mathcal{SE}|$; otherwise, $\gamma(G) = |\mathcal{SE}| + 1$.

Theorem 6.6. *Let $\mathcal{G}_\Gamma(\mathbf{V}, A, \Gamma)$ be a completed graph of $\mathcal{G}(\mathbf{V}, A)$, and let $c(\mathcal{G}_\Gamma)$ be its number of cycles. Then*

$$c(\mathcal{G}_\Gamma) \leq \frac{|A|}{2} + \gamma(G).$$

Proof. If any cycle C of $\mathcal{G}_\Gamma(\mathbf{V}, A, \Gamma)$ is "good," i.e., such that all black edges of C belong to the same supernatural graph of $\mathcal{SG}$, then, according to Lemmas 6.3 and 6.5, $c(\mathcal{G}_\Gamma) \leq \frac{|A|}{2} + \gamma(G)$.

Suppose now that there exist "bad cycles" in $\mathcal{G}_\Gamma(\mathbf{V}, A, \Gamma)$, i.e., cycles containing black edges of different supernatural graphs. Let c_b be the number of bad cycles, and let c_g be the number of good cycles of $\mathcal{G}_\Gamma(\mathbf{V}, A, \Gamma)$. Then $c(\mathcal{G}_\Gamma) = c_b + c_g$.

Let $\mathcal{G}_p(\mathbf{V}_p, A_p)$ be a supernatural graph, and let $\mathcal{C}_p$ be the set of cycles of $\mathcal{G}_\Gamma(\mathbf{V}, A, \Gamma)$ containing at least one edge in A_p. Let c_{g_p} be the number of good cycles and c_{b_p} the number of bad cycles of $\mathcal{C}_p$. Denote by $\{x_i, 1 \leq i \leq |\mathbf{V}_p|\}$ the set of vertices of $\mathbf{V}_p$.

Suppose that C is a bad cycle of $\mathcal{C}_p$ of size > 1. Denote $C = x_1x_2 - -x_3x_4 - - x_5x_6, \ldots$, where x_i's are the vertices in $\mathbf{V}_p$ and "$--$" denote paths in the cycle that do not contain any vertex in $\mathbf{V}_p$. Some of these paths can be empty.

We modify the bad cycles of $\mathcal{C}_p$ by the following procedure:

```
For any bad cycle $C = x_1x_2 - -x_3x_4 - -x_5x_6 \cdots$ and any $x_i$ with an even $i$, do
    If $x_{i+1} \neq \overline{x_i}$, do
        Remove the gray edges adjacent to $x_i$, $x_{i+1}$, $\overline{x_i}$, $\overline{x_{i+1}}$;
        Construct the gray edges $(x_i, x_{i+1})$ and $(\overline{x_i}, \overline{x_{i+1}})$;
    Else, there is another path of form $x_j - -\overline{x_j}$ (i.e., $x_{j+1} = \overline{x_j}$) either in $C$,
    or in another cycle of $\mathcal{C}_p$;
        Choose such a path, if possible in $C$, otherwise in another bad cycle,
        else, in a good cycle;
        Remove the gray edges adjacent to $x_i$, $x_{i+1}$, $x_j$, and $x_{j+1}$;
        Construct the gray edges $(x_i, x_j)$, $(x_{i+1}, y_{j+1})$, $(\overline{x_i}, \overline{x_j})$, and $(\overline{x_{i+1}}, \overline{x_{j+1}})$;
    End of If
End of For
```

The procedure constructs a completed graph $\mathcal{G}_p(\mathbf{V}_p, A_p, \Gamma_p)$. Let c_p be the number of cycles of $\mathcal{G}_p(\mathbf{V}_p, A_p, \Gamma_p)$. As the only way to decrease the number of cycles is to amalgamate pairs of bad cycles or to amalgamate at most once a bad cycle with a good one (lines 5 to 8 of the procedure), we have $c_p \geq c_{g_p} + c_{b_p} - \left\lceil \frac{c_{b_p}}{2} \right\rceil \geq c_{g_p} + \left\lfloor \frac{c_{b_p}}{2} \right\rfloor$. Let $c_{max,p}$ be the maximal number of cycles of a completed graph of $\mathcal{G}_p(\mathbf{V}_p, A_p)$. Then $c_{g_p} + \left\lfloor \frac{c_{b_p}}{2} \right\rfloor \leq c_{max,p}$.

Let now c_{max} be the maximal number of cycles of a completed graph of $\mathcal{G}(\mathbf{V}, A)$, and let c_d be the total number of bad cycles of $\mathcal{G}_\Gamma(\mathbf{V}, A, \Gamma)$. Then (1) $c_g + c_d \leq c_{max} + \left\lfloor \frac{c_d}{2} \right\rfloor \Longrightarrow c_g + \left\lceil \frac{c_d}{2} \right\rceil \leq c_{max}$.

On the other hand, as any bad cycle of a supernatural graph (a cycle containing at least one edge in the supernatural graph) corresponds to a bad cycle of another supernatural graph, the total number of cycles of $\mathcal{G}_\Gamma(\mathbf{V}, A, \Gamma)$ is (2) $c(\mathcal{G}_\Gamma) = c_g + c_b \leq c_g + \left\lceil \frac{c_d}{2} \right\rceil$. We deduce from inequalities (1) and (2) that $c(\mathcal{G}_\Gamma) \leq c_{max} \leq \frac{|A|}{2} + \gamma(G)$. □

7. Maximizing the number of cycles. Based on the decomposition of $\mathcal{G}(\mathbf{V}, A)$ into supernatural graphs, can we construct a completed graph $\mathcal{G}_\Gamma(\mathbf{V}, A, \Gamma)$ having $c(\mathcal{G}_\Gamma) = \gamma(G) + \frac{|A|}{2}$ cycles? By Theorem 6.6, this would necessarily be a *maximal completed graph*, that is, a completed graph with a maximal number of cycles. In this section, we focus on multichromosomal genomes. Modifications that have to be introduced in the case of circular genomes are presented in section 11.

We will use the following notation: for any set U of natural graphs, we denote by $\mathbf{V}_U$ the set of vertices of all natural graphs of U and by A_U the set of all black edges of U. For example, $\mathbf{V}_{\mathcal{SE}}$ will be the set of vertices of $\mathcal{SE}$.

We require a preliminary definition. A *fragment* of a genome is just a linear substring of G. For example, $F_1 = +g_1 \ -f_2 \ -d_2$ and $F_2 = 0_{11} \ +a_1 \ +b_1$ are two fragments of the genome represented by the partial graph of Figure 4.1. A fragment has two *endpoints*, unless it is restricted to one element of O. In the example given here, the two endpoints of F_1 are g_1^t and d_2^t, and the two endpoints of F_2 are 0_{11} and b_1^h. We call a fragment that has its two endpoints in V a $\mathcal{B}$-fragment.

Suppose that we have reached a certain step s in the construction, that Γ_s is the set of gray edges already constructed, and that $\mathcal{G}(\Gamma_s)$ is the "partially completed" graph obtained at this step. Suppose also that the natural graph being considered at this step is $\mathcal{G}_\alpha$, that the set of gray edges linking vertices of $\mathcal{G}_\alpha$ already constructed is $\Gamma_{s,\alpha}$, and that $\mathcal{G}_\alpha(\Gamma_{s,\alpha})$ is the obtained "partially completed" natural graph. A vertex of V is said to be *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 *unitary fragments*, which include not only $x^t x^h$ for all $x \in \mathcal{B}$ (the *$\mathcal{B}$-unitary fragments*) but also the $2N$ elements of O (the *O-unitary fragments*). As the construction proceeds, whenever a gray edge (x, y) is created, the fragment containing x and the one containing y are joined together.

DEFINITION 7.1. *Let $\mathcal{V}_s$ be a subset of the set of unlinked vertices at step s of the algorithm. The* border *of $\mathcal{V}_s$ is the set of all vertices x of $\mathcal{V}_s$ such that $x \in O$, or x is an endpoint of a $\mathcal{B}$-fragment $F \in \mathcal{F}$, and the second endpoint of F is not in $\mathcal{V}_s$.*

The graph $\mathcal{G}(\Gamma_s)$ is *bad* if there exists a subset U of $\mathcal{SN}$ such that the border of $\mathbf{V}_{s,U}$ is empty, where $\mathbf{V}_{s,U}$ is the set of unlinked vertices of $\mathbf{V}_U$ at step s. Otherwise, $\mathcal{G}(\Gamma_s)$ is a *good graph.* For an example, see Figure 7.1.

LEMMA 7.2. *Any set of gray edges linking the remaining unlinked vertices of a bad graph creates at least one circular fragment.*

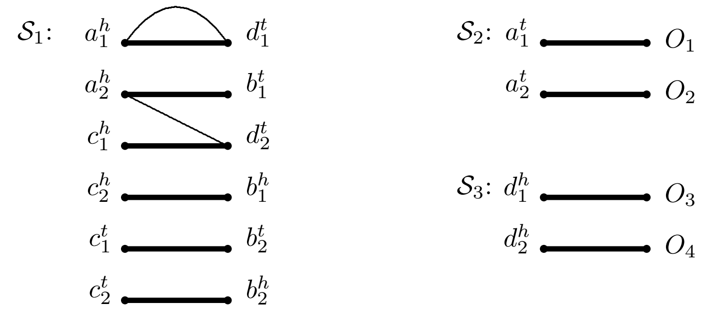

FIG. 7.1. *The partial graph corresponding to the genome with the two chromosomes:* O_1 $+a_1$ $+d_1$ O_3; O_2 $+a_2$ $+b_1$ $-c_2$ $-b_2$ $+c_1$ $+d_2$ O_4. *If we construct the two gray edges* $(a_1^h, d_1^t), (a_2^h, d_2^t)$*, the graph becomes bad, as the border of the supernatural graph* $\mathcal{S}_1$ *becomes empty.*

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

The above lemma implies that we have to be careful during the execution of the algorithm so as 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 circular 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 circular fragment if and only if one of the following properties is satisfied (see Figure 7.2).

Property I. The vertices $\{x, y\}$ are the endpoints of a $\mathcal{B}$-fragment of $\mathcal{F}$.

Property II. The pairs of vertices $\{x, \overline{y}\}, \{\overline{x}, y\}$ are the endpoints of two $\mathcal{B}$-fragments of $\mathcal{F}$.

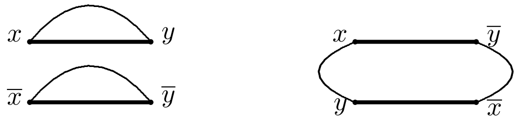

FIG. 7.2. *The left (resp., right) figure represents Property* I *(resp., Property* II*). Bold lines represent fragments, and thin lines represent the "potential" gray edges* $(x, y), (\overline{x}, \overline{y})$*. In any of these cases, the resulting fragment is circular.*

Now, let us 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, if any, is in $\mathcal{G}_\alpha$.

LEMMA 7.3. *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. Let U be any subset of $\mathcal{SN}$.

- Suppose F_1, F_2 have four endpoints, and all of these endpoints are in $\mathcal{G}_\alpha$. Then it is easy to see that linking F_1 to F_2 does not modify either the border corresponding to $\mathcal{G}_\alpha$ or that corresponding to U. Thus the state of $\mathcal{G}_\alpha$ (good or bad) could not have changed between steps s and $s+1$.

- Suppose that F_1, F_2 have at least three endpoints, and three such endpoints are in $\mathcal{G}_\alpha$. The subgraph U is bad if and only if $\mathbf{V}_{U,s+1}$ contains the fourth endpoint of F_1, F_2 not in $\mathcal{G}_\alpha$ and the border of $\mathbf{V}_{U,s+1}$ is empty. However, in that case, U would also have been bad at step s, which is a contradiction. □

For example, in Figure 7.1, the two gray edges $(a_2^h, b_1^t), (a_1^h, b_2^t)$ do not satisfy Property III, as the second endpoint of the fragment containing b_1^t is b_1^h, and b_1^h is in $\mathcal{S}_1$. Therefore, constructing these gray edges does not create a bad graph.

Corollary 7.4. *If a pair of potential gray edges* $\{(x,y), (\overline{x}, \overline{y})\}$ *of a good graph does not satisfy any of the Properties* I, II, *and* III, *then it is a possible pair of gray edges.*

Let x be an unlinked vertex of $\mathcal{G}_\alpha$. Then x is one of the two endpoints of a path (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. We say that a gray edge *closes the path* if and only if it links x to x^c.

Algorithm *dedouble* described in Figure 7.3 completes each supernatural graph of $\mathcal{SN}$, one after the other, in a specific order. The notation and edge order are those described in section 5.1.

Lemma 7.5. *At each step, algorithm* dedouble *constructs possible pairs of gray edges.*

Proof. Supernatural graphs of $\mathcal{SE}$*, with* $n=1$. At the beginning of the algorithm, the gray edges $(a_1, b_1), (\overline{a_1}, \overline{b_1})$ of $\mathcal{SE}$ are clearly possible, as they form fragments of the original genome G (Figure 7.4.(a)).

Suppose that we have reached a certain step in the construction and that the current supernatural graph of $\mathcal{SO}$ has the four vertices $b_1, b_2, \overline{b_1}, \overline{b_2}$ (Figure 7.4.(b)).

Suppose b_1, b_2 do not satisfy Property I, that is, they are the two endpoints of a fragment $F = b_1 \cdots b_2$. F cannot be a fragment of G, as in that case G would contain a circular fragment. Thus F should contain an adjacency (b_i, b_j) constructed from a supernatural subgraph of $\mathcal{SO}$, which means that $(a_i, \overline{a_i})$ and $(a_j, \overline{a_j})$ are two adjacencies in G. Then, if $F = b_1 \cdots a_i a_j \cdots b_2$, it is easy to see that G should contain two fragments of form $b_1 \cdots b_i \overline{b_i} \cdots \overline{b_1}$ and $b_2 \cdots b_j \overline{b_j} \cdots \overline{b_2}$. But since $(b_1, \overline{b_1}), (b_2, \overline{b_2})$ are two adjacencies in G (from the fact that the four vertices belong to a graph in $\mathcal{SO}$), this implies that G contains two circular fragments, which is impossible. Therefore, (b_1, b_2) (or, similarly, $(\overline{b_1}, \overline{b_2})$) does not create a circular fragment. We can prove in a similar way that $(b_1, b_2), (\overline{b_1}, \overline{b_2})$ do not satisfy Property II.

Suppose now that (b_1, b_2) creates a bad graph. Then there exists a subset U of $\mathcal{SN}$ such that the border of $\mathbf{V}_{U,s}$ is $B(U,s) = \{b_1, b_2, \overline{b_1}, \overline{b_2}\}$. This implies that the vertices of $\mathbf{V}_{U,s}$ belong to two fragments of G with the four endpoints $\{b_1, b_2, \overline{b_1}, \overline{b_2}\}$. This is also impossible as the two edges $(b_1, b_2), (\overline{b_1}, \overline{b_2})$ would give rise to a circular fragment in G.

