Whole Genome Duplications, Multi-Break Rearrangements, and Genome Halving Problem

Max A. Alekseyev and Pavel A. Pevzner
Department of Computer Science and Engineering
University of California at San Diego
{maxal,ppevzner}@cs.ucsd.edu

Abstract

The Genome Halving Problem, motivated by the whole genome duplication events in molecular evolution, was solved by El-Mabrouk and Sankoff. The El-Mabrouk–Sankoff algorithm is rather complex inspiring a quest for a simpler solution. An alternative approach to Genome Halving Problem based on the notion of the contracted breakpoint graph was recently proposed in [2]. This new technique reveals that while the El-Mabrouk–Sankoff result is correct in most cases, it does not hold in the case of unichromosomal genomes. This raises a problem of correcting El-Mabrouk–Sankoff analysis and devising an algorithm that deals adequately with all genomes. In this paper we efficiently classify all genomes into two classes and show that while the El-Mabrouk–Sankoff theorem holds for the first class, it is incorrect for the second class. The crux of our analysis is a new combinatorial invariant defined on duplicated permutations. Using this invariant we were able to come up with a full proof of the Genome Halving theorem and a polynomial algorithm for Genome Halving Problem (for unichromosomal genomes). We also give the first short proof of the original El-Mabrouk–Sankoff result for multichromosomal genomes. Finally, we discuss a generalization of Genome Halving Problem for a more general set of rearrangement operations (including transpositions) and propose an efficient algorithm for solving this problem.

1 Introduction

In 1970 Susumu Ohno came up with two fundamental theories of chromosome evolution that were subjects to many controversies in the last 35 years [38]. The first, Random Breakage theory, was embraced by biologists from the very beginning but was refuted by Pevzner and Tesler, 2003 [43]. The second, Whole Genome Duplication theory, postulated a new type of evolutionary events and had a very different fate. It was subject to controversy in the first 35 years and only recently was proven to be correct [33, 17]. Kellis et al., 2004 [33] sequenced yeast *K. waltii* genome, compared it with yeast *S. cerevisiae* genome, and demonstrated that nearly every region in *K. waltii* corresponds to two regions in *S. cerevisiae* thus proving that there was a whole genome duplication event in the course of yeast evolution. This discovery quickly followed by the discovery of the whole genome duplications in vertebrates [30, 45, 15] and plants [25]. Recently Dehal and Boore [16] found an evidence of two rounds of whole genome duplications on the evolutionary path from early vertebrates to human. Shortly afterwards, Meyer and Van de Peer [36] found an evidence of yet another (third) round of whole genome duplications in ray-finned fishes.

These recent studies provided an irrefutable evidence that the whole genome duplications represent a new type of events that may explain phenomena that the classical evolutionary studies had difficulties explaining (e.g., emergence of new metabolic pathways [33]). At the same time, they raised a problem of reconstructing the genomic architecture of the ancestral pre-duplicated genomes. Unfortunately, since the El-Mabrouk–Sankoff algorithm for solving this problem [22] has not resulted in a software tool yet, the recent studies of whole genome duplications did not attempt to rigorously reconstruct the architecture of the pre-duplicated genomes. We revisited the El-Mabrouk–Sankoff result, found a flaw in their analysis of unichromosomal genomes, re-formulated and proved the Genome Halving theorem and developed a new algorithm and software tool for studies of genome duplications. We further analyze a generalization of Genome Halving Problem for a more general set of rearrangement operations (including transpositions) and propose an efficient algorithm for solving this problem.

The *whole genome duplication* doubles the gene content of a genome R and results in a *perfect duplicated genome* Q that contains two copies of each chromosome of R. The genome then becomes subject to rearrangements that shuffle the genes in Q resulting in some *duplicated genome* P. The *Genome Halving Problem* is to reconstruct the ancestral perfect duplicated genome Q from the given duplicated genome P (Fig. 1a).

From an algorithmic perspective, the genome is a collection of chromosomes, and each chromosome is a sequence of genes. In this paper we assume that chromosomes are *circular*. DNA has two strands and genes on a chromosome have directionality that reflects the strand of the genes. We represent the order and directions of the genes on each chromosome as a circular sequence of *signed* elements (i.e., elements with signs "+" and "-"). We distinguish between *unichromosomal genomes* consisting of just a single chromosome and *multichromosomal genomes* consisting of one or more chromosomes.

For unichromosomal genomes, the rearrangements are limited to *reversals* that "flip" genes $x_i \dots x_j$ in a genome $x_1 x_2 \dots x_n$ as follows:

$$\begin{array}{c} x_1 \dots x_{i-1} \;\; x_i \;\; x_{i+1} \dots \;\; x_j x_{j+1} \dots x_n \\ \xrightarrow{\qquad\qquad} \\ \downarrow \\ x_1 \dots x_{i-1} - x_j - x_{j-1} \cdots - x_i x_{j+1} \dots x_n \\ \xleftarrow{\qquad\qquad} \end{array}$$

The *reversal distance* $d(P, Q)$ between genomes P and Q is defined as the minimal number of reversals required to transform one genome into the other (see Chapter 10 of [42] for a review of genome rearrangement algorithms).

We represent a circular chromosome R as a cycle formed by directed edges encoding the genes and their direction (Fig. 1b, center). There are two natural ways to represent duplication of the chromosome R resulting in a

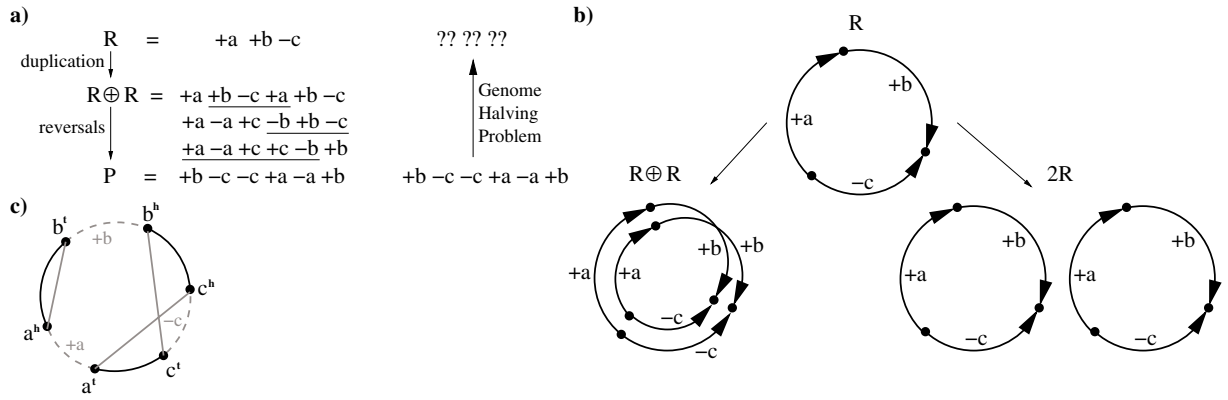

Figure 1: a) Whole genome duplication of genome $R = +a + b - c$ into a perfect duplicated genome $R \oplus R = +a + b - c + a + b - c$ followed by three reversals. b) Whole genome duplication of a circular chromosome R (center) resulting in $R \oplus R$ (left) or $2R$ (right). c) Breakpoint graph of genomes $+a + b - c$ and $+a + b + c$.

single chromosome $R \oplus R$ (Fig. 1b, left) or in two chromosomes $2R$ (Fig. 1b, right) but only the former one is applicable to unichromosomal genomes. A *unichromosomal duplicated genome* is a result of a series of reversals applied to the *unichromosomal perfect duplicated genome* $R \oplus R$. The Genome Halving Problems for unichromosomal genomes is formulated as follows:

Genome Halving Problem (unichromosomal genomes). Given a unichromosomal duplicated genome P, find a perfect unichromosomal duplicated genome $R \oplus R$ minimizing the reversal distance $d(P, R \oplus R)$.

For multichromosomal genomes, there is a broader set of rearrangement operations including *translocations* (that interchange segments between two chromosomes), *fissions* (that break a chromosome into two), *fusions* (that merge two chromosomes into a single one) as well as reversals (see [42]). Each of these operations can be viewed as a *2-break* that "cuts" circular chromosome(s) of the genome at 2 points and "connects" the resulting two linear fragments back into circular chromosome(s) in a certain order. Similarly to the reversal distance, we define the *genomic distance* $d_2(P, Q)$ between multichromosomal genomes P and Q as the minimum number of these operations (2-breaks) required to transform one genome into the other.

A whole genome duplication of a multichromosomal genome consisting of chromosomes $R_1, \ldots, R_k$ results in a *multichromosomal perfect duplicated genome*[1] where every chromosome R_i is duplicated either into $R_i \oplus R_i$ or into $2R_i$ (Fig. 1b). A *multichromosomal duplicated genome* is a result of a series of 2-breaks applied to a perfect duplicated genome. The Genome Halving Problem for multichromosomal genomes is formulated as follows.