Therefore, at the end of step 1 of the algorithm, the partial graph obtained is a good graph.

Supernatural graphs of $\mathcal{SE}$*, with* $n > 1$. Suppose that we have reached a good graph $\mathcal{G}(\Gamma_s)$ with a certain number of completed supernatural graphs. Suppose that $\mathcal{G}_\alpha$ is the supernatural graph currently being completed 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.

```
Algorithm dedouble:
Subgraphs in $\mathcal{SE}$, $n = 1$
Construct the gray edges $\{(a_1, b_1), (\overline{a_1}, \overline{b_1})\}$ (cf. Figure 7.4.(a));

Subgraphs in $\mathcal{SO}$, $n = 1$
Construct the gray edges $\{(b_1, b_2), (\overline{b_1}, \overline{b_2})\}$ (cf. Figure 7.4.(b));

Subgraphs in $\mathcal{SE}$, $n > 1$
For $i = 1$ to $n - 2$ Do
    Set $c = a_i^c$ and $d = \overline{a_i}^c$;
    If $p_{i,1} = \{(a_i, c), (\overline{a_i}, \overline{c})\}$ does not satisfy Properties I, II, and, III, Then
        Construct the gray edges of $p_{i,1}$;
    Else
        Construct the gray edges of $p_{i,2} = \{(\overline{a_i}, d), (a_i, \overline{d})\}$;
    End of if
End of for
Set $c = a_{n-1}^c$ and $d = \overline{a_{n-1}}^c$ (cf. Figure 7.4.(c));
If $p_{n-1,1} = \{(a_{n-1}, c), (\overline{a_{n-1}}, \overline{c})\}$ and $p_{n,1} = \{(a_n, d), (\overline{a_n}, \overline{d})\}$ do not
satisfy any of the Properties I, II, and III, Then
    Construct the gray edges of $p_{n-1,1}, p_{n,1}$;
Else
    Construct the gray edges of $p_{n-1,2} = \{(\overline{a_{n-1}}, d), (a_{n-1}, \overline{d})\}$ and
    $p_{n,2} = \{(a_n, \overline{c}), (\overline{a_n}, c)\}$;
End of if

Subgraph $\mathcal{G}_\alpha$ in $\mathcal{SO}$, $n > 1$
Let $\mathcal{G}_1, \mathcal{G}_2$ be the two natural graphs amalgamated to form $\mathcal{G}_\alpha$, and $n_1 > 1$;
For $i = 1$ to $n_1 - 2$ Do
    Construct gray edges as in the previous case;
End of for
For $i = n_1 + 1$ to $n - 1$ Do
    Construct gray edges as in the previous case;
End of for
Set $c = a_{n_1-1}^c, d = \overline{a_{n_1-1}}^c$ (cf. Figure 7.4.(d));
Let $e, \overline{e}$ be the only unlinked vertices in $\mathcal{G}_2$;
If $p_{n-1,1} = \{(a_{n_1-1}, c), (\overline{a_{n-1}}, \overline{c})\}$ and $p_{n,1} = \{(e, d), (\overline{e}, \overline{d})\}$ do not
satisfy any of the Properties I, II, and III, Then
    Construct the gray edges of $p_{n-1,1}, p_{n,1}$;
Else
    Construct the gray edges of $p_{n-1,2} = \{(\overline{a_{n_1-1}}, d), (a_{n_1-1}, \overline{d})\}$ and
    $p_{n,2} = \{(c, e), (\overline{c}, \overline{e})\}$;
End of if
```

FIG. 7.3. *Algorithm for constructing a maximal completed graph.*

Suppose now that $p_{i,1}$ is impossible. We want to prove that $p_{i,2}$ is possible. Suppose $p_{i,1}$ satisfies Property I. That means that a_i and c are the endpoints of the same fragment F. Therefore, a_i and $\overline{d}$ cannot be the endpoints of the same fragment, which means that $p_{i,1}$ does not satisfy Property I. The vertices a_i and d are not the endpoints of the same fragment either, which means that $p_{i,2}$ does not satisfy

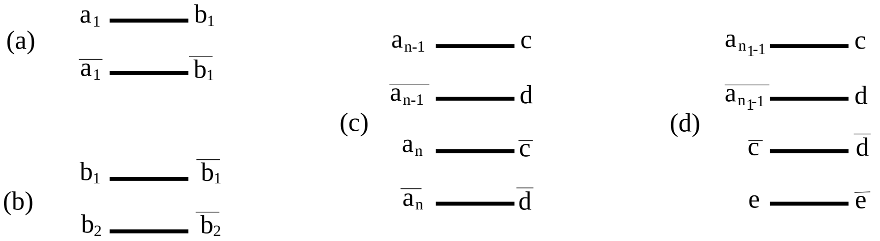

FIG. 7.4. *Different situations considered by* dedouble. (a) *A supernatural graph of $\mathcal{SE}$, with $n = 1$.* (b) *A supernatural graph of $\mathcal{SO}$, with $n = 1$.* (c) *The last step for a supernatural graph of $\mathcal{SE}$; corresponds to lines* 10 *to* 15 *of the algorithm.* (d) *The last step for a supernatural graph of $\mathcal{SO}$; corresponds to lines* 23 *to* 29 *of the algorithm. In* (c) *and* (d), *edges represent paths that can contain more than one edge.*

Property II. Now, since a_i and $\overline{d}$ are two endpoints of two fragments, one of them, which is F having both endpoints in $\mathcal{G}_\alpha$, $p_{i,2}$, does not satisfy Property III either.

We prove similarly that, if $p_{i,1}$ satisfies Property II, then $p_{i,1}$ cannot satisfy any of the three properties.

Suppose now that $p_{i,1}$ creates a bad graph. That means that there exists a subset U of $\mathcal{E}$ such that the border of $V_{U,s}$ is $B(U,s) = \{a_i, \overline{a_i}, c, \overline{c}\}$; then a_i and c should belong to two different fragments with the two other endpoints not in $\mathcal{G}_\alpha$. Then clearly $p_{i,2}$ cannot satisfy Property I or Property II. Suppose that it satisfies Property III. That means that there exists 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. The other vertices of $U \cup U'$ cannot be in O (as, otherwise, these vertices would have been part of the border), and if u is in $U \cup U'$, then $\overline{u}, u^s, \overline{u}^s$ are also in $U \cup U'$. Therefore, the number of vertices of $U \cup U'$ is $4m + 2$ for some m. However, 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 $p_{n-1,2}$ and $p_{n,2}$ are possible (Figure 7.4(c)).

Suppose (a_{n-1}, c) (and $(\overline{a_{n-1}}, \overline{c})$) satisfies Property I, that is, a_{n-1} and c are the endpoints of a fragment F. Then clearly neither $(\overline{a_{n-1}}, d)$ nor $(\overline{a_n}, c)$ satisfies Property I or Property II. Suppose $(\overline{a_{n-1}}, d), (\overline{a_n}, c)$ give rise to one circular fragment. This is possible if a_{n-1}, c and $\overline{a_n}, d$ are the endpoints of two fragments. However, in that case, the supernatural graph $\mathcal{G}_\alpha$ would have had an empty border just before the current step, and the graph would have been a bad graph. However, this contradicts the recurrence hypothesis. Suppose finally that $p_{n-1,2}$ and $p_{n,2}$ create a bad graph. Then there exists a subset U of $\mathcal{SN}$ such that the border of U is in $\{a_{n-1}, \overline{a_{n-1}}, a_n, \overline{a_n}, c, \overline{c}, d, \overline{d}\}$. However, just before this step, $U \cup \{\mathcal{G}_\alpha\}$ would have been a bad graph, which contradicts the recurrence hypothesis.

The remaining cases are treated in a similar way, and we prove with similar arguments that, in any of these cases, $p_{n-1,2}, p_{n,2}$ are possible.

Supernatural graphs of $\mathcal{SO}$ with $n > 1$. The construction method for $1 \leq i \leq n_1 - 2$ and $n_1 + 1 \leq i \leq n - 1$ is identical to that in a supernatural subgraph of $\mathcal{SE}$ for $1 \leq i \leq n - 2$. Therefore, the same proof as before holds in that case. Finally, we should prove that, if $p_{n-1,1}$ and $p_{n,1}$ are impossible, then $p_{n-1,2}$ and $p_{n,2}$ are possible. To do so, arguments similar to those for a supernatural subgraph of $\mathcal{SE}$ are used to treat each case. □

Example 3. Consider the genome G of Example 1 and the decomposition of its partial graph into the supernatural graphs $\{\mathcal{S}_1, \mathcal{S}_{25}, \mathcal{S}_3, \mathcal{S}_4\}$. Figure 7.5 depicts the completed graph produced by *dedouble*.

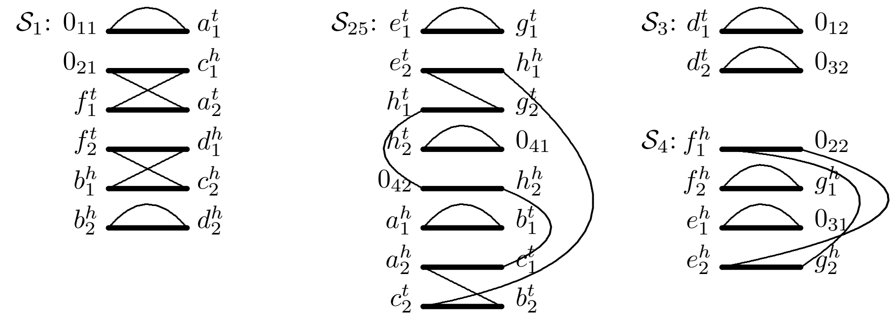

FIG. 7.5. *The completed graph* $\mathcal{G}_\Gamma(\mathbf{V}, A, \Gamma)$ *constructed by* dedouble.

The number of cycles in the completed graph is $c(\mathcal{G}) = 13$. As $\gamma(G) = 3$ and $|A| = 20$, according to Theorem 6.6, it is a maximal completed graph.

The corresponding duplicated genome H is made up of the four chromosomes

1. $O_{11} + a_1 \ + b_1 \ - d_1 \ O_{12}$;
2. $O_{21} + a_2 \ + b_2 \ - d_2 \ O_{32}$;
3. $O_{42} \ + h_1 \ + c_2 \ + f_2 \ - g_1 \ + e_1 \ O_{31}$;
4. $O_{41} \ + h_2 \ + c_1 \ + f_1 \ - g_2 \ + e_2 \ O_{22}$.

THEOREM 7.6. *Algorithm* dedouble *constructs a maximal completed graph* $\mathcal{G}_\Gamma(\mathbf{V}, A, \Gamma)$*, containing* $c(G) = \frac{|A|}{2} + \gamma(G)$ *cycles.*

Proof. To prove this result, it is sufficient to prove that, for every supernatural graph $\mathcal{G}_\alpha$ with $2n$ black edges, if $\mathcal{G}_\alpha \in \mathcal{SE}$, then the number of cycles in the completed graph obtained by *dedouble* is $c(\mathcal{G}_\alpha) = n+1$, and if $\mathcal{G}_\alpha \in \mathcal{SO}$, this number is $c(\mathcal{G}_\alpha) = n$.

Let $\mathcal{G}_\alpha$ be a supernatural graph of $\mathcal{SE}$. For each i, $1 \leq i \leq n-2$, the algorithm constructs either (a_i, a_i^c) or $(\overline{a_i}, \overline{a_i}^c)$. Thus, at each step of the construction, at least one path is closed to form a cycle. Finally, it is easy to see, from Figure 7.4(c), that instructions 11–16 close three more cycles. Therefore, in total, at least $n+1$ cycles are formed in $\mathcal{G}_\alpha$. According to Lemma 6.3, the maximal number of cycles of a completed graph of $\mathcal{SE}$ is $n+1$. Therefore, $c(\mathcal{G}_\alpha) = n+1$.

Similarly, for a supernatural graph $\mathcal{G}_\alpha$ of $\mathcal{SO}$ with $2n$ black edges, steps 17–22 of the algorithm close at least $n-2$ cycles. Then it is easy to see, from Figure 7.4(d), that instructions 24–29 close two more cycles. Therefore, as n is the maximal number of cycles of a completed graph of $\mathcal{SO}$ (Lemma 6.3), $c(\mathcal{G}_\alpha) = n$. □

The following theorem is a direct consequence of Theorems 6.6 and 7.6.

THEOREM 7.7. *The number of cycles of a maximal completed graph of* $\mathcal{G}(\mathbf{V}, A)$ *is*

$$c(G) = \frac{|A|}{2} + \gamma(G).$$

Complexity. At each step, algorithm *dedouble* considers at most four black edges of the graph and constructs two gray edges with four vertices of the considered black edges. Choosing the right vertices to connect requires checking Properties I, II, and III for at most two pairs of gray edges. This is clearly done in constant time. Thus each step of the algorithm takes constant time. As each step constructs two gray edges, the graph is completed in $\frac{|A|}{2}$ steps. Therefore, the time complexity of algorithm *dedouble* is $O(|A|)$.

8. Bad components. We turn now our attention to minimizing the number of bad components of a completed graph. Even if the concept of bad components is different for each of the three models considered in this paper (translocations-only, reversals-only, or both reversals and translocations), it is always related to the notion of "subpermutation" introduced by Hannenhalli [16] and summarized below.

Given two genomes H_1 and H_2 containing the same gene set, where each gene appears exactly once in each genome, a subpermutation of H_1 (or, similarly, of the breakpoint graph $\mathcal{G}_{12}$ associated with H_1 and H_2) is a subsequence $S = u_1u_2, \cdots u_{p-1}u_p$ of a chromosome X of H_1 such that there is a permutation P and a subsequence $T = P(S) = u_1v_2 \cdots v_{p-1}u_p$ of a chromosome Y of H_2, with $v_2 \neq u_2$ and $v_{p-1} \neq u_{p-1}$. A *minimal SP* (minSP) is an SP not containing any other SP, and a *maximal SP* (maxSP) is an SP not included in any other SP.

We call *the interval of a component* C the interval $I = [u_l, u_r]$, where u_l and u_r are the endpoints of C. The interval I is such that no gray edge links a vertex of I to a vertex outside of I, and at least one cycle of I is of size greater than 1. A *minimal component* is a component whose interval contains no other component. There is a bijection between the SPs of $\mathcal{G}_{12}$ and the components of $\mathcal{G}_{12}$. More precisely, let S be an SP, let $\Pi = \{\pi_1, \ldots, \pi_p\}$ be the set of components containing the vertices of S, and for any i, let $\mathbf{V_i}$ be the set of vertices of π_i. Then the following hold:

- S_i is an SP contained in S (inner SP of S, possibly S itself) if and only if S_i corresponds to an interval of a component π_i of Π. We call this component the *component of the SP* S_i.
- S_i is a minimal inner SP of S if and only if S_i corresponds to an interval of a minimal component of Π.

Example 4. Consider the following two circular genomes and the corresponding breakpoint graph (Figure 8.1):

$$G = +a_1 \ +b_1 \ +c_1 \ +d_1 \ +e_1 \ +d_2 \ -f_1 \ -e_2 \ -f_2 \ +a_2 \ -b_2 \ +c_2,$$
$$H = +a_1 \ +b_1 \ +c_1 \ +d_1 \ +e_1 \ +f_2 \ -f_1 \ -e_2 \ -d_2 \ -c_2 \ -b_2 \ -a_2.$$

Each of the three components of this graph is made up of a single cycle.

$\mathcal{C}_1$ is the component of the SP $S_1 = +e_1 \ +d_2 \ -f_1 \ -e_2 \ -f_2 \ +a_2 \ -b_2 \ +c_2 \ +a_1$.

$\mathcal{C}_2$ is the component of the SP $S_2 = +d_2 \ -f_1 \ -e_2 \ -f_2$.

$\mathcal{C}_3$ is the component of the SP $S_3 = +a_2 \ -b_2 \ +c_2$.

The only two minSPs are S_2 and S_3.

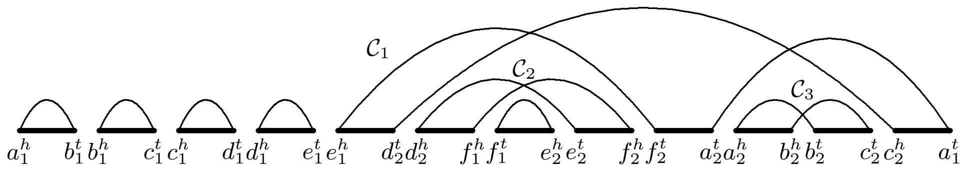

FIG. 8.1. *Breakpoint graph corresponding to genomes G and H.*

For the problem of rearrangement by translocations [16], all minSPs are bad components of an HP graph. More precisely, if $s(\mathcal{G}_{12})$ is the number of minSPs of $\mathcal{G}_{12}$, then, in formulae **HP1** (section 3), $m(\mathcal{G}_{12}) = s(\mathcal{G}_{12})$. For the problem of rearrangement by reversals, or by reversals and translocations, certain SPs can still be solved by proper operations, while others, the "bad components," require bad operations to be solved. The *hurdles* in the case of reversals [17] and the *knots* in the case of reversals and translocations [18] are the bad (intrachromosomal) minSPs and maxSPs.

Returning to our genome halving problem, we want to determine the minimal number of such (bad) SPs in a completed graph of $\mathcal{G}(\mathbf{V}, A)$. In the case of circular genomes, we need to distinguish between SPs that do not contain both x and $\overline{x}$ for the same vertex x, which we call *normal*, and those that do, the *special* ones. As duplicated multichromosomal genomes cannot have both x and $\overline{x}$ on one chromosome, all SPs are normal for multichromosomal genomes. In the rest of the paper, if not specified, an SP will designate a normal one.

Definition 8.1. *Let $S = x_1x_2 \cdots x_{n-1}x_n$ be a subsequence of a chromosome of G. S is a* local SP *of G if S is a* real local SP *or a* potential local SP*, namely:*

- *S is a real local SP of G if $\{x_1, \ldots, x_n\} \cap O = \emptyset$ and there exists another subsequence of a chromosome of G of form $\overline{S} = \overline{x_1}P(\overline{x_2}, \ldots, \overline{x_{n-1}})\overline{x_n}$, where P is a permutation other than the identity.*
- *S is a potential local SP if either* i. *$\{x_1, x_n\} \subset O$, and there exists a chromosome of G containing a subsequence $\overline{S} = O_1P(\overline{x_2}, \ldots, \overline{x_{n-1}})O_2$, where P is a permutation other than the identity and $\{O_1, O_2\} \in O$, or* ii. *$x_1 \in O$, and there exists a chromosome containing a subsequence $\overline{S} = O_1P(\overline{x_2}, \ldots, \overline{x_{n-1}})x_n$, where P is a permutation other than the identity and $O_1 \in O$. An analogous condition holds for x_n.*

We call $\overline{S}$ the complementary sequence *of S. We say that a local SP (real or potential) S is* minimal *if it does not contain any subsequence corresponding to another local SP.*

For circular genomes, as the notion of endpoints is irrelevant, potential SPs do not exist and all local SPs are real ones.

Example 5. Let $G = +a_1\ +b_1\ +c_1\ +d_1\ +e_1\ -d_2\ +b_1\ +c_1\ -a_2\ +e_2$. The subsequence $S = +a_1 +b_1 +c_1 +d_1$ is a local SP of G. In the genome G of Example 4, the subsequence $+a_1 +b_1 +c_1$ is a local SP of G.

8.1. Correcting the completed graph obtained by algorithm *dedouble*.

In this section, we describe a modification of algorithm *dedouble* that will be used to produce an optimal completed graph (i.e., a completed graph giving rise to a duplicated genome minimizing the rearrangement distance to G).

Let $\mathcal{G}_\Gamma(\mathbf{V}, A, \Gamma)$ be a maximal completed graph produced by *dedouble*, and let $S = x_1 \cdots x_n$ be an SP of $\mathcal{G}_\Gamma(\mathbf{V}, A, \Gamma)$. The following procedure applies to the SP S.

Procedure spoil-SP(S):

```
Remove all the edges of $\Gamma$ adjacent to the vertices of $\{x_1, \ldots, x_n, \overline{x_1}, \ldots, \overline{x_n}\}$;
Construct the edges $(x_k, x_{k+1})$ and $(\overline{x_k}, \overline{x_{k+1}})$ for all $k$, $1 \leq k < n$.
```

Consider the maximal completed graph $\mathcal{G}_\Gamma(\mathbf{V}, A, \Gamma)$ produced by *dedouble*. Let $\mathcal{S}$ be the set of SPs in $\mathcal{G}_\Gamma(\mathbf{V}, A, \Gamma)$ that do not correspond to local SPs of G. Correcting $\mathcal{G}_\Gamma$ consists in applying *spoil-SP* to each $S \in \mathcal{S}$.

Complexity. To correct the completed graph $\mathcal{G}$ produced by *dedouble*, we have to consider all the SPs of $\mathcal{G}$. This problem is equivalent to that of decomposing a breakpoint graph into its components. As shown in [22], this can be done in time $O(|A|)$. Then, verifying if an SP is a local SP of G and applying procedure *spoil-SP* takes time linear in the number of edges of the considered SP. Therefore, the total time needed to correct the graph is linear in the number of black edges $|A|$ of the graph. As algorithm *dedouble* has also been shown to be linear in $|A|$, the complexity of the whole algorithm (*dedouble* and graph correction) is $O(|A|)$.

8.2. Genome with no local SP. In this section, we show that for a genome with no local SP, the corrected graph is optimal.

LEMMA 8.2. *Suppose that the completed graph $\mathcal{G}_\Gamma(\mathbf{V}, A, \Gamma)$ produced by* dedouble *contains an SP S. If S is not a local SP of G, then* spoil-SP*(S) gives rise to a completed graph $\mathcal{G}_{\Gamma'}(\mathbf{V}, A, \Gamma')$ containing at least the same number of cycles as $\mathcal{G}_\Gamma(\mathbf{V}, A, \Gamma)$ and one less SP.*

Proof. Let $\mathcal{C}$ be the set of cycles of $\mathcal{G}_\Gamma(\mathbf{V}, A, \Gamma)$ containing at least one vertex in $V(x, \overline{x}) = \{x_1, \ldots, x_n, \overline{x_1}, \ldots, \overline{x_n}\}$, $\mathcal{C}(x)$ the subset of $\mathcal{C}$ containing cycles with at least one vertex in $V(x) = \{x_1, \ldots, x_n\}$, and $\mathcal{C}(\overline{x})$ the subset of $\mathcal{C}$ containing cycles with at least one vertex in $V(\overline{x}) = \{\overline{x_1}, \ldots, \overline{x_n}\}$. Let $c = |\mathcal{C}|$, $c(x) = |\mathcal{C}(x)|$, and $c(\overline{x}) = |\mathcal{C}(\overline{x})|$. As S is an SP, $\mathcal{C}(x) \cap \mathcal{C}(\overline{x}) = \emptyset$.

Let $\mathcal{CP}$ be the set of cycles pairs (C_k, C_l) of $\mathcal{C}(x)$ such that $C_k \neq C_l$, and there is a vertex x_k in C_k and a vertex x_l in C_l such that $\overline{x_k}$ and $\overline{x_l}$ belong to the same cycle in $\mathcal{C}(\overline{x})$. Let $\pi = |\mathcal{CP}|$.

There are at most $\frac{n}{2} - \pi$ cycles in $\mathcal{C}(\overline{x})$, so $c \leq c(x) + \frac{n}{2} - \pi$.

Let $\mathcal{G}_{\Gamma'}(\mathbf{V}, A, \Gamma')$ be the graph obtained after applying procedure *spoil-SP*$(x_1, \ldots, x_n)$, and let $\mathcal{C}'$, $\mathcal{C}'(x)$ and $\mathcal{C}'(\overline{x})$ be the sets defined respectively as $\mathcal{C}$, $\mathcal{C}(x)$ and $\mathcal{C}(\overline{x})$ but for $\mathcal{G}_\Gamma(\mathbf{V}, A, \Gamma')$. Let $c' = |\mathcal{C}'|$, $c'(x) = |\mathcal{C}'(x)|$, and $c'(\overline{x}) = |\mathcal{C}'(\overline{x})|$. As only size 1 cycles are formed with $V(x)$ vertices, $c'(x) = \frac{n}{2}$.

Let $C_1, \ldots, C_m$ be the cycles of size > 1 of $\mathcal{C}(x)$ for all r, $1 \leq r \leq m$, $|C_r| = p_r$, and let $\{x_{r_1}, \ldots, x_{r_{q_r}}\}$ be the vertices of $V(x)$ contained in C_r. Suppose we transform C_1 into $\frac{p_1}{2}$ size 1 cycles. The vertices $\{\overline{x_{1_1}}, \ldots, \overline{x_{1_{q_1}}}\}$ belong to at least one cycle. Then transform C_2 into $\frac{n}{2}$ size 1 cycles. If $(C_1, C_2) \in \mathcal{CP}$, then the vertices $\{\overline{x_{1_1}}, \ldots, \overline{x_{1_{q_1}}}, \overline{x_{2_1}}, \ldots, \overline{x_{2_{q_2}}}\}$ belong to at least 2 cycles; otherwise, they belong to at least 1 cycle.

By continuing this reasoning until C_m, we show that $c'(\overline{x}) \geq c(x) - \pi$. Thus $c' \geq \frac{n}{2} + c(x) - \pi$, and so $c \leq c'$. The completed graph $\mathcal{G}_{\Gamma'}(\mathbf{V}, A, \Gamma')$ is then also maximal but no longer contains the SP $x_1 \cdots x_n$.

Suppose that the procedure *spoil-SP*$(x_1 \cdots x_n)$ creates an SP that was not in $\mathcal{G}_\Gamma(\mathbf{V}, A, \Gamma)$. Since the only modified edges are those linking vertices of $V(x, \overline{x})$, this new SP has to be formed by vertices in $V(\overline{x})$. Thus $x_1 \cdots x_n$ is necessarily a local SP of G. □

As a corollary to Lemma 8.2 we have the following.

COROLLARY 8.3. *For a genome G with no local SP, the corrected graph produced by* dedouble *is maximal and contains no SP.*

Then, from the formula **HP1** (section 3) and from the fact that, for all three rearrangement models considered, a bad component is attached to an SP of the graph (section 8), if G is a genome with no local SP, then the corrected graph produced by *dedouble* is optimal (i.e., gives rise to a duplicated genome minimizing the rearrangement distance to G).

In the remainder of this paper, it will be implicit that the correction of the graph, as described in the previous section, is incorporated at the end of algorithm *dedouble*. We turn next to the case in which G contains local SPs.

8.3. General formula for the rearrangement distance. The next lemma shows that any maximal completed graph should contain at least as many SPs as the number of real local SPs of G.

LEMMA 8.4. *Suppose that G contains a real local SP $S = x_1 \cdots x_n$. Suppose that the completed graph $\mathcal{G}_\Gamma(\mathbf{V}, A, \Gamma)$ contains no SP made up of the vertices $\{x_1, \ldots, x_n\}$.*

If c_{max} *is the maximal number of cycles of a completed graph of* $\mathcal{G}(\mathbf{V}, A)$ *and* $c(\mathcal{G}_\Gamma)$ *is the number of cycles of* $\mathcal{G}_\Gamma$*, then* $c(\mathcal{G}_\Gamma) \leq c_{max} - 2$.

Proof. Let $X_S = \{x_1, \ldots, x_n\}$ be the vertices of a subsequence $S = x_1 \cdots x_n$ of a certain chromosome of G, and let $\overline{X_S} = \{\overline{x_1}, \ldots, \overline{x_n}\}$ be the vertices of the complementary sequence $\overline{S}$ contained in a chromosome of G (another one or the same).

Let $\mathcal{G}_\Gamma(\mathbf{V}, A, \Gamma)$ be a maximal completed graph. Suppose that some vertices in X_S are linked by gray edges to vertices outside X_S. Let X be this set of vertices. Vertices in X are of two types: those linked to vertices in $\overline{X}$ and those linked to vertices outside $X \cup \overline{X}$. Denote $X_1 = \{x_{k_1}, \ldots, x_{k_l}\}$ as the set of l vertices of the first type, $X_2 = \{x_{p_1}, \ldots, x_{p_m}\}$ as the set of m vertices of the second type, and $Y = \{y_1, \ldots, y_m\} \subset \mathbf{V} \setminus X \cup \overline{X}$ as the vertices adjacent to them.

As all X vertices are adjacent to each other by black edges, a cycle containing a vertex in $X \cup Y$ contains at least two vertices of this set. Thus at most $\frac{l+m}{2}$ cycles contain a vertex in $X \cup Y$. Similarly, at most $\frac{m}{2}$ cycles contain a vertex in $\overline{X_2} \cup \overline{Y}$. Moreover, a cycle containing a vertex in $\overline{X_1}$ should contain a vertex in X_1. Therefore, the number of cycles containing a vertex in $X \cup Y \cup \overline{XY}$ is at most $\frac{l+m}{2} + \frac{m}{2} = m + \frac{l}{2}$.