Genome Halving Problem (multichromosomal genomes). Given a duplicated genome P, find a perfect duplicated genome Q minimizing the genomic distance $d_2(P, Q)$.

The Genome Halving Problem was studied in a series of papers by El-Mabrouk and Sankoff [20, 21, 19] culminating in a rather complex algorithm in [22]. The El-Mabrouk–Sankoff algorithm is one of the most technically challenging results in computational biology and its proof spans over 30 pages in [22]. Recently Alekseyev and Pevzner, 2006 [2] revisited the El-Mabrouk–Sankoff work and presented an alternative approach based on the notion of *contracted breakpoint graph*.

After paper [2] was written, our studies of the contracted breakpoint graph led us to realize that El-Mabrouk–Sankoff analysis has a flaw and the problem of finding $\min_R d(P, R \oplus R)$ remains unsolved for for unichromosomal genomes. Below we show that this flaw is a rule rather than a pathological case: it affects a large family of duplicated genomes. We further proceed to give a full analysis of the Genome Halving Problem that is based on introducing an invariant that divides the set of all rearranged duplicated genomes into 2 classes. We show that the El-Mabrouk–Sankoff formula is correct for the first class and is off by 1 for the second class. We remark that our approach is very different from [22] and we do not know whether the technique in [22] can be adjusted to address the described complication.

In addition to reversals, fusions, fissions, and translocations, multichromosomal genomes are also subject to *transpositions* and *inverted transpositions* (although these operations are believed to be rare in mammalian evolution). Transpositions cut genome at 3 points (creating 3 linear fragments) and further connect the resulting linear fragments in a *certain* order. Following [1], we model transpositions as *3-breaks* that "cut" the genome at three points and "connect" the resulting fragments back into a single genome in an *arbitrary* way. 3-Breaks include as a particular case 2-breaks (i.e., reversals/fusions/fissions/translocations), transpositions, inverted transpositions, as well as *3-way fissions*. We define the 3-break distance $d_3(P, Q)$ between genomes P and Q as the minimal number of 3-breaks required to transform one genome into the other. In paper we solve the following 3-Break Genome Halving Problem:

3-*Break Genome Halving Problem.* Given a duplicated genome P, find a perfect duplicated genome Q minimizing the 3-break distance $d_3(P, Q)$.

The paper is organized as follows. Section 2 introduces the notion of breakpoint graph and explains its relation to rearrangement analysis. Section 3 presents the concept of contracted breakpoint graph and extends some results from [2] to the case of multichromosomal genomes. We solve the Genome Halving Problem for multichromosomal genomes and the 3-Break Genome Halving Problem in Sections 4 and 5 respectively. Section 6 describes a flaw in El-Mabrouk-Sankoff analysis. Finally, in Section 7 we

[1] Note that in difference from the unichromosomal genomes, the whole genome duplication of a multichromosomal genome is not uniquely defined.

solve the Genome Halving Problem for unichromosomal genomes and classify the genomes for which the original El-Mabrouk–Sankoff theorem is incorrect.

2 Reversals, 2-Breaks, and Breakpoint Graphs

2.1 Unichromosomal Genomes A duality theorem and a polynomial algorithm for computing reversal distance between two signed permutations was proposed by Hannenhalli and Pevzner [27] and later was generalized for multichromosomal genomes [26]. The algorithm was further simplified and improved in a series of papers [8, 31, 3, 9, 49, 32] and applied in a variety of biological studies [37, 12, 10, 41, 7].

A signed permutation on n elements can be transformed into an unsigned permutation on $2n$ elements (see [5]) by substituting every element x in the signed permutation by two elements x^t and x^h in the unsigned permutation[2]. Each element $+x$ in the permutation P is replaced with $x^t x^h$, and each element $-x$ is replaced with $x^h x^t$ resulting in an unsigned circular permutation $\pi(P)$. For example, a permutation $+a+b-c$ will be transformed into $a^t a^h b^t b^h c^h c^t$. Element x^t is called an *obverse* of element x^h, and vice versa.

Let P and Q be circular signed permutations (unichromosomal genomes) on the same set of elements (genes) $\mathcal{G}$, and $\pi(P)$ and $\pi(Q)$ be corresponding unsigned permutations. The *breakpoint graph* $G(P, Q)$ is defined on the set of vertices $V = \{x^t, x^h \mid x \in \mathcal{G}\}$ with edges of three colors[3]: "obverse", black, and gray (Fig. 1c). Edges of each color form a matching on V:

- pairs of obverse elements form an *obverse matching*;
- adjacent elements in $\pi(P)$, other than obverses, form a *black matching*;
- adjacent elements in $\pi(Q)$, other than obverses, form a *gray matching*.

Every pair of matchings forms a collection of *alternating* cycles in G, called *black-gray*, *black-obverse*, and *gray-obverse* cycles respectively (a cycle is alternating if colors of its edges alternate). The permutation $\pi(P)$ can be read along a single black-obverse cycle while the permutation $\pi(Q)$ can be read along a single gray-obverse cycle in G. The black-gray cycles in the breakpoint graph play an important role in computing the reversal distance. According to the Hannenhalli–Pevzner theorem, the reversal distance between permutations P and Q is given by the formula:

$$d(P, Q) = |P| - c(P, Q) + h(P, Q) \tag{2.1}$$

where $|P| = |Q|$ is the size of P and Q, $c(P, Q) = c(G(P, Q))$ is the number of black-gray cycles in the breakpoint graph $G(P, Q)$, and $h(P, Q)$ is an easily computable combinatorial parameter. While this result leads to a fast algorithm for computing reversal distance between two signed permutations, the problem of computing reversal distance between two genomes with duplicated genes remains unsolved.

Let P and Q be duplicated genomes on the same set of genes $\mathcal{G}$. If one labels copies of each gene x as x_1 and x_2 then the genomes P and Q become signed permutations and (2.1) applies. Similarly to non-duplicated genomes, for the labelled duplicated genomes P and Q, we can also construct unsigned permutations $\pi(P)$, $\pi(Q)$ and the breakpoint graph $G(P, Q)$ on a vertex set $V = \{x_1^t, x_1^h, x_2^t, x_2^h \mid x \in \mathcal{G}\}$. The pairs of vertices x_1^i and x_2^i ($i \in \{t, h\}$) form yet another matching in the breakpoint graph $G(P, Q)$ called *counterpart* (see legend for Fig. 2). Counterpart of a vertex v is denoted $\bar{v}$ so that $\bar{x}_1^i = x_2^i$ and $\bar{x}_2^i = x_1^i$.

We remark that different labellings may lead to different breakpoint graphs (on the same vertex set) for the same genomes P and Q (Fig. 2) and it is not clear how to choose a labelling that results in the minimum reversal distance between the labelled copies of P and Q. Recently there were many attempts to generalize the Hannenhalli-Pevzner theory for genomes with duplicated and deleted genes [11, 13, 18, 46, 47, 48]. However, the only known option for finding the reversal distance between two duplicated genomes *exactly* is to consider all possible labellings, to compute the reversal distance for each labelling, and to choose a labelling with the minimal reversal distance. For duplicated genomes with n genes this leads to 2^n invocations of the Hannenhalli–Pevzner algorithm rendering this approach impractical. Moreover, the problem remains open if one of the genomes is perfectly duplicated (i.e., computing $d(P, R \oplus R)$). Surprisingly, the problem of computing $\min_R d(P, R \oplus R)$ that we address in this paper is solvable in polynomial time.

Using the concept of the breakpoint graph and formula (2.1), the Genome Halving Problem for unichromosomal genomes can be posed as follows. For a duplicated genome P, find a perfect duplicated genome $Q = R \oplus R$ and a labelling of gene copies in P and Q such that the value of $d(P, Q) = |P| - c(P, Q) + h(P, Q)$ is minimum. Since $|P|$ is constant and $h(P, Q)$ is typically small (see [42]), the value of $d(P, Q)$ depends mainly on $c(P, Q)$. El-Mabrouk and Sankoff [22] established that the problems of maximizing $c(P, Q)$ and minimizing $h(P, Q)$ can be solved separately in a consecutive manner.[4] In this paper we focus on the former and harder problem:

Weak Genome Halving Problem. For a given unichromosomal duplicated genome P, find a perfect duplicated genome $Q = R \oplus R$ and a labelling of P and Q that maximizes the number of black-gray cycles $c(P, Q)$ in the breakpoint graph $G(P, Q)$ of the labelled genomes P and Q.

2.2 Multichromosomal genomes Let P be a multichromosomal genome represented as a collection of black-obverse cycles (chromosomes). For any two black edges (u, v) and (x, y) in a genome (graph) P, we define a *2-break* rearrangement as replacement of these edges with either a pair of edges (u, x), (v, y), or a pair of edges (u, y), (v, x) (Fig. 3). 2-breaks correspond to standard rearrangement operations of reversals (Fig. 3a), fissions (Fig. 3b), or fusions/translocations[5] (Fig. 3c).

Let P and Q be two multichromosomal genomes on the same set of genes $\mathcal{G}$. Similarly to the unichromosomal case, the *breakpoint graph* $G(P, Q)$ is defined on the set of vertices $V = \{x^t, x^h \mid x \in \mathcal{G}\}$ with edges of three colors: "obverse", black, and gray. Edges of each color form a matching on V: *obverse matching* (pairs of obverse

[2] Indices "t" and "h" stand for "tail" and "head" respectively.

[3] We have chosen rather unusual names for the "colors" (obverse, black, and gray) to be consistent with previous papers on genome rearrangements.

[4] An analog of formula (2.1) without the "$h(G)$" term corresponds to the 2-break distance between multichromosomal genomes (Theorem 2.1 below).

[5] This definition of elementary rearrangement operations follows the standard definitions of reversals, translocations, fissions, and fusions for the case of circular chromosomes. For circular chromosomes fusions and translocations are not distinguishable.

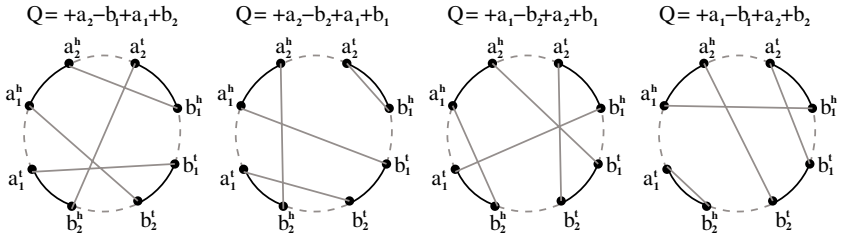

Figure 2: Breakpoint graphs for $P = +a - a - b + b$ and four different labellings $Q = +a - b + a + b$ (we assume that the labelling of P as $+a_1 - a_2 - b_1 + b_2$ is fixed). Two out of four breakpoint graphs have $c(P, Q) = 1$, while two others have $c(P, Q) = 2$. The counterpart matching in these graphs is formed by pairs (a_1^t, a_2^t), (a_1^h, a_2^h), (b_1^t, b_2^t), (b_1^h, b_2^h).

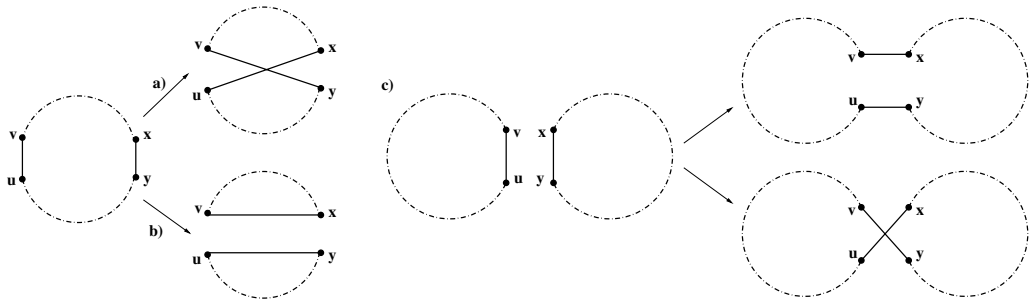

Figure 3: 2-break on edges (u, v) and (x, y) corresponding to a) Reversal: the edges belong to the same black-obverse cycle that is rearranged after 2-break; b) Fission: the edges belong to the same black-obverse cycle that is split by 2-break; c) Translocation/fusion: the edges belong to different black-obverse cycles that are joined by 2-break.

vertices), *black matching* (adjacent vertices in P), and *gray matching* (adjacent vertices in Q). Every pair of matchings forms a collection of *alternating* cycles in G, called *black-gray*, *black-obverse*, and *gray-obverse* cycles respectively. The chromosomes of genome P (resp. Q) represent black-obverse (resp. gray-obverse) cycles in $G(P, Q)$.

Every 2-break in a genome P corresponds to a transformation of the breakpoint graph $G(P, Q)$. Since the breakpoint graph of two identical genomes is a collection of *trivial* black-gray cycles of length 2 (the *identity breakpoint graph*), the problem of transforming the genome P into the genome Q by 2-breaks can be formulated as the problem of transforming the breakpoint graph $G(P, Q)$ into the identity breakpoint graph. This is equivalent to the following problem:

2-*Break Distance Problem.* Given two perfect matchings (black and gray matchings) in a graph, find a shortest series of 2-breaks that transforms one matching into the other.

In difference from the Genomic Distance Problem [26, 50, 39] (for linear multichromosomal genomes), the 2-Break Distance Problem for circular multichromosomal genomes is trivial (compare to [53]):

Theorem 2.1 *The 2-break distance between a black matching P and a gray matching Q is $d_2(P, Q) = |P| - c(P, Q)$.*

Proof It is easy to see that every non-trivial black-gray cycle can be split into two by a 2-break. Since no 2-break can increase the number of black-gray cycles by more than 1, the 2-break distance between P and Q is $|P| - c(P, Q)$. □

While Theorem 2.1 leads to a polynomial algorithm for computing the 2-break distance between genomes with non-duplicated genes, it is unclear how one can compute this distance between duplicated genomes without going over all possible labellings of the genomes. In the next section we describe the contracted breakpoint graphs that address this complication.

3 Duplicated Genomes and Contracted Breakpoint Graphs

While unichromosomal genomes can be considered as a particular case of multichromosomal genomes, many statements about multichromosomal genomes appear to be simpler than their analogs for unichromosomal genomes. Below we review basic concepts from [2] and extend them to multichromosomal genomes.

Let P and Q be multichromosomal duplicated genomes on the same set of genes $\mathcal{G}$. We represent each chromosome of the genome P (resp. Q) as a black-obverse (resp. gray-obverse) cycle on vertices from the set $V = \{x^t, x^h \mid x \in \mathcal{G}\}$. Given a set of edge-labelled graphs, the *de Bruijn graph* of this set is defined as the result of "gluing" edges with the same label in all graphs in the set (see [40, 2]). Let $\hat{P}$ be the de Bruijn graph of the genome P (Fig. 4a) and $\hat{Q}$ be the de Bruijn graph of the genome Q (Fig. 4b). Obviously, the de Bruijn graph of the genomes P and Q coincides with the de Bruijn graph of $\hat{P}$ and $\hat{Q}$ (Fig. 4d).

The *contracted breakpoint graph* $G'(P, Q)$ of duplicated genomes P and Q is an undirected vertex-labelled version of the de Bruijn graph (Fig. 4d,e) such that

- "tail" and "head" of every directed obverse edge x in the de Bruijn corresponds to vertices x^t and x^h in $G'(P, Q)$;
- each obverse edge in $G'(P, Q)$ is viewed as a *pair* of parallel edges.

Below we give an equivalent definition of the contracted breakpoint graph and show its connection to the breakpoint graphs for labelled genomes.

Let $G(P, Q)$ be a breakpoint graph defined by some labelling of P and Q. The contracted breakpoint graph

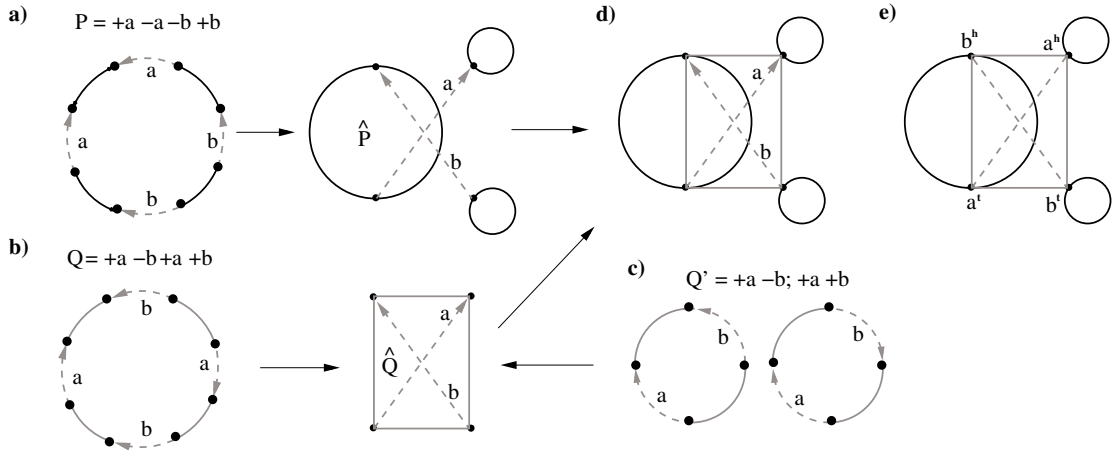

Figure 4: a) Genome $P = +a - a - b + b$ as a black-obverse cycle and its transformation into $\hat{P}$ by gluing identically labelled edges; b) Genome $Q = +a - b + a + b$ as a gray-obverse cycle and its transformation into $\hat{Q}$ by gluing identically labelled edges; c) Two-chromosomal genome $Q' = (+a - b)(+a + b)$ that is equivalent to the genome Q ($\hat{Q'} = \hat{Q}$); d) de Bruijn graph for P and Q; e) Contracted breakpoint graph $G'(P, Q)$.