Now, let $\mathcal{G}_{\Gamma'}(\mathbf{V}, A, \Gamma')$ be the completed graph obtained from $\mathcal{G}_\Gamma(\mathbf{V}, A, \Gamma)$ by the following procedure:

```
For all $x \in X$ do
    - Remove gray edges adjacent to $x$ and $\overline{x}$;
    - Construct the gray edge $(x, x')$, where $x'$ is the vertex in $X$ linked to $x$ by a
      black edge;
    - Construct the gray edge $(\overline{x}, \overline{x'})$;
For all $y \in Y$ do
    - Construct the gray edge $(y, y')$, where $y'$ is the vertex in $\mathbf{V} \setminus X$ linked to $y$
      by a black edge;
    - Construct the gray edge $(\overline{y}, \overline{y'})$.
```

Then exactly $\frac{l+m}{2}$ cycles have vertices in X and they are all of size 1, and exactly $\frac{m}{2}$ cycles have vertices in Y and they are also of size 1. Moreover, there is no cycle containing at the same time a vertex in $\overline{X}$ and another one in $\overline{Y}$, and there are at least two cycles with a vertex in $\overline{X}$ or a vertex in $\overline{Y}$. Therefore, the number of cycles containing vertices in $X \cup Y \cup \overline{XY}$ is at least $\frac{l+m}{2} + \frac{m}{2} + 2 = m + \frac{l}{2} + 2$. As the above procedure does not modify the other cycles, $\mathcal{G}_{\Gamma'}(\mathbf{V}, A, \Gamma')$ has at least two more cycles than $\mathcal{G}_\Gamma(\mathbf{V}, A, \Gamma)$, which is a contradiction with the fact that $\mathcal{G}_\Gamma(\mathbf{V}, A, \Gamma)$ is a maximal completed graph. □

Remark 1. Let S be a local SP of G and $\overline{S}$ the complementary sequence of S. We can suppose, without loss of generality, that *dedouble* constructs only cycles of size 1 with vertices of $\overline{S}$ and that the SPs of the final completed graph are formed by the vertices of S. (We can always modify the resulting completed graph so that it satisfies these properties.)

For multichromosomal genomes, potential SPs give rise to additional problems. The goal is to minimize the number of such potential SPs that become SPs of the final completed graph. For circular genomes, all local SPs are real ones. However, in that case, one additional problem is due to special SPs.

Let $RO(G)$ be the minimal number of rearrangement operations required to transform G into a duplicated genome. Though $RO(G)$ is different depending on the model considered (reversals, translocations, reversals and translocations), we will prove in the coming sections that all results can be summarized by the following formula:

$$RO(G) = \frac{|A|}{2} - \gamma(G) + m(G) + \phi(G),$$

where $m(G)$ is the number of bad real local SPs of G and $\phi(G)$ is a correction depending on bad potential local SPs (for multichromosomal genomes) and special SPs (for circular genomes).

Moreover, we will show that, with an appropriate construction of natural graphs, and with other minor corrections in the case of sorting by translocations and reversals, the completed graph produced by algorithm *dedouble* gives rise to the rearrangement distance.

9. Genome halving with translocations only. For two multichromosomal genomes H_1 and H_2, if $\mathcal{G}_{12}$ is the breakpoint graph associated to H_1 and H_2, the minimal number $T(H_1, H_2)$ of translocations required to transform H_1 to H_2 is given by the formulae proved in [16]:

$$\textbf{HP2}: \quad T(H_1, H_2) = b(\mathcal{G}_{12}) - c(\mathcal{G}_{12}) + s(\mathcal{G}_{12}) + f(\mathcal{G}_{12}),$$

where $s(\mathcal{G}_{12})$ is the number of minSPs of H_1. In other words, in the formula **HP1**, we have $m(\mathcal{G}_{12}) = s(\mathcal{G}_{12})$.

The value of the parameter $f(\mathcal{G}_{12})$ depends on a characteristic of the breakpoint graph, defined in [16]. The graph $\mathcal{G}_{12}$ has an *even-isolation* if the next three conditions are satisfied:

1. All minSPs of $\mathcal{G}_{12}$ are on a single chromosome of H_1.
2. $s(\mathcal{G}_{12})$ is even.
3. All minSPs are contained within a single SP.

If $\mathcal{G}_{12}$ has an even-isolation, then $f(\mathcal{G}_{12}) = 2$; if $\mathcal{G}_{12}$ has an odd number of minSPs, then $f(\mathcal{G}_{12}) = 1$; otherwise, $f(\mathcal{G}_{12}) = 0$ [16].

Returning to our problem of genome halving, denote by $\mathbf{T(G)}$ the minimal number of translocations required to transform G into a perfectly duplicated genome. In section 7, we described an algorithm for constructing a maximal completed graph in the case of multichromosomal genomes. We also proved, in section 8, that the minimal number of SPs of a completed graph can be deduced from the local (real or potential) SPs of G. The following corollary is a direct consequence of these results (Theorem 7.7, Corollary 8.3, and formula **HP2**).

Corollary 9.1. *If G does not contain any local SP, then $T(G) = |A| - c(G) = \frac{|A|}{2} - \gamma(G)$.*

Moreover, in section 8, we treated the case of real SPs. It remains now to consider potential local SPs.

Let S be a potential SP with two ends O_1, O_1' in O, and let O_2, O_2' be the two ends of $\overline{S}$. S becomes a real SP if and only if $\overline{O_2} = O_1$ and $\overline{O_2'} = O_1'$. Similarly, let S be a potential SP with only one end O_1 in O, and let O_2 be the vertex of $\overline{S}$ in O. S becomes a real SP if and only if $\overline{O_2} = O_1$. The problem is to avoid such situations.

According to formula **HP2**, we need only to minimize the number of minSPs of a completed graph (instead of SPs). Therefore, we consider only potential local minSPs. In the ensuing discussion, we just call them potential SPs.

Remark 2. Let S be a potential SP and $V(S, \overline{S})$ the set of vertices of S and $\overline{S}$ excluding those in O. The number of natural graphs containing vertices both in

$V(S, \overline{S})$ and O is exactly two if S has both its ends in O, and one if S has only one end in O. We call these graphs the *graphs associated to* S.

We distinguish between two kinds of potential SPs.

Definition 9.2. *A potential SP is* even *(PES) if its associated graphs (one or two) are in $\mathcal{NE}$, i.e., are of even size. Otherwise, the potential SP is* odd *(POS). A POS necessarily has both its ends in O and thus two associated graphs in $\mathcal{NO}_+$.*

Notation 3. We denote $\mathcal{PES}$ the set of all PES, and $e = |\mathcal{PES}|$. For i, $1 \leq i \leq e$, P_i is the set of (one or two) graphs associated to the ith PES for an arbitrary ordering of the PESs.

We denote $\mathcal{POS}$ the set of all POS, and $o = |\mathcal{POS}|$. For every i, $1 \leq i \leq o$, denote by $Q_i = (A_i, A'_i)$ the pair of graphs associated to the ith POS for an arbitrary ordering of the POSs.

In section 5, we arbitrarily amalgamated pairs of natural graphs of odd size to form supernatural graphs. To avoid transforming a POS into an SP, we introduce a more deterministic way to amalgamate graphs of $\mathcal{POS}$. If $|\mathcal{POS}| > 1$, we proceed as follows:

Procedure amalgamating POS. For every i, $1 \leq i \leq o$, amalgamate each graph of the pair Q_i with a graph of a pair Q_j, where $j \neq i$.

Similar constraints are required in amalgamating PESs to avoid transforming them into SPs. If $|\mathcal{PES}| > 1$, we proceed as follows:

Procedure amalgamating PES. For every i, $1 \leq i \leq e$, amalgamate each graph of P_i with a graph in P_j, where $j \neq i$. Moreover, if $\mathcal{PES}$ has at least one P_i with two graphs and if a last nonamalgamated graph G_P remains, then G_P should belong to a P_i of size 2. Suppose G_1 and G_2 are amalgamated, O_1 and O'_1 are the two vertices of $G_1 \cap O$, and O_2 and O'_2 are the two vertices of $G_2 \cap O$. Then set $\overline{O_1} = O_2$ and $\overline{O'_1} = O'_2$.

Note that, in the case of the PESs, we amalgamate even size (completable) natural graphs. The consequence is that *dedouble*, applied to such supernatural graphs, generates a completed graph that is no longer maximal. This gives rise to additional difficulties.

After amalgamating the graphs of $\mathcal{PES} \cup \mathcal{POS}$ by the procedures described above, there remain some nonamalgamated graphs. This gives rise to eight possible configurations. For some of them, additional graphs are amalgamated.

C1. There remain no nonamalgamated graphs.

C2. There remains one Q_i in $\mathcal{POS}$. This happens when $\mathcal{POS}$ contains a single POS.

C3. There remains one P_i of two graphs in $\mathcal{PES}$. This happens when $\mathcal{PES}$ contains a single PES.

C4. There remains one graph in $\mathcal{PES}$, and it belongs to a P_i of size 2.

C5. There remains one graph in $\mathcal{PES}$, and it belongs to a P_i of size 1. This happens when all P_is are of size 1 and e is even.

C6. There remains one $Q_i = (G_1, G_2)$ in $\mathcal{POS}$ and one G_3 in $\mathcal{PES}$ belonging to a P_i of size 1. Then we amalgamate the three graphs G_1, G_2, and G_3 if that does not create an even-isolation. If O_1 and O'_1 are the vertices of $G_1 \cap O$, O_2 and O'_2 are the vertices of $G_2 \cap O$, and O_3 and O'_3 are the vertices of $G_3 \cap O$, then we set $\overline{O_1} = O_3$, $\overline{O'_1} = O'_2$, and $\overline{O_2} = O'_3$.

C6' will denote the configuration that would give rise to an even-isolation. In this case, the graphs are not amalgamated.

C7. There remains one Q_i in $\mathcal{POS}$ and one graph in $\mathcal{PES}$ belonging to a P_i of size 2.

C8. There remains one $Q_i = (G_1, G_2)$ in $\mathcal{POS}$ and one $P_i = (G_3, G_4)$ in $\mathcal{PES}$. Then we amalgamate G_1 and G_2 and one of the two graphs of P_i if that does not create an even-isolation. Counterpart elements are set similarly to C6.

C8' will denote the configuration that would give rise to an even-isolation. In that case, the graphs are not amalgamated.

A local SP that is either real or potential, but not solved by the amalgamating procedure described above, is called a *final SP*.

In the remainder of this section, $\mathcal{SG}$ will designate the set of completable graphs obtained by the procedure described above for the graphs in $\mathcal{POS} \cup \mathcal{PES}$, and by the usual way (section 5) for the other natural graphs.

Notation 4. Consider the following parameters:

- $\mathbf{s(G)}$ is the number of real minSPs of G;
- $\mathbf{sp(G)}$ is the number of graphs obtained by amalgamating $\mathcal{PES}$ graphs;
- $\psi(\mathbf{G}) = 1$ if configuration C6 or C8 is encountered, and $\psi(\mathbf{G}) = 0$ otherwise;
- $\mathbf{sr(G)} = 0$ if one of the configurations C1, C4, C6, or C8 is encountered; $sr(G) = 1$ for C2, C3, C5, or C7; $sr(G) = 2$ for C6' or C8'. $sr(G)$ is the number of potential SPs that become final SPs.
- $\mathbf{f(G)} = 2$ if the set of final SPs represents an even-isolation; $f(G) = 1$ if the number of final SPs is odd; $f(G) = 0$ otherwise.

Recall that $c(G)$ is the number of cycles of a maximal completed graph of $\mathcal{G}(\mathbf{V}, A)$ (Theorem 7.6).

THEOREM 9.3. *Let $\mathcal{G}_\Gamma(\mathbf{V}, A, \Gamma)$ be the completed graph produced by* dedouble. *Let H be the resulting duplicated genome. Then $c(\mathcal{G}_\Gamma) = c(G) - sp(G) - \psi(G)$, $s(\mathcal{G}_\Gamma) = s(G) + sr(G)$, and*

$$T(G, H) = |A| - c(G) + sp(G) + \psi(G) + s(G) + sr(G) + f(G).$$

The minimal number of translocations required to transform G into a duplicated genome is $T(G) = T(G, H)$.

Proof. According to Corollary 8.3 and Lemma 8.4, if G does not contain any PESs, then *dedouble* produces a maximal completed graph $\mathcal{G}_\Gamma(\mathbf{V}, A, \Gamma)$ with $c(G)$ cycles.

Suppose now that G contains local SPs. Let $\mathcal{G}_3$ be a graph of $\mathcal{SG}$ obtained by amalgamating two natural graphs of $\mathcal{PES}$: $\mathcal{G}_1$ of size n_1 and $\mathcal{G}_2$ of size n_2. Given that this graph has as many left edges as right edges, a proof similar to that of Lemma 6.3 shows that the maximal number of cycles of a completed graph of $\mathcal{G}_3$ is $\frac{n_1+n_2}{2} + 1$, and *dedouble* produces such a maximal completed graph. If we apply *dedouble* to each of the two graphs $\mathcal{G}_1$ and $\mathcal{G}_2$, we obtain two completed graphs with a total of $\frac{n_1}{2} + 1 + \frac{n_2}{2} + 1$ cycles, that is, one more cycle than for $\mathcal{G}_3$. Thus, if we apply *dedouble* to the graphs of $\mathcal{SG}$ obtained by amalgamating graphs in $\mathcal{PES}$, we obtain $sp(G)$ fewer cycles than if we apply the algorithm to each graph of $\mathcal{PES}$. Moreover, one fewer cycle is also obtained by amalgamating one graph pair of $\mathcal{POS}$ and one graph of $\mathcal{PES}$. As these are the only modifications to the original procedure of graph amalgamating that changes the number of cycles, $c(\mathcal{G}_\Gamma) = c(G) - sp(G) - \psi(G)$.

Moreover, also according to Corollary 8.3 and Lemma 8.4, if G does not contain any PESs, then *dedouble* produces a maximal completed graph $\mathcal{G}_\Gamma(\mathbf{V}, A, \Gamma)$ with $s(\mathcal{G}_\Gamma)$ SPs corresponding to the $s(G)$ local SPs of G and to the only existing POS, if any. Moreover, the $sr(G)$ potential SPs not amalgamated are the only potential SPs that become real SPs. Therefore, $s(\mathcal{G}) = s(G) - sr(G)$.

We deduce that

$$T(G,H) = |A| - c(\mathcal{G}_\Gamma) + s(\mathcal{G}_\Gamma) + f(\mathcal{G}_\Gamma) = |A| - c(G) + sp(G) + \psi(G) + s(G) + sr(G) + f(G).$$

Suppose $T(G,H) > T(G)$. Then there is a completed graph $\mathcal{G}_{\Gamma'}(\mathbf{V}, A, \Gamma')$ containing $c(\mathcal{G}_{\Gamma'})$ cycles, $s(\mathcal{G}_{\Gamma'})$ SPs, and a value of $f(\mathcal{G}_{\Gamma'})$ such that (1) $c(\mathcal{G}_{\Gamma'}) - s(\mathcal{G}_{\Gamma'}) - f(\mathcal{G}_{\Gamma'}) > c(\mathcal{G}_\Gamma) - s(\mathcal{G}_\Gamma) - f(\mathcal{G}_\Gamma)$, i.e., $c(\mathcal{G}_{\Gamma'}) - c(\mathcal{G}_\Gamma) > (s(\mathcal{G}_{\Gamma'}) - s(\mathcal{G}_\Gamma)) + (f(\mathcal{G}_{\Gamma'}) - f(\mathcal{G}_\Gamma))$.