$G'(P, Q)$ is the result of contracting every pair of vertices x_1^i, x_2^i (where $x \in \mathcal{G},\ i \in \{t, h\}$) in the breakpoint graph $G(P, Q)$ into a single vertex x^i. Hence, the contracted breakpoint graph $G'(P, Q)$ is a graph on the set of vertices $V' = \{x^t, x^h \mid x \in \mathcal{G}\}$ with each vertex incident to two black, two gray, and a *pair* of parallel obverse edges (Fig. 4e). Note that the contracted breakpoint graph $G'(P, Q)$ does not depend on a particular labelling of P and Q.

The following theorem gives a characterization of the contracted breakpoint graphs (for multichromosomal genomes).

Theorem 3.1 *A graph H with black, gray, and obverse edges is a contracted breakpoint graph for some duplicated genomes iff each vertex in H is incident to two black edges, two gray edges, and two parallel obverse edges.*

Proof If H is a contracted breakpoint graph of some duplicated genomes then the theorem follows from the definition of contracted breakpoint graph.

Let H be a graph with each vertex incident to two black edges, two gray edges, and a pair of parallel obverse edges. Label endpoints of each obverse edge x in H by x^t and x^h. Since the black degree of each vertex of H is even and so is obverse degree, there exist alternating black-obverse cycles traversing all black and obverse edges in this graph. These cycles define some duplicated genome P. Similarly, since the gray degree of each vertex of H is even, there exist alternating gray-obverse cycles traversing all gray and obverse edges. These cycles define some duplicated genome Q. Then the graph H is a contracted breakpoint graph for the genomes P and Q. □

In the case when Q is a perfect duplicated genome, the gray edges in the contracted breakpoint graph $G'(P, Q)$ form pairs of parallel gray edges that we refer to as *double* gray edges. Similar to the double obverse edges, the double gray edges form a matching in G' (Fig. 6a).

Let $G(P, Q)$ be a breakpoint graph for some labelling of P and Q. A set of black-gray cycles in $G(P, Q)$ is contracted into a set of black-gray cycles in the contracted breakpoint graph $G'(P, Q)$, thus forming a black-gray cycle decomposition of $G'(P, Q)$. Therefore, each labelling of P and Q *induces* a black-gray cycle decomposition of $G'(P, Q)$. We are interested in the following problem:

Labelling Problem. Given a black-gray cycle decomposition of the contracted breakpoint graph $G'(P, Q)$ of duplicated genomes P and Q, find a labelling of P and Q that induces this cycle decomposition.

This problem may not always have a solution for unichromosomal genomes (Fig. 5) and this is exactly the factor that leads to a counterexample to the El-Mabrouk–Sankoff theorem in Section 6. For multichromosomal genomes, the Labelling Problem can be addressed by considering equivalent genomes.

We call genomes Q and Q' *equivalent* if their de Bruijn graphs are equal, i.e., $\hat{Q} = \hat{Q'}$. If Q and Q' are equivalent then the contracted breakpoint graphs $G'(P, Q)$ and $G'(P, Q')$ are the same for any genome P (Fig. 4d).

Lemma 3.1 *If Q is a perfect duplicated genome and a genome Q' is equivalent to Q then Q' is perfect duplicated as well.*

Proof Consider gray and obverse matchings in the de Bruijn graph $\hat{Q} = \hat{Q'}$ formed by pairs of double gray and double obverse edges. These matchings form a set of gray-obverse cycles (consisting of double edges). Every such cycle c is the result of gluing some gray-obverse cycles $c_1, c_2, \ldots, c_k$ in Q' such that $|c_1|+|c_2|+\ldots+|c_k| = 2\cdot|c|$. Neither of these cycles can be shorter than c since such a short cycle would remain short after gluing. This implies that $k = 1$ or $k = 2$, i.e., the genome Q' has either a single cycle (a chromosome $R \oplus R$) traversing the cycle c two times or two cycles (a pair of chromosomes $2R$) each traversing c once. Therefore, the genome Q' represents a set of sub-genomes of the form $R \oplus R$ or $2R$ implying that Q' is a perfect duplicated genome. □

Theorem 3.1 and Lemma 3.1 imply

Theorem 3.2 *A graph H with black, gray, and obverse edges is a contracted breakpoint graph $G'(P, Q)$ for some duplicated genome P and perfect duplicated genome Q iff each vertex in H is incident to two black edges, a double gray edge, and a double obverse edge.*

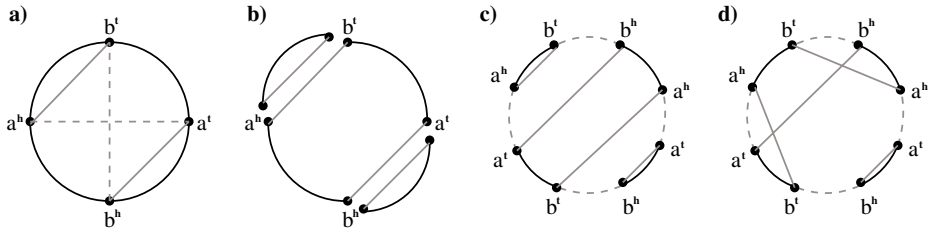

Figure 5: a) Contracted breakpoint graph $G'(P, R \oplus R)$ for $P = +a + b - a - b$ and $R = +a + b$; b) Black-gray cycle decomposition C of G' which is not induced by any labelling of P and $R \oplus R$; c) Breakpoint graph $G(P, 2R)$ inducing C; d) Breakpoint graph $G(P, R \oplus R)$ (unique up to re-labelling of vertices) with $c(G) = 2 < |C|$.

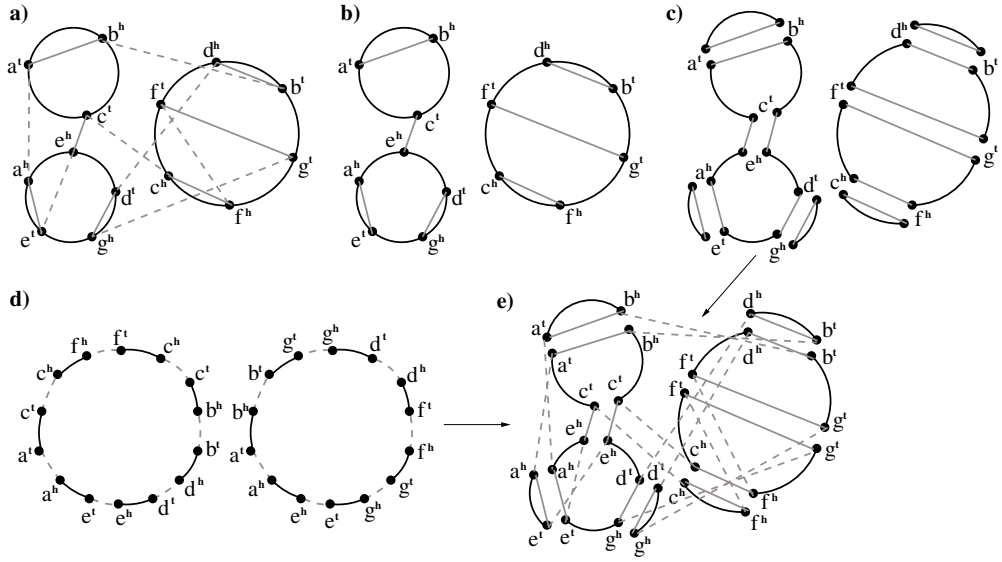

Figure 6: For genomes $P = (-a-b+g+d+f+g+e)(-a+c-f-c-b-d-e)$ and $R = -a-b-d-g+f-c-e$, a) contracted breakpoint graph $G'(P, R \oplus R)$; b) BG-graph corresponding to G'; c) maximal black-gray cycle decomposition (split decomposition) C of G' forming graph H; d) genome P as black-obverse cycles; e) breakpoint graph $G(P, Q')$ inducing the cycle decomposition C.

While the Labelling Problem may not have a solution, the following theorem provides a "compromise" substitute for its solution.

Theorem 3.3 *Let P and Q be multichromosomal duplicated genomes and C be a black-gray cycle decomposition of the contracted breakpoint graph $G'(P, Q)$. Then there exists a genome Q' equivalent to Q and a labelling of P and Q' such that the breakpoint graph $G(P, Q')$ induces the cycle decomposition C.*

Proof Consider a contracted breakpoint graph $G' = G'(P, Q)$ of the genomes P and Q and its black-gray cycle decomposition C (Fig. 6a gives an example of a contracted breakpoint graph while Fig. 6c gives an example of its black-gray cycle decomposition). Without loss of generality we can assume that the labelling of P is fixed. In order to prove the theorem we need to find a breakpoint graph $G(P, Q')$ of the labelled genomes P and Q' (Q' is equivalent to Q) whose black-gray cycle decomposition is contracted into C.

We will find it convenient to represent the cycle decomposition of G' as a graph H (Fig. 6c) where every cycle in C forms its own connected component and will assume that every vertex of the graph G' has two copies in H with identical labels (i.e., graph H has twice the number of vertices as compared to G'). We will show how to transform H into a breakpoint graph $G(P, Q')$ of the labelled genomes P and Q'. To achieve this goal we need to re-label the identically labelled vertices x and x in H into x_1 and x_2, and satisfy the condition that H is a breakpoint graph $G(P, Q')$ for the labelled genomes P and Q' with $\hat{Q}' = \hat{Q}$.

The genome P defines a collection of black-obverse cycles (Fig. 6d). Traversing black edges in graph H in the order given by these cycles defines a set of obverse edges in H (Fig. 6e) and a labelling of vertices in H as imposed by the fixed labelling of P. The set of these obverse edges forms matching in H and defines a gray-obverse cycle decomposition. This gray-obverse cycle decomposition defines a labelled multichromosomal genome Q' that is equivalent to Q. By the construction, the graph H with the set of obverse edges represents a breakpoint graph $G(P, Q')$ that induces the cycle decomposition C. □

4 2-Break Genome Halving Problem

In this section we solve the 2-Break Genome Halving Problem for a duplicated genome P by minimizing the 2-break distance $d_2(P, Q)$ over all perfect duplicated genomes Q.

Let $c_{max}(P, Q)$ be the number of cycles in a maximal black-gray cycle decomposition of the contracted breakpoint graph $G'(P, Q)$. Theorems 2.1 and 3.3 motivate the following reformulation of the 2-Break Genome Halving

Problem:

Cycle Decomposition Problem. For a given duplicated (unichromosomal or multichromosomal) genome P, find a perfect duplicated (resp. unichromosomal or multichromosomal) genome Q maximizing $c_{max}(P, Q)$.

Alekseyev and Pevzner, 2005 [2] introduced *BG-graphs* to address the Cycle Decomposition Problem in the case of unichromosomal genomes. A BG-graph is defined as a graph with black and double gray edges such that the black edges form black cycles and the double gray edges form a gray matching (Fig. 6b). Every contracted breakpoint graph $G'(P, Q)$ where Q is perfect duplicated genome represents a BG-graph.

A double gray edge in a BG-graph is called *intra-edge* (resp. *interedge*) if its ends belong to the same black cycle (resp. distinct black cycles). A BG-graph is called *simple* if it contains a single black cycle of even length and *paired* if it contains exactly two black cycles of odd length. A BG-graph is called *primitive* if its black-gray connected components (that are BG-graphs by themselves) are either simple components or paired components with a single interedge. Black cycles and intra-edges can be viewed as circles and chords on a plane. A BG-graph is called *non-crossing* if its intra-edges do not cross as chords within black circles (Fig. 6b). For a BG-graph B, let $c_{max}(B)$ be the number of cycles in a maximal black-gray cycle decomposition of B.

Theorem 4.1 ([2]) *For any BG-graph B, $c_{max}(B) \leq \frac{|B|}{2} + b_e(B)$, where $|B|$ is the number of vertices in B and $b_e(B)$ is the number of even black cycles in B. Moreover, if the graph B is primitive and non-crossing then $c_{max}(B) = |B|/2 + b_e(B)$.*

The upper bound on $c_{max}(B)$ in Theorem 4.1 depends only on black edges and does not depend on gray edges in B. Hence, for contracted breakpoint graphs Theorem 4.1 implies the following theorem:

Theorem 4.2 ([2]) *For a duplicated genome P and a perfect duplicated genome Q, $c_{max}(P, Q) \leq \frac{|P|}{2} + b_e(P)$, where $b_e(P) = b_e(\hat{P})$ is the number of even black cycles in the de Bruijn graph $\hat{P}$. Moreover, if the contracted breakpoint graph $G'(P, Q)$ is primitive and non-crossing then $c_{max}(P, Q) = |P|/2 + b_e(P)$.*

A maximum black-gray cycle decomposition of a primitive non-crossing contracted breakpoint graph can be informally represented as "splitting" the graph placed on a plane along double gray edges as shown in Fig. 6c. The following theorem provides a solution to the Cycle Decomposition Problem for multichromosomal genomes:

Theorem 4.3 *For any duplicated genome P, there exists a perfect duplicated genome Q with $c_{max}(P, Q) = |P|/2 + b_e(P)$.*

Proof If $\hat{P}$ contains some odd black cycles then we group them into pairs (formed arbitrarily), and introduce an arbitrary interedge connecting the cycles in each pair. We complete each black cycle with an arbitrary non-crossing gray matching. The resulting graph is a contracted breakpoint graph $G'(P, Q)$ of P and some perfect duplicated genome Q defined (not uniquely) by the double gray-obverse cycles. Since $G'(P, Q)$ is primitive and non-crossing, Theorem 4.2 implies that $c_{max}(P, Q) = |P|/2 + b_e(P)$. □

To solve the 2-Break Genome Halving Problem for a multichromosomal genome P we first find a perfect duplicated genome Q satisfying Theorem 4.3. Then applying Theorem 3.3 to a maximum black-gray cycle decomposition of $G'(P, Q)$ we get a labelling of P and some genome Q' (Q' is equivalent to Q) for which $c(P, Q') = c_{max}(P, Q') = |P|/2 + b_e(P)$. It follows from Lemma 3.1 that the genome Q' is a perfect duplicated genome. Theorem 4.2 guarantees that the decomposition of $G(P, Q')$ into $|P|/2 + b_e(P)$ black-gray cycles represents a maximal cycle decomposition, while Theorem 2.1 implies that it corresponds to the minimum 2-break distance between P and Q'. Therefore, the perfect duplicated genome Q' is a solution of the 2-Break Genome Halving Problem for the genome P.

5 3-Break Genome Halving Problem

Let P be a genome represented as a collection of black-obverse cycles. Given 3 black edges forming a matching on 6 vertices, define a 3-*break* as replacement of these edges with a set of 3 edges forming another matching on the same 6 vertices.

While 2-breaks correspond to standard rearrangements, 3-breaks add transposition-like operations (transpositions and inverted transpositions) as well as 3-way fissions to the set of rearrangements (Fig. 7). In difference from standard rearrangements (modelled as 2-breaks), transpositions introduce 3 breaks in the genome making them notoriously difficult to analyze. Computing the minimum number of transpositions transforming one genome into another is called *sorting by transpositions*. After Bafna and Pevzner, 1995 [6] gave a first 1.5-approximation algorithm for sorting by transpositions, a number of faster algorithms with the same approximation ratio were proposed [14, 52, 28] culminating in a recent 1.375-approximation algorithm by Elias and Hartman [23]. A number of researchers considered transpositions in conjunction with other rearrangement operations [24, 29, 34, 35, 44, 51, 4]. The complexity of sorting by transpositions remains unknown.

Theorem 5.1 ([1]) *The 3-break distance between a black matching P and a gray matching Q is $d_3(P, Q) = \frac{|P| - c^{odd}(P,Q)}{2}$ where $c^{odd}(P, Q)$ is the number of odd black-gray cycles in $G(P, Q)$ (i.e., cycles with odd number of black edges).*

In this section we solve the 3-Break Genome Halving Problem for a duplicated genome P by minimizing the 3-break distance $d_3(P, Q)$ over all perfect duplicated genomes Q. Let $c^{odd}_{max}(B)$ be the maximum number of odd black-gray cycles in a cycle decomposition among all cycle decompositions of a BG-graph B. Theorems 5.1 and 3.3 suggest the following reformulation of the 3-Break Genome Halving Problem.

Odd Cycle Decomposition Problem. Given a duplicated genome P, find a perfect duplicated genome Q maximizing $c^{odd}_{max}(G'(P, Q))$.