First, suppose that $\mathcal{G}_{\Gamma'}(\mathbf{V}, A, \Gamma')$ contains p fewer SPs than $\mathcal{G}_\Gamma(\mathbf{V}, A, \Gamma)$. Suppose first that $p = 1$ and that this SP is a real local SP of G. Then, by Lemma 8.4, $\mathcal{G}_{\Gamma'}(\mathbf{V}, A, \Gamma')$ is a completed graph that is not maximal and contains at most $c(\mathcal{G}_\Gamma) - 2$ cycles. More generally, a construction that removes p real local SPs of G gives rise to a completed graph with at most $c - 2p$ cycles. Suppose now that $\mathcal{G}_{\Gamma'}(\mathbf{V}, A, \Gamma')$ has one less SP than $\mathcal{G}_\Gamma(\mathbf{V}, A, \Gamma)$, but this SP is not a real local SP of G. That means that it corresponds to a potential SP transformed into a final SP. This occurs in configurations C2, C3, C5, C6', C7, and C8. In all cases, it is easy to show that at least two cycles would necessarily be removed if we remove such an SP. Therefore, (2) $c(\mathcal{G}_{\Gamma'}) \leq c(\mathcal{G}_\Gamma) - 2(s(\mathcal{G}_{\Gamma'}) - s(\mathcal{G}_{\Gamma'}))$.

We deduce from (1) and (2) that $s(\mathcal{G}_\Gamma) - s(\mathcal{G}_{\Gamma'}) < f(\mathcal{G}_\Gamma) - f(\mathcal{G}_{\Gamma'})$.

As $s(\mathcal{G}_\Gamma) - s(\mathcal{G}_{\Gamma'}) \geq 0$ and $f(\mathcal{G}_\Gamma) - f(\mathcal{G}_{\Gamma'}) \leq 2$, we should have $s(\mathcal{G}_\Gamma) - s(\mathcal{G}_{\Gamma'}) = 1$ and $f(\mathcal{G}_\Gamma) - f(\mathcal{G}_{\Gamma'}) = 2$. We can see, from the definition of f, that this configuration is impossible.

Suppose now that $\mathcal{G}_{\Gamma'}(\mathbf{V}, A, \Gamma')$ contains p more SPs than $\mathcal{G}_\Gamma(\mathbf{V}, A, \Gamma)$. If these SPs that are in $\mathcal{G}_{\Gamma'}(\mathbf{V}, A, \Gamma')$ but not in $\mathcal{G}_\Gamma(\mathbf{V}, A, \Gamma)$ do not correspond to potential local SPs of G, then, from Lemma 8.2 and the fact that $f(\mathcal{G}_\Gamma) \leq 2$, the value of $-c(\mathcal{G}_\Gamma) + s(\mathcal{G}_\Gamma + f(\mathcal{G}_\Gamma)$ is not changed if we remove these SPs. Thus these SPs correspond necessarily to potential local SPs that are transformed into final SPs in $\mathcal{G}_{\Gamma'}(\mathbf{V}, A, \Gamma')$ but not in $\mathcal{G}_\Gamma(\mathbf{V}, A, \Gamma)$. Necessarily, $p \geq 2$.

Suppose first $p = 2$. If these two SPs correspond to

- two POS, then amalgamating these two SPs gives rise to two fewer SPs and to the same number of cycles,
- two PES, then amalgamating these two SPs gives rise to two fewer SPs and one less cycle,
- one POS and one PES, then amalgamating these SPs gives rise to two fewer SPs and one less cycle.

More generally, a graph containing $2p$ more SPs contains at most p more cycles than $\mathcal{G}(\mathbf{V}, A, \Gamma)$. Therefore, (3) $c(\mathcal{G}_{\Gamma'}) - c(\mathcal{G}_\Gamma) \leq \frac{s(\mathcal{G}_{\Gamma'}) - s(\mathcal{G}_\Gamma)}{2}$.

We deduce from (1) and (3) that $s(\mathcal{G}_{\Gamma'}) - s(\mathcal{G}_\Gamma) < 2(f(\mathcal{G}_\Gamma) - f(\mathcal{G}_{\Gamma'}))$.

As $s(\mathcal{G}_{\Gamma'}) - s(\mathcal{G}_\Gamma) \geq 2$ and $f(\mathcal{G}_\Gamma) - f(\mathcal{G}_{\Gamma'}) \leq 2$, $s(\mathcal{G}_{\Gamma'}) - s(\mathcal{G}_\Gamma) = 2$ and $f(\mathcal{G}_\Gamma) - f(\mathcal{G}_{\Gamma'}) = 2$. That means that amalgamating the two potential SPs gives rise to an even-isolation, which is in contradiction with the properties of the amalgamating procedure. □

10. Genome halving with translocations and reversals.

10.1. The HP method. Given two genomes H_1 and H_2 with the same number of chromosomes, HP [18] determined the minimal number of reversals and translocations $RT(H_1, H_2)$ required to transform H_1 into H_2. Formula **HP1** (section 3) is a general description of the result. A more precise description requires a deeper consideration of the problem.

We will only sketch the HP procedure, which is rather complex. The first step in the comparison of two multichromosomal genomes through translocations and reversals is to reduce it to a problem of comparing two single chromosome genomes through reversals only. These latter genomes are constructed essentially by concatenating the individual chromosomes in the original genomes end-to-end in an arbitrary order. Additional dummy genes, called *caps*, must be appropriately inserted at the ends of the original chromosomes of both genomes. A translocation in an original genome becomes a reversal in the new one.

More precisely, let $H = C_1, \dots, C_N$ be a genome of N chromosomes written in a particular order. An H *concatenate* is a genome $\tilde{H}$ with a single chromosome: $\tilde{H} = (s_1C_1)\cdots(s_NC_N)$, where each s_i, $1 \leq i \leq N$, is in $\{-1, 1\}$. The *identity concatenate* of H is the H concatenate satisfying $s = (s_1, \dots, s_N) = (1, \dots, 1)$.

Let $O = \{O_0, \dots, O_{2N-1}\}$ be a set of caps and $\hat{H}_1$ the genome obtained by adding one cap at each end of each chromosome of H_1. Any sequence of reversals/translocations transforming H_1 into H_2 induces a sequence of reversals transforming $\hat{H}_1$ into a genome $\hat{H}_2$, where $\hat{H}_2$ is a particular capping of H_2. We can prove that $RT(H_1, H_2) = \min_{\hat{H}_2 \in \hat{\mathcal{H}}} RT(\hat{H}_1, \hat{H}_2)$, where $\hat{\mathcal{H}}$ is the set of all possible cappings of H_2.

Let $\tilde{H}_1$ be the identity concatenate of $\hat{H}_1$. Let $\mathcal{G}_s(\mathbf{V}, A, \Gamma_s)$ be the graph defined as follows: $V = \{x^{s \in \{t,h\}},\ x \in \mathcal{B}\}$ and $\mathbf{V} = V \cup O$; A is the set of black edges connecting adjacent vertices in H_1 other than (x^t, x^h) for the same x; Γ_s is the set of gray edges connecting adjacent vertices in H_2. Denote by V_e the set of vertices of V located at the ends of H_2 chromosomes. Note that vertices of $V_e \cup O$ are not connected by gray edges in $\mathcal{G}_s(\mathbf{V}, A, \Gamma_s)$. $\mathcal{G}_s(\mathbf{V}, A, \Gamma_s)$ is called the *semicompleted graph* associated to H_1 and H_2. It is a collection of cycles and paths. Paths are of three kinds: those ending with a vertex in O and another in V, called *good paths*, those ending with two vertices in O, called *bad paths*, and those ending with two vertices in V. Denote, respectively, $\mathcal{OV}$, $\mathcal{OO}$, and $\mathcal{VV}$ as these three path sets. We have $|\mathcal{OO}| = |\mathcal{VV}|$.

A gray edge in a cycle or a path of size > 1 is *oriented* if it links the vertices at the left ends of two black edges or at the right ends of two black edges, while an *unoriented gray edge* links two different sides of two black edges. A cycle or a path is *good* if it contains at least one oriented gray edge, and *bad* otherwise. A component is *good* if it has at least one good cycle or path and thus at least one oriented gray edge, and *bad* otherwise.

HP showed that a good component can be *solved*, i.e., transformed to a set of cycles (and paths) of size 1, by a series of proper reversals (reversals increasing the number of cycles; see section 3). However, bad components often require bad reversals. The set of bad components is subdivided into subsets, depending on the difficulty of solving them (i.e., transforming them into good components). This subdivision is explained below.

An edge of Γ_s is *intrachromosomal* if it connects two vertices both belonging to the same chromosome of H_1 and *interchromosomal* otherwise. A component of $\mathcal{G}_s(\mathbf{V}, A, \Gamma_s)$ is intrachromosomal if it contains only intrachromosomal edges, and interchromosomal otherwise. We say that the component U *separates* two components U' and U'' if any edge we tried to draw from a vertex of U' to one of U'' would cut a gray edge of U. A *knot* is an intrachromosomal bad component which does not separate any pair of bad components. Now, a *real knot* is a knot that contains only cycles (no paths), and a *semiknot* is a knot containing at least one path in $\mathcal{OV}$ and no path in $\mathcal{OO} \cup \mathcal{VV}$.

The underlying idea is that a bad component U that separates two bad components U' and U'' is automatically solved by solving U' and U'' and thus may just as well be considered to be a good one. On the other hand, a real knot requires bad reversals to be solved, while a semiknot can be transformed into a good component if paths are closed appropriately.

HP proved that the problem of sorting by reversals/translocations is reduced to a problem of finding an appropriate capping of H_2, that is, finding appropriate connections between vertices of V_e and vertices of O. Finally, they proved that the minimal number of reversals/translocations required to transform H_1 into H_2 is

$$\textbf{HP3}: \quad RT(H_1,H_2) = b(\mathcal{G}_s) - cp(\mathcal{G}_s) + bp(\mathcal{G}_s) + rr(\mathcal{G}_s) + \left\lceil \frac{s(\mathcal{G}_s) - gr(\mathcal{G}_s) + fr(\mathcal{G}_s)}{2} \right\rceil,$$

where $b(\mathcal{G}_s) = |A|$; $cp(\mathcal{G}_s)$ is the number of cycles and paths of $\mathcal{G}_s(\mathbf{V}, A, \Gamma_s)$; $bp(\mathcal{G}_s)$ is the number of bad paths; $rr(\mathcal{G}_s)$ is the number of real knots obtained after closing paths of $\mathcal{OV}$ that are not included in semiknots; $s(\mathcal{G}_s)$ is the number of semiknots; and $gr(\mathcal{G}_s)$ and $fr(\mathcal{G}_s)$ take values 0 or 1, depending on the set of real knots and semiknots.

Returning to the problem of genome halving, we represent the genome G as H_1, i.e., by adding caps at the ends of the chromosomes, and by concatenating the resulting chromosomes. The partial graph $\mathcal{G}(\mathbf{V}, A)$ associated to G is thus represented on a single line instead of $2N$ lines. Algorithm *dedouble* can be applied to such a partial graph as well. The goal is to construct a semicompleted graph $\mathcal{G}_s(\mathbf{V}, A, \Gamma_s)$ that minimizes formula **HP3**. We first construct a *maximal semicompleted graph* that maximizes $cp(\mathcal{G}_s) - bp(\mathcal{G}_s)$.

10.2. Constructing a maximal semicompleted graph. Denote by $c(G)$ the number of cycles of a maximal completed graph of $\mathcal{G}(\mathbf{V}, A)$ obtained by *dedouble*. For any semicompleted graph $\mathcal{G}_s(\mathbf{V}, A, \Gamma_s)$, denote by $c(\mathcal{G}_s)$ its number of cycles and by $p(\mathcal{G}_s)$ its total number of paths. Denote also $cc(\mathcal{G}_s) = cp(\mathcal{G}_s) - bp(\mathcal{G}_s)$.

LEMMA 10.1. *Let $\mathcal{G}_s(\mathbf{V}, A, \Gamma_s)$ be a maximal semicompleted graph of $\mathcal{G}(\mathbf{V}, A)$. Then $cc(\mathcal{G}_s) \leq c(G)$.*

Proof. As mentioned above, we have $|\mathcal{OO}| = |\mathcal{VV}|$. Let $\mathcal{G}_\Gamma(\mathbf{V}, A, \Gamma)$ be the graph obtained by closing good paths and by constructing cycles with remaining paths, each of these paths obtained by connecting a path of $\mathcal{OO}$ with a path of $\mathcal{VV}$. Such a graph is clearly a completed graph of $\mathcal{G}(\mathbf{V}, A)$ with $c(\mathcal{G}_s) - p(\mathcal{G}_s)$ cycles. As $c(G)$ is the number of cycles of a maximal completed graph, we have $c(\mathcal{G}_s) - p(\mathcal{G}_s) \leq c(G)$. □

We now construct a semicompleted graph $\mathcal{G}_s(\mathbf{V}, A, \Gamma_s)$ satisfying $cc(\mathcal{G}_s) = c(G)$. From Lemma 10.1, this graph must be maximal.

THEOREM 10.2. *Let $\mathcal{G}_\Gamma(\mathbf{V}, A, \Gamma)$ be the maximal completed graph obtained by applying* dedouble *to $\mathcal{G}(\mathbf{V}, A)$. Let Γ_s be the set of gray edges obtained from Γ by removing all edges adjacent to at least one vertex in O, and consider the semicompleted graph $\mathcal{G}_s(\mathbf{V}, A, \Gamma_s)$. Then $\mathcal{G}_s(\mathbf{V}, A, \Gamma_s)$ is a maximal semicompleted graph of $\mathcal{G}(\mathbf{V}, A)$. Moreover, $cp(\Gamma_s) = c(G)$.*

Proof. According to the construction of natural and supernatural graphs, each supernatural graph contains 0, 2, or 4 vertices in O. Moreover, it is easy to see that each cycle of a maximal completed graph contains at most two vertices in O as if this is not satisfied, then the cycle could be subdivided into at least two cycles, and thus the completed graph could not be maximal.

Let $\mathcal{G}'(\mathbf{V}', A')$ be a supernatural graph. Let $\mathcal{G}_{\Gamma'}(\mathbf{V}', A', \Gamma')$ be this supernatural graph completed by *dedouble*, and let $\mathcal{G}'_s(\mathbf{V}', A', \Gamma'_s)$ be the semicompleted supernatural graph obtained by removing from Γ' edges with at least one end in O. Let $c(\Gamma')$ be the number of cycles of $\mathcal{G}_{\Gamma'}(\mathbf{V}', A', \Gamma')$.