Below we use $c^{odd}_{max}(P, Q)$ as a shortcut for $c^{odd}_{max}(G'(P, Q))$. For a BG-graph B, we define $U(B)$ as

$$U(B) = \frac{|B|}{2} + b_2(B) - \frac{|b_1(B) - b_3(B)|}{2}$$

where $|B|$ is the number of black edges in B and $b_i(B)$ is the number of black cycles of length i modulo 4 in

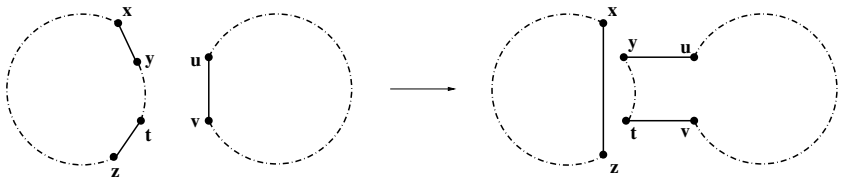

Figure 7: An example of a 3-break on edges (u, v), (x, y) and (z, t) corresponding to transposition of a segment $y \dots t$ from one chromosome to another. A *transposition* cuts off a segment of one chromosome and inserts it into the same or another chromosome. A transposition of a segment $\pi_i\pi_{i+1}\dots\pi_j$ of a chromosome $\pi_1\pi_2\dots\pi_i\pi_{i+1}\dots\pi_j\dots\pi_k\pi_{k+1}\dots\pi_m$ into a position k of the same chromosome results a chromosome $\pi_1\pi_2\dots\pi_{i-1}\pi_{j+1}\dots\pi_k\pi_i\pi_{i+1}\dots\pi_j\pi_{k+1}\dots\pi_m$. For chromosomes $\pi = \pi_1\pi_2\dots\pi_i\pi_{i+1}\dots\pi_j\dots\pi_m$ and $\sigma = \sigma_1\sigma_2\dots\sigma_n$ a transposition of a segment $\pi_i\pi_{i+1}\dots\pi_j$ of chromosome π into a position k in the chromosome σ results in chromosomes $\pi_1\pi_2\dots\pi_{i-1}\pi_{j+1}\pi_{j+2}\dots\pi_m$ and $\sigma_1\sigma_2\dots\sigma_{k-1}\pi_i\pi_{i+1}\dots\pi_j\sigma_k\dots\sigma_n$.

B. Later in this section (Theorem 5.3) we will show that for any BG-graph B, $c^{odd}_{max}(B) \le U(B)$. Since $U(B)$ depends only on black edges in B (i.e., only on the genome P if $B = G'(P, Q)$), this inequality implies that for any perfect duplicated genome Q, $c^{odd}_{max}(P, Q) \le \frac{|\hat{P}|}{2} + b_2(\hat{P}) - \frac{|b_1(\hat{P}) - b_3(\hat{P})|}{2}$. The following theorem shows how to achieve this upper bound.

Theorem 5.2 *Given a duplicated genome P, there exists a perfect duplicated genome Q with*

$$c^{odd}_{max}(P, Q) = \frac{|\hat{P}|}{2} + b_2(\hat{P}) - \frac{|b_1(\hat{P}) - b_3(\hat{P})|}{2}.$$

Proof Genome P defines the set of black cycles in the de Bruijn graph $\hat{P}$. We will complete $\hat{P}$ with a gray matching to obtain the BG-graph $G'(P, Q)$.

First we pair every black cycle of length 1 modulo 4 with a cycle of length 3 modulo 4 (if possible) resulting in $\min\{b_1(\hat{P}), b_3(\hat{P})\}$ such pairs. The remaining $|b_1(\hat{P}) - b_3(\hat{P})|$ odd black cycles form $\frac{|b_1(\hat{P}) - b_3(\hat{P})|}{2}$ pairs arbitrarily. For each pair of odd black cycles, we introduce an arbitrary gray interedge connecting them.

For each even black cycle $(v_1, v_2, \dots, v_{2n})$, we add n gray edges (v_1, v_2), (v_3, v_4), …, (v_{2n-1}, v_{2n}) as shown in Fig. 8a. For each odd black cycle $(v_1, v_2, \dots, v_{2n}, v_{2n+1})$ (where v_{2n+1} is incident to an interedge), we add n gray edges (v_1, v_2), (v_3, v_4), …, (v_{2n-1}, v_{2n}) as shown in Fig. 8b. We remark that n gray edges (v_1, v_2), (v_3, v_4), …, (v_{2n-1}, v_{2n}) form n trivial cycles with black edges of the cycle. By Theorem 3.2 the resulting graph is a contracted breakpoint graph $G'(P, Q)$ for some perfect duplicated genome Q. Below we show that there exists a black-gray cycle decomposition C of the graph $G'(P, Q)$ with $c^{odd}(C) = \frac{|\hat{P}|}{2} + b_2(\hat{P}) - \frac{|b_1(\hat{P}) - b_3(\hat{P})|}{2}$ cycles.

We construct the black-gray cycle decomposition of the resulting BG-graph as follows. We decompose every even black cycle c on vertices $(v_1, v_2, \dots, v_{2n})$ into n trivial black-gray cycles (with edges (v_1, v_2), (v_3, v_4), …, (v_{2n-1}, v_{2n})) and one more cycle on the remaining n black edges (Fig. 8a). This cycle is odd iff $n = |c|/2$ is odd. Therefore, every even cycle c corresponds either to $|c|/2$ odd cycles (if $|c| = 0$ modulo 4) or to $|c|/2 + 1$ odd cycles (if $|c| = 2$ modulo 4).

Similarly, each paired component p formed by odd cycles $(v_1, v_2, \dots, v_{2n+1})$ and $(w_1, w_2, \dots, w_{2m+1})$ can be decomposed into $n+m$ trivial black-gray cycles (formed by edges (v_1, v_2), (v_3, v_4), …, (v_{2n-1}, v_{2n}) and (w_1, w_2), (w_3, w_4), …, (w_{2m-1}, w_{2m})) and one more "large" cycle on the remaining $n + m + 2$ black edges of the component (Fig. 8b). This "large" cycle is odd iff $n+m+2 = |p|/2+1$ is odd. Therefore, every paired component p corresponds either to $|p|/2$ odd cycles (if $|p| = 0$ modulo 4) or to $|p|/2 - 1$ odd cycles (if $|p| = 2$ modulo 4).

Therefore, each component with n black edges is decomposed into $n/2$ odd cycles unless it is an even cycle of length 2 modulo 4 (in this case it is one odd cycle more) or a paired component of size 2 modulo 4 (in this case it is one odd cycle less). Summing over all connected components we get $\frac{|\hat{P}|}{2} + b_2(\hat{P}) - \frac{|b_1(\hat{P}) - b_3(\hat{P})|}{2}$ cycles. □

The rest of this section is devoted to the proof of the following theorem and the outline of the 3-Break Halving Algorithm.

Theorem 5.3 *For any BG-graph B, $c^{odd}_{max}(B) \le U(B)$.*

Proof We first give a sketch of the proof to provide an intuition for the follow up Lemmas 5.1-5.5.

We prove Theorem 5.3 by induction on the number of interedges $\imath(B)$ in B. Lemma 5.1 proves the base case of $\imath(B) = 0$. For any BG-graph B with $\imath(B) > 0$ and its black-gray cycle decomposition C with the maximum number of odd black-gray cycles (i.e., $c^{odd}(C) = c^{odd}_{max}(B)$) we show how to transform it into a BG-graph B' with a black-gray cycle decomposition C' such that $\imath(B') < \imath(B)$, $U(B') \le U(B)$, and $c^{odd}(C') \ge c^{odd}(C)$. Then by induction

$$c^{odd}_{max}(B) = c^{odd}(C) \le c^{odd}(C') \le U(B') \le U(B).$$

The construction of such a pair (B', C') breaks into two cases depending on whether every interedge in B is shared by two distinct odd cycles from C or not. The latter case is addressed by Lemmas 5.2 and 5.3, while the former case is addressed by Lemmas 5.4 and 5.5. □

Lemma 5.1 *For a simple BG-graph B, $c^{odd}_{max}(B) \le U(B)$.*

Proof For a simple BG-graph B, $b_1(B) = b_3(B) = 0$ and $U(B) = \frac{|B|}{2} + b_2(B)$. Theorem 4.1 implies $c^{odd}_{max}(B) \le c_{max}(B) \le |B|/2 + b_e(B) = |B|/2 + 1$. If $|B| = 2$ modulo 4 then $U(B) = |B|/2+1$ and $c^{odd}_{max}(B) \le U(B)$. If $|B| = 0$ modulo 4 then $U(B) = |B|/2$ while $c^{odd}_{max}(B) \le |B|/2 + 1$. However, in this case the inequality $c^{odd}_{max}(B) \le |B|/2 + 1$ is not tight since the overall number of odd cycles in every cycle decomposition of a simple BG-graph is even while $|B|/2 + 1$ is odd. Therefore, $c^{odd}_{max}(B) \le |B|/2 = U(B)$. □

Our proof of Theorem 5.3 is based on the notion of *e-transformations* introduced in [2]. For a double gray

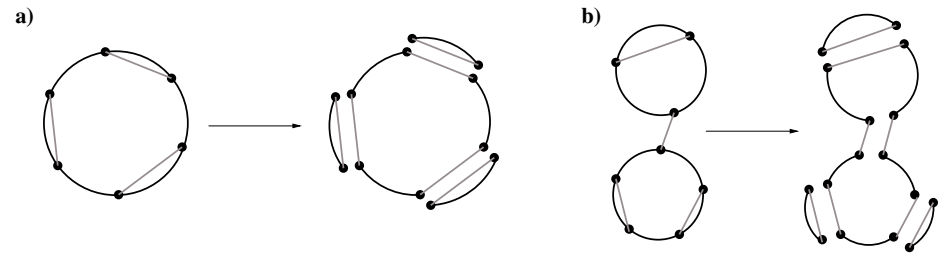

Figure 8: a) Cycle decomposition of a simple BG-graph on $2m$ vertices into m cycles of length 2 and one cycle of length $2m$; b) Cycle decomposition of a paired BG-graph on $2m$ vertices into $m-1$ cycles of length 2 and one cycle of length $2(m+1)$.

edge $e=(x,y)$ in B, e-transformation transforms the BG-graph B and its black-gray cycle decomposition C into a new BG-graph B^e with a black-gray cycle decomposition C^e as follows. Let c_1 and c_2 be two cycles (that may coincide) from C sharing the double gray edge $e=(x,y)$ and suppose that the cycle c_1 (resp. c_2) passes through the vertices u,x,y,v (resp. t,x,y,w) in a row. Then the BG-graph B^e is defined as the BG-graph B with the vertices x,y and all the incident edges replaced with two new black edges (u,v) and (w,t) (Fig. 9). A black-gray cycle decomposition C^e of the graph B^e is obtained from C by replacing u,x,y,v in the cycle c_1 with a single black edge (u,v) and replacing t,x,y,w in the cycle c_2 with a single black edge (w,t).

Note that if e is an interedge then e-transformation eliminates the interedge e and "merges" two black cycles in B into a single cycle in B^e (Fig. 9). Since such merging cannot create new interedges, we have $\imath(B^e) \leq \imath(B)-1$. In the following two Lemmas we study how e-transformations affect the parameters $U(B)$ and $c^{odd}(C)$.

Lemma 5.2 *If e is an interedge in B then e-transformation does not increase $U(B)$, i.e, $U(B^e) \leq U(B)$.*

Proof Every e-transformation reduces $|B|$ by two and may change each of the expressions $b_2(B)$ and $\frac{|b_1(B)-b_3(B)|}{2}$ by at most 1. However, if e-transformation increases $b_2(B)$ by 1 then it cannot change $\frac{|b_1(B)-b_3(B)|}{2}$ thus implying that $U(B^e)=\frac{|B^e|}{2}+b_2(B^e)-\frac{|b_1(B^e)-b_3(B^e)|}{2} \leq \frac{|B|-2}{2}+(b_2(B)+1)-\frac{|b_1(B)-b_3(B)|}{2}=U(B)$. Indeed, if $b_2(B)$ increases (i.e., $b_2(B^e)=b_2(B)+1$) then e-transformation creates a new cycle of length 2 modulo 4 implying that the interedge e connects black cycles whose lengths sum up to 0 modulo 4. If their lengths are 1 and 3 modulo 4 then $\frac{|b_1(B)-b_3(B)|}{2}$ does not change, and if they both are of even length then $\frac{|b_1(B)-b_3(B)|}{2}$ does not change either. □

Lemma 5.3 *Let C be a cycle decomposition of a BG-graph B and e be an interedge shared by cycles c_1 and c_2 from C. Then e-transformation does not reduce $c^{odd}(C)$ (i.e., $c^{odd}(C^e) \geq c^{odd}(C)$) unless c_1 and c_2 are two distinct odd cycles (in this case $c^{odd}(C^e)=c^{odd}(C)-2$).*

Proof If $c_1=c_2$ (i.e., cycles c_1 and c_2 are the same) then e-transformation simply reduces the number of black edges in this cycle by 2, i.e., $c^{odd}(C^e)=c^{odd}(C)$. If c_1 and c_2 are distinct cycles then e-transformation reduces the number of black edges in each of these cycles by 1. Therefore, $c^{odd}(C)$ may reduce by 2, increase by 2 or remain the same. The reduction happens only if both c_1 and c_2 are odd cycles. □

As soon as there is an interedge e in a BG-graph B that does not belong to two distinct odd cycles in a black-gray cycle decomposition C, Lemmas 5.2 and 5.3 allow one to perform the induction step in the proof of Theorem 5.3. To analyze cycle decompositions with every interedge shared by two distinct odd cycles, we introduce (e,g)-*transformations* of BG-graphs that replace a pair of gray edges $e=(x,y)$ and $g=(z,t)$ belonging to the same cycle c from C with a pair of gray edges (y,z) and (x,t). There may be up to two more cycles in C containing the gray edges e and g: c_e (cycle containing e) and c_g (cycle containing g). The (e,g)-transformation splits cycle c and transforms cycles c_e and c_g as follows.

If $c_e \neq c_g$, cycles c_e and c_g are merged into a single cycle (Fig. 10a). If $c_e=c_g$ then there are two possibilities (Fig. 10b,c) depending on how c_e traverses edges e and g: either as $(\ldots,x,y,\ldots,z,t)$ or as $(\ldots,x,y,\ldots,t,z)$. In the former case c_e is split into two cycles (Fig. 10b) while in the latter case it is rearranged (Fig. 10c).

In summary, (e,g)-transformation $(B,C) \rightarrow (B^{(e,g)},C^{(e,g)})$, may either merge cycles c_e and c_g (if $c_e \neq c_g$), or rearrange/split them (if $c_e=c_g$). Note that since B and $B^{(e,g)}$ have the same black subgraph, $U(B^{(e,g)})=U(B)$.

Lemma 5.4 *Let C be a black-gray cycle decomposition of a BG-graph B where a double gray edge e is shared by two distinct even black-gray cycles. Then there exists an (e,g)-transformation with $U(B^{(e,g)})=U(B)$, $c^{odd}(C^{(e,g)}) \geq c^{odd}(C)+2$, and $\imath(B^{(e,g)}) \leq \imath(B)$.*

Proof Let c and c_e be two even black-gray cycles in C containing double gray edge $e=(x,y)$. Let $g=(z,t)$ be the "next" double gray edge in c after e (i.e., (y,z) form a black edge in B) and c_g be a cycle in C sharing the edge g with c. Consider an (e,g)-transformation $(B,C) \longrightarrow (B^{(e,g)},C^{(e,g)})$.

Note that among two new double gray edges (x,t) and (y,z) only the double gray edge (x,t) may be an interedge in $B^{(e,g)}$. Moreover, (x,t) is an interedge in $B^{(e,g)}$ iff either (x,y) or (z,t) is an interedge in B. Therefore, $\imath(B^{(e,g)}) \leq \imath(B)$.

Note that (e,g)-transformation splits c into two odd cycles increasing the number of odd cycles by 2. Note also that (e,g)-transformation either merges c_e and c_g into a cycle c_e+c_g (in case $c_e \neq c_g$) or splits/rearranges c_e (in case $c_e=c_g$). Since c_e is even then in the former case c_e+c_g is odd iff c_g is odd while in the latter case the number of odd cycle can only increase. Therefore, $c^{odd}(C^{(e,g)}) \geq c^{odd}(C)+2$. □

Lemma 5.5 *Let B be a BG-graph with $\imath(B)>0$ and C be its black-gray cycle decomposition such that every*

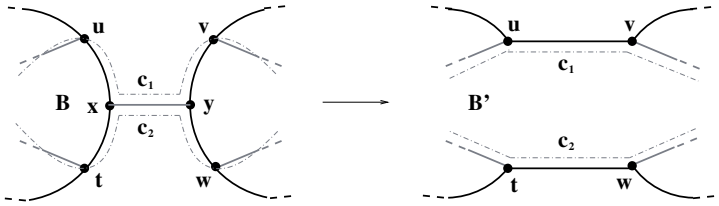

Figure 9: e-transformation of BG-graph B into a BG-graph B' where $e = (x, y)$ is a double gray edge passed through by two cycles c_1 and c_2.