- Suppose that $\mathbf{V}'$ does not contain any vertex in O. In this case, no edge is removed from Γ' to form $\mathcal{G}'_s(\mathbf{V}', A', \Gamma'_s)$, and thus $c(\mathcal{G}'_s) - p(\mathcal{G}'_s) = c(\Gamma')$.
- Suppose that $\mathbf{V}'$ contains two vertices in O. Suppose that these two vertices are in two different cycles of $\mathcal{G}_{\Gamma'}(\mathbf{V}', A', \Gamma')$. Then removing the two gray edges connecting these two vertices transforms the two corresponding cycles into two good paths. Thus $c(\mathcal{G}'_s) - p(\mathcal{G}'_s) = c(\Gamma')$.

  Suppose now that both vertices are in the same cycle. Then removing the two gray edges connecting these two vertices transforms the cycle into two paths, and at most one of them is bad. (In this case, the second path is in $\mathcal{VV}$.) Thus $c(\mathcal{G}'_s) - p(\mathcal{G}'_s) \geq (c(\Gamma') - 1) + 2 - 1 = c(\Gamma')$.
- Suppose that $\mathbf{V}'$ contains four vertices in O. If these vertices are in four or three different cycles, then we prove by an argument similar to the previous case that $c(\mathcal{G}'_s) - p(\mathcal{G}'_s) \geq c(\Gamma')$.

  Otherwise, if these vertices are in two cycles, then each of these cycles contains two of the four vertices. In that case, removing the four gray edges adjacent to these four vertices transforms the two cycles into four paths, and at most two of them are bad. Thus $c(\mathcal{G}'_s) - p(\mathcal{G}'_s) \geq (c(\Gamma') - 2) + 4 - 2 = c(\Gamma')$.

In all cases, $c(\mathcal{G}'_s) - p(\mathcal{G}'_s) \geq c(\Gamma')$. We deduce that $c(\mathcal{G}_s) - p(\mathcal{G}_s) \geq c(G)$. As $\mathcal{G}_\Gamma(\mathbf{V}, A, \Gamma)$ is a maximal completed graph, from Lemma 10.1, $c(\mathcal{G}_s) - p(\mathcal{G}_s) = c(G)$ and $\mathcal{G}_s(\mathbf{V}, A, \Gamma_s)$ is a maximal semicompleted graph. □

We call *semidedouble* the algorithm obtained by incorporating, at the end of *dedouble*, the procedure that removes the gray edges adjacent to at least one vertex of O. From Theorem 10.2, *semidedouble* constructs a maximal semicompleted graph.

Remark 3. Let C be a cycle of a completed graph $\mathcal{G}(\mathbf{V}, A, \Gamma)$ containing two vertices O_1 and O_2 in O. Let C_1 and C_2 be the two paths obtained by removing the two gray edges adjacent to O_1 and O_2. Then one of the following situations is satisfied: 1. C_1 and C_2 are two paths in $\mathcal{OV}$; or 2. C_1 is a bad path (in $\mathcal{OO}$) and C_2 is in $\mathcal{VV}$.

The first situation occurs when O_1 and O_2 are separated by an odd number of vertices in C (to the right or to the left), and the second situation occurs when O_1 and O_2 are separated by an even number of vertices.

10.3. Knots. We now turn our attention to minimizing, in formula **HP3**, the expression $rr(\mathcal{G}_s) + \lceil \frac{s(\mathcal{G}_s) - gr(\mathcal{G}_s) + fr(\mathcal{G}_s)}{2} \rceil$. Denote by $RT(G)$ the minimal number of reversals/translocations required to transform G into a duplicated genome. We deduce the following corollary from Theorem 7.7, Corollary 8.3, and formula **HP3**.

Corollary 10.3. *If G does not contain any local SP, then $RT(G) = T(G) = |A| - c(G) = \frac{|A|}{2} - \gamma(G)$.*

Suppose now that G contains local SPs. Let S be a local SP, and let $\Pi = \{\pi_1, \ldots, \pi_p\}$ be the set of components of the maximal semicompleted graph obtained by *semidedouble*, containing the vertices of S. In order to consider the components which may form knots, we introduce another definition. Let $\mathcal{U} = \{u_1, \ldots, u_p\}$ be a subset of $\mathcal{B}$, and $\overline{\mathcal{U}} = \{\overline{u_1}, \ldots, \overline{u_p}\}$. We say that $\mathcal{U}$ is *unoriented* if genes u_i and $\overline{u_i}$ have either the same sign in G or opposite signs for all i. Otherwise, $\mathcal{U}$ is *oriented.* Let $\pi_i \in \Pi$, and let $\mathbf{V_i}$ be its vertex set. $\mathbf{V_i}$ is oriented if and only if the set of genes corresponding to the vertices in $\mathbf{V_i}$ is oriented.

Lemma 10.4. *π_i is a good component if and only if $\mathbf{V_i}$ is oriented.*

Proof. π_i is good if and only if π_i contains at least one cycle with at least one

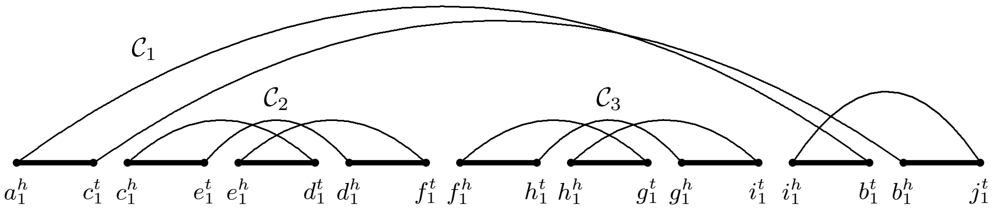

FIG. 10.1. *Inner SPs corresponding to* S_1.

oriented gray edge. Suppose that $\mathbf{V_i}$ is unoriented. We can suppose, without loss of generality, that all corresponding genes are signed +. All $\mathcal{C}$ cycles of π_i are such that left vertices are of form x^h and right edges are of form x^t (Definition 6.1). Moreover, there is no supernatural graph of $\mathcal{SO}$ obtained by amalgamating natural graphs containing vertices in $\mathbf{V_i}$. Therefore, a gray edge necessarily connects a left vertex to a right one in $\mathcal{C}$ cycles, and thus a vertex of form x^h to one of form x^t. From the definition of gray edge orientation, all gray edges of π_i are unoriented, and thus π_i is bad.

Conversely, if $\mathbf{V_i}$ is oriented, then we can assume, without loss of generality, that there exist two genes a and b such that a and b are adjacent and are both signed positively in S, but a and b do not have the same sign in $\overline{S}$. Algorithm *dedouble* constructs the gray edges (a^h, b^t) and $(\overline{a}^h, \overline{b}^t)$. As a and b do not have the same sign in $\overline{S}$, $\overline{a}^h$ and $\overline{b}^t$ are either both left ends or both right ends of black edges. Therefore, the gray edge $(\overline{a}^h, \overline{b}^t)$ is oriented. □

We say that a *real local SP is oriented* if the set of vertices in its component is oriented and is unoriented otherwise. Knots produced by *semidedouble* then correspond to the real minimal unoriented SPs, which we call *real minimal SPs*, and to at most one other SP, the maximal one, defined as follows.

DEFINITION 10.5. *Let* $S = x_1 \cdots x_n$ *be a local SP. The* outer SP *of S is the largest SP* S_e *contained in S satisfying the following three conditions:*

1. *The component* π_e *of* S_e *is bad;*
2. S_e *is not minimal, and the interval of* S_e *contains all the real minimal SPs of S;*
3. S_e *does not separate two real minimal SPs.*

A local SP S is maximal *if all the real minimal SPs of G are inner SPs of S and if there exists an outer SP of S.*

Example 6. Suppose that the genome G contains the local SP $S_1 = a_1\ c_1\ e_1\ d_1\ f_1\ h_1\ g_1\ i_1\ b_1\ j_1$, with complement $\overline{S_1} = a_2\ b_2\ c_2\ d_2\ e_2\ f_2\ g_2\ h_2\ i_2\ j_2$. The components of S_1 and the inner SPs of S_1 are depicted in Figure 10.1. The two components $\mathcal{C}_2$ and $\mathcal{C}_3$ correspond to components of the two minimal SPs of S. $\mathcal{C}_1$ is the component of S_1. It is bad and does not separate two minimal SPs. S_1 is thus an outer SP.

A *bad real SP* is a real SP which is either minimal or maximal. We denote by $\mathbf{brs(G)}$ the number of bad SPs of G.

As for semiknots, they are associated to potential SPs. To minimize them, we must minimize the number of potential SPs that become bad SPs of the final graph.

We already saw in section 9 how to solve POSs, provided that at least two of them exist. We can therefore assume that at most one POS exists. Suppose that one such POS S exists. In that case, we amalgamate the two odd size natural graphs G_1 and G_2 associated to S. Suppose that $G_1 = O_1 x_1 \cdots x_n O_1'$ and $G_2 = O_2 y_1 \cdots s\overline{x_1} \cdots y_n O_2'$, where s is the sign of $\overline{x_1}$. If $s = +$, then we set $\overline{O_1} = O'_2$ and $\overline{O'_1} = O_2$. Otherwise ($s = -$), we set $\overline{O_1} = O_2$ and $\overline{O'_1} = O'_2$. With this construction, we ensure that the set of vertices $\{O_1^h, x_1, \ldots, x_n, O_1^{\prime t}\}$ is oriented.

Consider now the PESs. For every graph associated to a PES and containing two vertices O_1 and O_2 of O, *semidedouble* sets $\overline{O_1} = O_2$. We say that a PES S is unoriented if the set of vertices of S is unoriented and oriented otherwise. As oriented PESs do not give rise to any problem, we consider only, in the ensuing discussion, an unoriented PES S. S is a *minimal PES* if S is minimal. Moreover, an *outer PES* of S is defined in a similar way as for a real SP (Definition 10.5) by replacing "real minimal SPs" by "real minimal SPs and minimal PESs." We similarly define a *maximal PES* and a *bad PES*.

We denote by *BPES* a bad PES, b is the number of BPESs, and $\mathcal{BPES} = \{P_1, \ldots, P_b\}$ with, for every i, P_i as the set of graphs associated to the BPES i.

To minimize the number of semiknots, graphs of *BPES* are amalgamated with procedure *amalgamating-PES* described in section 9. Three configurations can arise after applying the procedure:

R1. There remains no nonamalgamated graph.

R2. There remains only one nonamalgamated graph, and it belongs to a P_i of size 2.

R3. There remains one $P_i \in \mathcal{BPES}$ with one or two nonamalgamated graphs. This happens if only one BPES exists or if b is odd and $\mathcal{BPES}$ contains only sets of size 1.

In the remainder of this section, $\mathcal{SG}$ is the set of completable graphs obtained by amalgamating the two graphs of a POS, if any, as described above, by using procedure *amalgamating-PES* for the graphs of $\mathcal{BPES}$, and by the usual way for the other natural graphs.

LEMMA 10.6. *Let $\mathcal{G}_s(\mathbf{V}, A, \Gamma_s)$ be the semicompleted graph obtained by* semidedouble*. Then $\mathcal{G}_s(\mathbf{V}, A, \Gamma_s)$ is a maximal semicompleted graph.*

Proof. Suppose that G does not contain any BPESs. The amalgamating procedure is then identical to that of section 5. Thus, from Theorem 10.2, *dedouble* constructs a maximal completed graph.

Suppose now that G contains BPESs. The only graphs of $\mathcal{SG}$ not corresponding to those of section 5 are those obtained by amalgamating graphs of $\mathcal{BPES}$. Let $\mathcal{G}_3(\mathbf{V}_3, A_3)$ be a graph of size n_3 obtained by amalgamating the two graphs $\mathcal{G}_1 = (\mathbf{V}_1, A_1)$ of size n_1 and $\mathcal{G}_2 = (\mathbf{V}_2, A_2)$ of size n_2 of $\mathcal{BPES}$. By arguments similar to those used in the proof of Theorem 9.3, we can see that *dedouble* gives rise to one less cycle when it is applied to a set of graphs containing $\mathcal{G}_3$, instead of a set of graphs containing the two supernatural graphs $\mathcal{G}_1$ and $\mathcal{G}_2$. More precisely, let c_{max} be the maximal number of cycles containing edges of $A_1 \cup A_2$ obtained when the two graphs $\mathcal{G}_1$, $\mathcal{G}_2$ are considered and c the number of cycles containing edges of A_3 obtained when $\mathcal{G}_3$ is considered. Then $c = c_{max} - 1$.

Let Γ'_3 be the set of gray edges linking the vertices of $\mathbf{V}_3$ in a completed graph obtained by applying *dedouble* to a set of graphs containing $\mathcal{G}_3(\mathbf{V}_3, A_3)$. $\mathbf{V}_3$ has four vertices in O, denoted by the set O'. *Dedouble* constructs two cycles of size 1, each containing one of the vertices of O', and one cycle C of size > 1 containing the two remaining vertices of O'. Let $\mathcal{G}_{3,s}(\mathbf{V}_3, A_3, \Gamma_3)$ be the semicompleted graph of $\mathcal{G}_3(\mathbf{V}_3, A_3)$ obtained by *semidedouble*, that is, by removing from Γ'_3 the edges adjacent to vertices of O. From Remark 3, vertices of O are either all left vertices or all right vertices. Therefore, removing from Γ'_3 the edges adjacent to the vertices of O

transforms C into two good paths. The number of bad paths of $\mathcal{G}_{3,s}$ is then $bp_3 = 0$. Moreover, $cc(\mathcal{G}_{3,s}) = c_{max}$.

We deduce that $cc(\mathcal{G}_s) = c(G)$. $\mathcal{G}_s(\mathbf{V}, A, \Gamma_s)$ is thus a maximal semicompleted graph. □

A local SP that is either real or a BPES not solved by the procedure *amalgamating-PES* is called a *final SP*.

Notation 5. Consider the following parameters:

- $\mathbf{s(G)}$ is the number of BPESs that become semiknots. $s(G) = 0$ if configurations R1, R2 are encountered, and $s(G) = 1$ otherwise.
- $\mathbf{brs(G)}$ is the number of bad real SPs of G.
- $\mathbf{fr(G)}$ and $\mathbf{gr(G)}$ are defined like $fr(\mathcal{G}_s)$ and $gr(\mathcal{G}_s)$ [18]. They depend on the set of real knots and semiknots determined by the set of final SPs of G.

We require one more lemma.

LEMMA 10.7. *Suppose that G contains an unoriented local SP S. Let π be the component of S. Then any maximal completed graph must contain an unoriented component made up of the vertices of π.*

Proof. Suppose that G contains an unoriented real SP S. From Lemma 8.4, any maximal completed graph $\mathcal{G}(\mathbf{V}, A, \Gamma)$ contains an SP formed by S vertices. As S does not contain any vertex in O, any maximal semicompleted graph also contains an SP formed by S vertices. On the other hand, all supernatural graphs containing vertices of $S \cup \overline{S}$ are in $\mathcal{SE}$, and the corresponding completed supernatural graphs (in a maximal completed graph) give rise to at least one component. We want to show that each of these components contains exclusively unoriented gray edges.

As S is unoriented, we can assume that all genes corresponding to S vertices are signed positively and that all left vertices of S are of form x^h and all right vertices of form x^t. Let $\mathcal{G}_{sn}(\mathbf{V_{sn}}, A_{sn})$ be a supernatural graph containing vertices of S. Suppose that the corresponding completed supernatural graph $\mathcal{G}_{sn}(\mathbf{V_{sn}}, A_{sn}, \Gamma_{sn})$ contains an oriented edge. Such an edge necessarily links two left vertices or two right vertices. Arguments similar to those used in the proof of Lemma 6.3 show that $\mathcal{G}_{sn}(\mathbf{V_{sn}}, A_{sn}, \Gamma_{sn})$ contains at least one cycle less than a completed supernatural graph corresponding to a maximal completed graph. □

THEOREM 10.8. *Let $\mathcal{G}_s(\mathbf{V}, A, \Gamma_s)$ be the semicompleted graph produced by* semidedouble. *Let H be the resulting duplicated genome. Then $cp(\mathcal{G}_s) - bp(\mathcal{G}_s) = c(G) = \frac{1}{2}|A| + \gamma(G)$, $rr(\mathcal{G}_s) = brs(G)$, $s(\mathcal{G}_s) = s(G)$, $fr(\mathcal{G}_s) = fr(G)$, $gr(\mathcal{G}_s) = gr(G)$, and*

$$RT(G, H) = \frac{1}{2}|A| - \gamma(G) + brs(G) + \left\lceil \frac{s(G) - gr(G) + fr(G)}{2} \right\rceil .$$

Moreover, $RT(G, H) = RT(G)$.

Proof. To simplify the notation, denote by cc, cp, bp, rr, s, gr, and fr the parameters corresponding to the graph $\mathcal{G}_s$.

From Lemma 10.6, $\mathcal{G}_s(\mathbf{V}, A, \Gamma_s)$ is a maximal semicompleted graph, and $cp - bp = c(G)$. Now, we know that real knots correspond to bad real SPs, plus at most one maximal SP. Thus $rr = brs(G)$. As for semiknots, they correspond to bad components containing at least one good path and for which the corresponding interval does not contain any path in $\mathcal{OO} \cup \mathcal{VV}$. As POSs do not give rise to bad components, the only remaining nonamalgamated PES, if any, is the only SP giving rise to a bad component with vertices in O. As these vertices (in O) are either all left vertices or all right vertices, from Remark 3, removing gray edges adjacent to these vertices gives rise to only good cycles. One semiknot is then due to this nonamalgamated PES. Therefore, $s = s(G)$. We deduce

$$\begin{aligned} RT(G,H) &= |A| - cp + bp + rr + \left\lceil \frac{s - gr + fr}{2} \right\rceil \\ &= \frac{1}{2}|A| - \gamma(G) + brs(G) + \left\lceil \frac{s(G) - gr(G) + fr(G)}{2} \right\rceil . \end{aligned}$$

Suppose $RT(G,H) > RT(G)$. This means that there exists a completed graph $\mathcal{G}_{s'}(\mathbf{V}, A, \Gamma'_s)$ with parameters cc', cp', bp', rr', s', gr', and fr' such that $cp' - bp' - rr' - \left\lceil \frac{s'-gr'+fr'}{2} \right\rceil > cp - bp - rr - \left\lceil \frac{s-gr+fr}{2} \right\rceil$, i.e.,

$$(1) \qquad cc' - cc > (rr' - rr) + \left(\left\lceil \frac{s' - gr' + fr'}{2} \right\rceil - \left\lceil \frac{s - gr + fr}{2} \right\rceil \right) .$$

Suppose first that the completed graph $\mathcal{G}_{s'}(\mathbf{V}, A, \Gamma'_s)$ contains x fewer real knots than $\mathcal{G}_s(\mathbf{V}, A, \Gamma_s)$. Suppose first that $x = 1$ and that it corresponds to a bad real SP. Then, from Lemma 10.7, $\mathcal{G}_{s'}(\mathbf{V}, A, \Gamma'_s)$ is a completed graph that is not maximal. More generally, a construction that removes x bad real SPs gives rise to a completed graph for which $cp' \leq c(G) - x$.

Suppose now that $\mathcal{G}_{s'}(\mathbf{V}, A, \Gamma'_s)$ contains one less semiknot than $\mathcal{G}_s(\mathbf{V}, A, \Gamma_s)$. This can occur in situations R2 or R3. However, removing such a semiknot, for example by constructing a good component, would also remove at least two cycles. Therefore,

$$(2) \qquad cp' - cp \leq (rr' - rr) + (s' - s).$$

We deduce from the above observations that

$$(\mathrm{I}) \qquad \left\lceil \frac{s' - gr' + fr'}{2} \right\rceil - \left\lceil \frac{s - gr + fr}{2} \right\rceil < s' - s.$$

Two possible situations occur:

1. $s' - gr' + fr'$ is even and $s - gr + fr$ is odd. In that case, inequality (I) induces $(s - s') < (-gr + fr) - (-gr' + fr') + 1$.
   But $s - s' \geq 0$ and $(-gr + fr) - (-gr' + fr') \leq 1$. This is due to the fact that if $fr = 1$, then $gr = 1$. (The same holds for fr' and gr'.)
   There are three possible cases: (a) $s - s' = 0$ and $(-gr+fr) - (-gr'+fr') = 0$; (b) $s - s' = 0$ and $(-gr + fr) - (-gr' + fr') = 1$; (c) $s - s' = 0$ and $(-gr + fr) - (-gr' + fr') = 1$.
   Cases (a) and (b) contradict the fact that $s' - gr' + fr'$ and $s - gr + fr$ are not both even or both odd.
2. All other situations for $s' - gr' + fr'$ and $s - gr + fr$ (other than $s' - gr' + fr'$ even and $s - gr + fr$ odd). In that case, inequality (I) induces $(s - s') < (-gr + fr) - (-gr' + fr')$. As $s - s' \geq 0$ and $(-gr + fr) - (-gr' + fr') \leq 1$, we should have $s - s' = 0$ and $(-gr + fr) - (-gr' + fr') = 1$.

Thus the only situation remaining is $s = s'$ and $(-gr + fr) = (-gr' + fr') + 1$. However, from the definitions of gr, fr, gr', and fr', this situation is impossible.

On the other hand, as the amalgamating procedure of $\mathcal{BPES}$ graphs preserves a maximal completed graph, a completed graph that contained more real knots or semiknots than $\mathcal{G}_s$ would not satisfy inequality (I). □

11. Genome halving with reversals.

11.1. The HP result. The problem of reconstructing a duplicated circular genome by reversals is a special case of the problem of reconstructing a duplicated multichromosomal genome by reversals and translocations. As the notion of endpoints is irrelevant for circular genomes, the distinction between a semicompleted graph and a completed graph is absent in this case. Let $\mathcal{G}(\mathbf{V}, A, \Gamma)$ be the completed graph obtained by *dedouble*. This graph can be decomposed into a set of alternating cycles (no paths). We define good and bad components in a similar way as for multichromosomal genomes (section 10), but by considering only cycles (no paths). Moreover, the concept of knots is here replaced by the concept of *hurdles*. Note that the concepts of "real hurdles" and "semihurdles" are irrelevant.

Let H_1 and H_2 be two single chromosome genomes, and let $\mathcal{G}_{12}$ be the breakpoint graph associated to H_1 and H_2. HP proved [17] that the minimal number of reversals required to transform H_1 to H_2 is

$$\textbf{HP4}: \quad R(H_1, H_2) = b(\mathcal{G}_{12}) - c(\mathcal{G}_{12}) + h(\mathcal{G}_{12}) + f(\mathcal{G}_{12}),$$

where $h(\mathcal{G}_{12})$ is the number of hurdles of $\mathcal{G}_{12}$ and $f(\mathcal{G}_{12})$ is a correction of size 0 or 1. In other words, in formula **HP1** (section 3), $m(\mathcal{G}_{12}) = h(\mathcal{G}_{12})$.

11.2. Maximizing the number of cycles. We denote by $R(G)$ the minimum number of reversals necessary to transform G into a duplicated genome. Denote by $c(G)$ the number of cycles of a maximal completed graph of $\mathcal{G}(\mathbf{V}, A)$. Theorem 6.6 gives an upper bound for $c(G)$. We would like to construct a completed graph with a number of cycles equal to this upper bound. This completed graph would then be maximal.

The method is almost identical to that described in section 7 for multichromosomal genomes. In particular, if we set $O = \emptyset$, then all the definitions and notation introduced in section 7 are valid for the circular genome case.

During the construction of gray edges, we still have to be careful not to create a circular fragment as long as unlinked vertices remain in the partially completed graph. In other words, the last step of the algorithm is the only one "closing" a fragment, eventually by constructing two gray edges of form $(x, \overline{x})$. Therefore, at each step except the last one (when there remain just four gray edges to be constructed to complete the graph), we have to construct possible pairs of gray edges, that is, pairs of gray edges that do not satisfy Properties I, II, and III (section 7).

In the case of circular genomes, if $\mathcal{SO}$ is not empty, then the set of "good" supernatural graphs, that is, the supernatural graphs of size $2n$ that can be completed by forming $n+1$ cycles, contains one supernatural graph of $\mathcal{SO}$ (Lemma 6.5 and Theorem 6.6). However, constructing $n+1$ cycles on a supernatural graph of $\mathcal{SO}$ creates a circular fragment. Therefore, to be able to construct a maximal number of cycles, we have to be careful to end up with a supernatural graph of $\mathcal{SO}$, if any.

The algorithm used in this case is also *dedouble*, with the slight difference described above. This algorithm constructs a maximal completed graph, that is, a completed graph with $c(\mathcal{G}_\Gamma) = \gamma + \frac{|A|}{2}$ cycles.

Example 7. Consider the genome $G = +a\ +b\ -c\ +b\ -d\ -e\ +a\ +c\ -d\ -e$. The decomposition of the partial graph into supernatural graphs is shown in Figure 11.1. We have $|A| = 10, \gamma = 3$, and thus $c = \gamma + \frac{A}{2} = 8$. Figure 11.1 depicts the completed graph produced by *dedouble*. The corresponding circular duplicated genome is

$$H = +c_1\ -b_1\ -a_1\ +e_2\ +d_2\ -d_1\ -e_1\ +a_2\ +b_2\ -c_2.$$

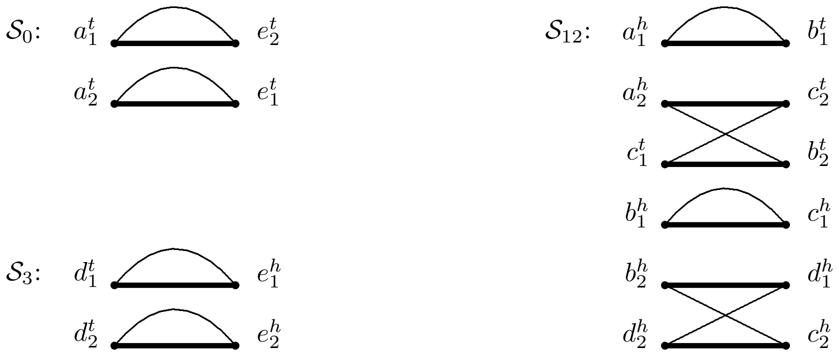

FIG. 11.1. *Completed graph $\mathcal{G}_\Gamma(\mathbf{V}, A, \Gamma)$ constructed by* dedouble.

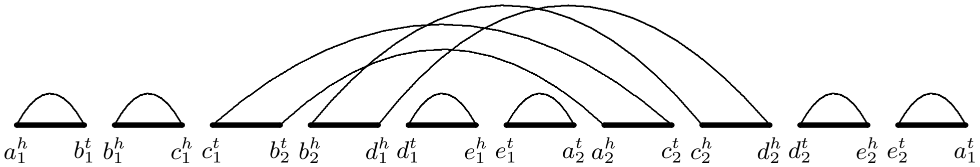

FIG. 11.2. *The completed graph constructed by* dedouble *for the genome G of Example* 7.

11.3. Hurdles. We now evaluate the number of hurdles contained in the maximal completed graph obtained by *dedouble*.

For circular genomes, the notion of a potential local SP is irrelevant, and only real local SPs remain. We saw, in section 8, how to modify *dedouble* so that, applied to a genome that does not contain any local SP, it gives rise to a completed graph containing no real SP.

The concepts of maximal, minimal, and bad SPs are defined as in section 10.3. Let $\mathbf{brs(G)}$ be the number of bad (real) SPs of G. Then, from Lemma 8.4 and Theorem 7.6, the completed graph $\mathcal{G}_\Gamma(\mathbf{V}, A, \Gamma)$ produced by *dedouble* contains exactly $brs(G)$ hurdles corresponding to these bad SPs. In addition, there may be at most two more special hurdles due to the special SPs defined in section 8.

Consider the parameter $f(G)$ which is 1 if the hurdles determined by the bad SPs of G form a fortress [17] and 0 otherwise. The next theorem is proved by arguments similar to those used for Theorem 10.8.

THEOREM 11.1. *Let $\mathcal{G}_\Gamma(\mathbf{V}, A, \Gamma)$ be the completed graph produced by* dedouble, *and let H be the resulting duplicated genome. Then*

$$\frac{|A|}{2} - \gamma(G) + brs(G) + f(G) \leq R(G, H) \leq \frac{|A|}{2} - \gamma(G) + brs(G) + f(G) + 2.$$

In addition,

$$\frac{|A|}{2} - \gamma(G) + brs(G) + f(G) \leq R(G) \leq \frac{|A|}{2} - \gamma(G) + brs(G) + f(G) + 2.$$

After Theorem 11.1, we have the following corollary.

COROLLARY 11.2. *Let $\mathcal{G}_\Gamma(\mathbf{V}, A, \Gamma)$ be the completed graph of $\mathcal{G}(\mathbf{V}, A)$ produced by* dedouble-circular. *If $\mathcal{G}_\Gamma(\mathbf{V}, A, \Gamma)$ does not contain any special hurdle, then*

$$R(G) = \frac{|A|}{2} - \gamma(G) + brs(G) + f(G).$$

Example 8. Consider genome G of Example 7 and the corresponding completed graph of Figure 7.5. Figure 11.2 gives a planar representation of this graph.

The number of cycles of this graph is $c(G) = 8$, $|A| = 10$, $brs(G) = 0$, and $f(G) = 0$. It does not contain any hurdles. Thus the minimum number of reversals necessary to transform genome G into a duplicated genome is $R(G) = 10-8+0+0 = 2$.

12. Analyzing the yeast genome.

Wolfe and Shields [39] proposed that yeast is a degenerate tetraploid resulting from a genome duplication 10^8 years ago. They identified 55 duplicated regions, representing 50% of the genome.

12.1. Sorting by translocations.

Applying our algorithm to the yeast genome data of Table 12.1, we obtain the perfect duplicated genome G_d represented in Table 12.2. The number of cycles of the corresponding completed graph $\mathcal{G}(V, A, \Gamma)$ is $c = 81$. Since G does not contain any local SPs, we can deduce that the minimal number of translocations required to transform G into G_d is

$$t = 2|\mathcal{B}| + |\mathcal{O}| - 2N - c = 142 - 16 - 81 = 45.$$

TABLE 12.1

Order of Wolfe and Shields' blocks on each of the 16 *chromosomes of the yeast genome. Signs indicate transcriptional orientation. In each chromosome, the* • *indicates the position of the centromere.*

I	:	+2 • −1
II	:	+4 • −3 − 7 +**8** − 5 + 6
III	:	+9 • −10 − 11
IV	:	+20 + 12 + 12 + 54 + 15 + 21 • −3 − 13 − 16 + 17 − 24 − 22 − 14 −23 − 19 + 18 − 9
V	:	+28 • −25 − 27 − 4 − 26 − 13
VI	:	+55 • −36
VII	:	+36 + 25 + 26 + 32 +**6** − **33** + 5 • −30 − 34 − 31 − 29
VIII	:	+35 • −14 − 37 − 29 − 1
IX	:	+38 + 39 + 27 •
X	:	+10 + 40 + 41 • −28 − 42
XI	:	+42 + 40 + 43 + 35 • −41 − 52 − 38
XII	:	+53 • −53 − 31 − 55 − 16 −**18** − **17** − 45 − 30 − 15 − 44
XIII	:	+46 + 44 + 19 • −43 − 54 − 48 − 47 − 46
XIV	:	+49 + 20 + 37 + 50 + 39 • −11
XV	:	+49 + 21 • −22 − 52 − 50 − 23 − 45 − 51 − 47 − 2
XVI	:	+48 + 32 + 33 + 51 + 8 + 24 • −7 − 34

TABLE 12.2

Order of Wolfe and Shields' blocks on each of the 16 *chromosomes of the A solution for the ancestral genome. The present-day yeast genome can be obtained from this one by genome doubling followed by* 45 *translocations.*

1	:	+2 − 1
2	:	+46 + 47 + 48 + 54 + 43 + 35 − 41 − 40 − 42
3	:	+9 − 10 − 11
4	:	+44 + 15 + 21 − 22 − 14 − 23 − 19 + 18 + 16;+13 + 26 +32 + 33 + 51 + 45 + 17 − 24 − 8 + 7 + 3 − 4
5	:	+55 − 36
6	:	+38 + 39 + 27 + 25 − 28
7	:	+29 + 37 + 50 + 52 − 53
8	:	+49 + 20 + 12 + 31 + 34 + 30 − 5 + 6

12.2. Sorting by reversals and translocations. As the yeast genome does not contain any real or potential local SPs, our method for sorting by reversals and translocations does not involve any reversals, so 45 translocations are still required.

13. An application on a circular genome.

The mitochondrial genome of the liverwort plant *Marchantia polymorpha* is rather unusual in that many of its genes are manifested in two or three copies [30]. It is very unlikely that these arose from genome doubling, since this would not account for the numerous triplicates, nor is it consistent with comparative data on mitochondrial genomes. Nevertheless, it provides a convenient small example to test our method. A somewhat artificial map was extracted from the Genbank entry, deleting all singleton genes and one gene from each triplet. (The two genes furthest apart were saved from each triplet.) This led to a "rearranged duplicated genome" with 25 pairs of genes. A single supernatural graph in $\mathcal{SE}$ emerged from the analysis. This produced a minimum of 25 inversions, which is what one would expect from a random distribution of the duplicate genes on the genome. Any trace of genome duplication, were this even biologically plausible, has been obscured.

14. Conclusions.

Calculating the HP formula for the edit distance between two genomes requires a rather intricate evaluation of the bicolored graph, including up to seven different structural parameters. In minimizing these formulae over the set of all (diploid) genomes, it is somewhat surprising that we can reconstruct an optimal ancestral genome exactly in all cases except the simplest reversals-only model. In the latter case, the uncertainty is not a deficiency of the algorithm but is due to ambiguity in how the doubled genome is constructed.

This work completes the major part of the program we undertook in [11]. In that article, we proposed a suite of "genome halving" problems and offered an algorithm for one of them in which a diploid genome is considered to be a set of genes partitioned among a number of subsets called chromosomes. The only operation is translocation considered as an exchange of subsets between two chromosomes. For the reconstruction problem in that context, in all likelihood NP-hard, we offered an effective heuristic which functions well on trial data. The present work shows that by adding gene order and transcription direction (strandedness, sign, polarity) to chromosome structure and adding the reversal operation, exact linear algorithms are possible. (Gene order alone, without transcription direction, would likely not suffice to permit polynomial-time exact algorithms; cf. [7].)

An additional level of structure to increase the realism of the model would be to incorporate a *centromere* on each chromosome. The centromere can occur anywhere in the linear order of genes, the centromeres are structurally indistinguishable from each other (for our purposes), and there is normally exactly one centromere per chromosome. This condition excludes some translocations, namely, those which result in one chromosome with two centromeres and one with none. The algorithms we have developed for the reconstruction of doubled genomes are not easily adaptable in this context. It should be noted that the condition on single centromeres is occasionally violated in nature, as, for example, with fissions and fusions, so that ideally the model should be complicated to allow for centromere creation and disappearance. As a final note, this work, together with [12] on hybridization and [9] on segment duplication, represents the use of computational biology techniques first developed for comparative genomics, as tools for the internal reconstruction of the evolutionary history of a single genome.

Acknowledgments. The authors are Scholar and Fellow of the Evolutionary Biology Program of the Canadian Institute for Advanced Research.

REFERENCES

[1] S. Ahn and S. D. Tanksley, *Comparative linkage maps of rice and maize genomes*, Proc. Natl. Acad. Sci. USA, 90 (1993), pp. 7980–7984.

[2] D. J. Amor and K. H. Choo, *Links neocentromeres: Role in human disease, evolution, and centromere study*, Amer. J. Human Genetics, 71 (2002), pp. 695–714.

[3] Arabidopsis genome initiative, *Analysis of the genome sequence of the flowering plant Arabidopsis thaliana*, Nature, 408 (2000), pp. 796–815.

[4] N. B. Atkin and S. Ohno, *DNA values of four primitive chordates*, Chromosoma, 23 (1967), pp. 10–13.

[5] D. A. Bader, B. M. E. Moret, and M. Yan, *A linear-time algorithm for computing inversion distances between signed permutations with an experimental study*, J. Comput. Biol., 8 (2001), pp. 483–491.

[6] A. Bergeron, *A very elementary presentation of the Hannenhalli-Pevzner theory*, in Proceedings of the 12th Annual Symposium on Combinatorial Pattern Matching, Lecture Notes in Comput. Sci. 2089, A. Amir and G. M. Landau, eds., Springer-Verlag, New York, 2001, pp. 106–117.

[7] A. Caprara, *Sorting by reversals is difficult*, in Proceedings of the First Annual International Conference on Computational Molecular Biology (RECOMB), ACM, New York, 1997, pp. 75–83.

[8] D. Durand, *Vertebrate evolution: Doubling and shuffling with a full deck*, Trends in Genetics, 19 (2003), pp. 2–5.

[9] N. El-Mabrouk, *Reconstructing an ancestral genome using minimum segments duplications and reversals*, J. Comput. System Sci., Special Issue on Computational Molecular Biology, 65 (2002), pp. 442–464.

[10] N. El-Mabrouk, B. Bryant, and D. Sankoff, *Reconstructing the pre-doubling genome*, in Proceedings of the Third Annual International Conference on Computational Molecular Biology (RECOMB), ACM, New York, 1999, pp. 154–163.

[11] N. El-Mabrouk, J. H. Nadeau, and D. Sankoff, *Genome halving*, in Proceedings of the 9th Annual Symposium on Combinatorial Pattern Matching, Lecture Notes in Comput. Sci. 1448, M. Farach, ed., Springer-Verlag, New York, 1998, pp. 235–250.

[12] N. El-Mabrouk and D. Sankoff, *Hybridization and genome rearrangement*, in Proceedings of the 10th Annual Symposium on Combinatorial Pattern Matching, Lecture Notes in Comput. Sci. 1645, M. Crochemore and M. Paterson, eds., Springer-Verlag, New York, 1999, pp. 78–87.

[13] R. Friedman and A. L. Hughes, *Pattern and timing of gene duplication in animal genomes*, Genome Res., 11 (2001), pp. 1842–1847.

[14] K. J. Fryxell, *The coevolution of gene family trees*, Trends in Genetics, 12 (1996), pp. 364–369.

[15] B. S. Gaut and J. F. Doebley, *DNA sequence evidence for the segmental allotetraploid origin of maize*, Proc. Natl. Acad. Sci. USA, 94 (1997), pp. 6809–6814.

[16] S. Hannenhalli, *Polynomial-time algorithm for computing translocation distance between genomes*, in Proceedings of the 6th Annual Symposium on Combinatorial Pattern Matching, Lecture Notes in Comput. Sci. 937, Z. Galil and E. Ukkonen, eds., Springer-Verlag, New York, 1995, pp. 162–176.

[17] 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 Theory of Computing, ACM, New York, SIAM, Philadelphia, 1995, pp. 178–189.

[18] S. Hannenhalli and P. A. Pevzner, *Transforming men into mice (polynomial algorithm for genomic distance problem)*, in Proceedings of the 36th IEEE Annual Symposium on Foundations of Computer Science, IEEE Computer Society, Los Alamitos, CA, 1995, pp. 581–592.

[19] M. Herdman, *The evolution of bacterial genomes*, in The Evolution of Genome Size, T. Cavalier-Smith, ed., John Wiley and Sons, New York, 1985, pp. 37–68.

[20] R. Hinegardner, *Evolution of cellular DNS content in teleost fishes*, American Naturalist, 102 (1968), pp. 517–523.

[21] A. L. Hughes, *Phylogenies of developmentally important proteins do not support the hypothesis of two rounds of genome duplication early in vertebrate history*, J. Molecular Evolution, 48 (1999), pp. 565–576.

[22] H. Kaplan, R. Shamir, and R. E. Tarjan, *A faster and simpler algorithm for sorting signed permutations by reversals*, SIAM J. Comput., 29 (1999), pp. 880–892.

[23] T. Kunisawa, *Identification and chromosomal distribution of DNA sequence segments conserved since divergence of Escherichia coli and Bacillus subtilis*, J. Molecular Evolution, 40 (1995), pp. 585–593.

[24] M. Lynch and J. S. Conery, *The evolutionary fate and consequences of duplicate genes*, Science, 290 (2000), pp. 1151–1155.

[25] A. McLysaght, K. Hokamp, and K. H. Wolfe, *Extensive genomic duplication during early chordate evolution*, Nature Genetics, 31 (2002), pp. 200–204.

[26] G. Moore, K. M. Devos, Z. Wang, and M. D. Gale, *Grasses, line up and form a circle*, Current Biology, 5 (1995), pp. 737–739.

[27] F. Muller, V. Bernard, and H. Tobler, *Chromatin diminution in nematodes*, Bioessays, 18 (1996), pp. 133–138.

[28] J. H. Nadeau, *Genome duplication and comparative mapping*, in Advanced Techniques in Chromosome Research, K. T. Adolph, ed., Marcel Dekker, New York, 1991, pp. 269–296.

[29] J. H. Nadeau and D. Sankoff, *Comparable rates of gene loss and functional divergence after genome duplications early in vertebrate evolution*, Genetics, 147 (1997), pp. 1259–1266.

[30] K. Oda, K. Yamato, E. Ohta, Y. Nakamura, M. Takemura, N. Nozato, T. Kohchi, Y. Ogura, T. Kanegae, K. Akashi, and K. Ohyama, *Gene organization deduced from the complete sequence of liverwort Marchantia polymorpha mitochondrial DNA. A primitive form of plant mitochondrial genome*, J. Molecular Biol., 223 (1992), pp. 1–7.

[31] S. Ohno, U. Wolf, and N. B. Atkin, *Evolution from fish to mammals by gene duplication*, Hereditas, 59 (1968), pp. 169–187.

[32] A. H. Paterson, T. H. Lan, K. P. Reischmann, C. Chang, Y. R. Lin, S. C. Liu, M. D. Burow, S. P. Kowalski, C. S. Katsar, T. A. DelMonte, K. A. Feldmann, K. F. Schertz, and J. F. Wendel, *Toward a unified genetic map of higher plants, transcending the monocot-dicot divergence*, Nature Genetics, 14 (1996), pp. 380–382.

[33] J. H. Postlethwait, Y. L. Yan, M. A. Gates, S. Horne, A. Amores, A. Brownlie, A. Donovan, E. S. Egan, A. Force, Z. Gong, C. Goutel, A. Fritz, R. Kelsh, E. Knapik, E. Liao, B. Paw, D. Ransom, A. Singer, T. Thomson, T. S. Abduljabbar, P. Yelick, D. Beier, J. S. Joly, D. Larhammar, F. Rosa, M. Westerfield, L. I. Zon, and W. S. Talbot, *Vertebrate genome evolution and the zebrafish gene map*, Nature Genetics, 18 (1998), pp. 345–349.

[34] J. A. Scheffler, A. G. Sharpe, H. Schmidt, P. Sperling, I. A. P. Parkin, W. Lühs, D. J. Lydiate, and E. Heinz, *Desaturase multigene families of Brassica napus arose through genome duplication*, Theoretical and Applied Genetics, 94 (1997), pp. 583–591.

[35] C. Seoighe and K. H. Wolfe, *Extent of genomic rearrangement after genome duplication in yeast*, Proc. Natl. Acad. Sci. USA, 95 (1998), pp. 4447–4452.

[36] R. C. Shoemaker, K. Polzin, J. Labate, J. Specht, E. C. Brummer, T. Olson, N. Young, V. Concibido, J. Wilcox, J. P. Tamulonis, G. Kochert, and H. R. Boerma, *Genome duplication in soybean (Glycine subgenus soja)*, Genetics, 144 (1996), pp. 329–228.

[37] A. C. Siepel, *An algorithm to find all sorting reversals*, in Proceedings of the 6th Annual International Conference on Computational Molecular Biology (RECOMB), ACM, New York, 2002, pp. 281–290.

[38] L. Skrabanek and K. H. Wolfe, *Eukaryote genome duplication, where's the evidence?*, Current Opinion in Genetics and Development, 8 (1998), pp. 694–700.

[39] K. H. Wolfe and D. C. Shields, *Molecular evidence for an ancient duplication of the entire yeast genome*, Nature, 387 (1997), pp. 708–713.

[40] R. H. Xu, J. Kim, M. Taira, J. J. Lin, C. H. Zhang, D. Sredni, T. Evans, and H. F. Kung, *Differential regulation of neurogenesis by the two Xenopus GATA-1 genes*, Molecular and Cellular Biology, 17 (1997), pp. 436–443.