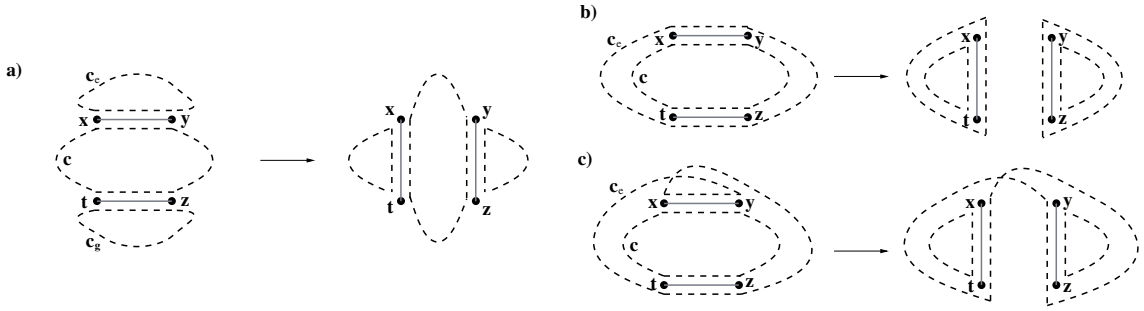

Figure 10: Three types of (e, g)-transformations operating on double gray edges $e = (x, t)$ and $g = (z, t)$ of cycle c: a) cycles c_e and c_g are different; b) cycle $c_e = c_g$ traverses vertices x, y, z, t as $(\ldots, x, y, \ldots, z, t)$; c) cycle $c_e = c_g$ traverses vertices x, y, z, t as $(\ldots, x, y, \ldots, t, z)$.

interedge is shared by two distinct odd cycles from C. Then there exist a BG-graph B' with black-gray cycle decomposition C' such that $U(B') \leq U(B)$, $c^{odd}(C') \geq c^{odd}(C)$, and $\imath(B') < \imath(B)$.

Proof Let $e = (x, y)$ be an interedge in B and c be an odd black-gray cycles from C passing through e. Cycle c has at least two interedges and let $g = (z, t)$ be the "next" interedge in cycle c after e (i.e., there is no other interedges between y and z while travelling along c, implying that y and z belong to the same black cycle in B). Note that since e is shared by two distinct odd cycles, $g \neq e$.

Consider an (e, g)-transformation $(B, C) \longrightarrow (B^{(e,g)}, C^{(e,g)})$ that replaces (x, y) and (z, t) with (y, z) and (x, t). This transformation removes two interedges from B and introduce at most one interedge (since (y, z) is not an interedge in $B^{(e,g)}$), implying $\imath(B^{(e,g)}) < \imath(B)$.

The (e, g)-transformation splits the odd cycle c into an even cycle (which we denote c') and an odd cycle. Hence, such splitting does not affect the number of odd cycles. We now analyze how the (e, g)-transformation affects odd cycles c_e and c_g (that are different from c) passing through edges e and g, correspondingly.

If $c_e = c_g$ then the (e, g)-transformation either rearranges this odd cycle (preserving the length) or splits it into two. In either case, that does not affect the number of odd cycles, i.e., $c^{odd}(C^{(e,g)}) = c^{odd}(C)$. Therefore, letting $B' = B^{(e,g)}$ and $C' = C^{(e,g)}$ proves the theorem.

If $c_e \neq c_g$ then these odd cycles are merged into an even cycle c'' in $C^{(e,g)}$ implying that $c^{odd}(C^{(e,g)}) = c^{odd}(C) - 2$. But in this case the even cycles c' and c'' share a double gray edge (i.e., either (x, t) or (y, z)). By Lemma 5.4 there exist a BG-graph B' and its black-gray cycle decomposition C' such that $c^{odd}(C') \geq c^{odd}(C^{(e,g)}) + 2 = c^{odd}(C)$, $U(B') = U(B)$, and $\imath(B') < \imath(B)$. □

This completes the proof of Theorem 5.3. We now outline the linear-time 3-Break Genome Halving Algorithm:

1. For a given duplicated genome P, find a perfect duplicated genome Q such that $c^{odd}_{max}(P, Q) = |P|/2 + b_2(\hat{P}) - \frac{|b_1(\hat{P}) - b_3(\hat{P})|}{2}$ and a maximum black-gray cycle decomposition C of the graph $G'(P, Q)$ (Theorem 5.2).
2. Find a labelling of the genomes P and Q' (Q' is equivalent to Q) and a breakpoint graph $G(P, Q')$ inducing C (Theorem 3.3). Output Q' as a solution of the 3-Break Genome Halving Problem.

6 A Flaw in El-Mabrouk–Sankoff Analysis

El-Mabrouk and Sankoff came up with a theorem describing the minimum distance from the given rearranged duplicated genome to a perfect duplicated genome. Given a rearranged duplicated genome P, the crux of their approach is an algorithm for computing $c(G)$ – the number of cycles of so-called *maximal completed graph*, i.e., a breakpoint graph[6] with the maximum number of black-gray cycles. In [22] they demonstrate that $c(G)$ equals the number of genes plus $\gamma(G)$ where $\gamma(G)$ is the parameter defined below. We illustrate the concepts from [22] using the genome $P = +a + b - c + b - d - e + a + c - d - e$ on the set of genes $\mathcal{B} = \{a, b, c, d, e\}$ (p. 757 in [22]). El-Mabrouk and Sankoff first arbitrarily label two copies of each gene x as x_1 and x_2 for each $x \in \mathcal{B}$ and further transform the signed permutation G into an unsigned permutation $a_1^t a_1^h b_1^t b_1^h c_1^h c_1^t b_2^t b_2^h d_1^h d_1^t e_1^h e_1^t a_2^t a_2^h c_2^t c_2^h d_2^h d_2^t e_2^h e_2^t$.

Let $\mathbf{V} = \{x_1^t, x_1^h, x_2^t, x_2^h \mid x \in \mathcal{B}\}$. The *partial graph* $\mathcal{G}(\mathbf{V}, A)$ associated with P has the edge set A of black edges linking adjacent term (other than obverses x_i^t and x_i^h) in the corresponding unsigned permutation (Fig. 11a).

Black edges together with counterpart edges (i.e., edges between x_1^t and x_2^t or between x_1^h and x_2^h) form a graph shown in Fig. 11b. The connected components of

[6] Following El-Mabrouk and Sankoff [22], we ignore obverse edges in breakpoint graphs throughout Section 6.

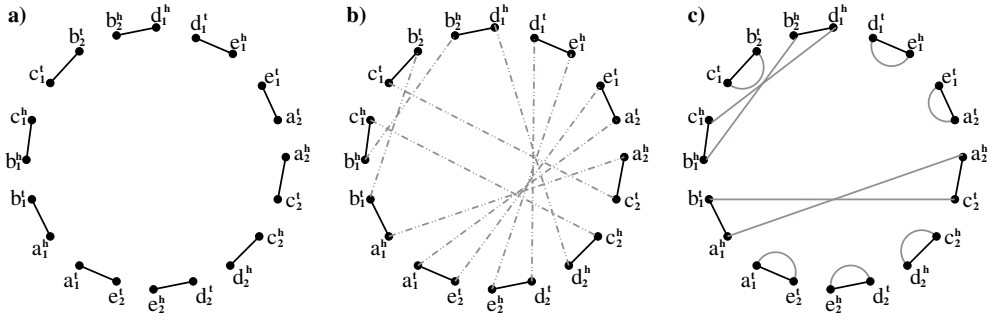

Figure 11: a) A set of black edges forming the partial graph $\mathcal{G}(\mathbf{V}, A)$ corresponding to the genome $P = +a + b - c + b - d - e + a + c - d - e$; b) Natural graphs as connected components in the partial graph with counterpart edges; c) A completed graph $\mathcal{G}(\mathbf{V}, A, \Gamma)$ with maximum number of cycles $c(G) = 8$. $\mathcal{G}(\mathbf{V}, A, \Gamma)$ is a breakpoint graph of the circular genome $P = +a_1 + b_1 - c_1 + b_2 - d_1 - e_1 + a_2 + c_2 - d_2 - e_2$ and a perfect duplicated genome $(-a_1 + e_2 + d_2 - c_2 + b_1)(-b_2 + c_1 - d_1 - e_1 + a_2)$ (of the form $R \ominus R$).

this graph are called *natural graphs* in [22]. There are four connected components (natural graphs) in the graph in Fig. 11b, two of them have 3 black edges (odd natural graphs) and two of them have 2 black edges (even natural graphs). Let NE be the number of even natural graphs ($NE = 2$ in Fig. 11b).

El-Mabrouk and Sankoff define the parameter

$$\gamma(G) = \begin{cases} NE, & \text{if all natural graphs are even} \\ NE + 1, & \text{otherwise} \end{cases}$$

A graph $\mathcal{G}(\mathbf{V}, A, \Gamma)$ obtained from the partial graph $\mathcal{G}(\mathbf{V}, A)$ by introducing a set of gray edges Γ is called a *completed graph* if $\mathcal{G}(\mathbf{V}, A, \Gamma)$ is a breakpoint graph for some genomes on the set of genes $\{x_1, x_2 \mid x \in \mathcal{B}\}$. The following theorem (Theorem 7.7 in [22]) characterizes the maximum number of cycles in the completed graph $\mathcal{G}(\mathbf{V}, A, \Gamma)$.

Theorem. The maximal number of cycles in a completed graph of $\mathcal{G}(\mathbf{V}, A)$ is $c(G) = \frac{|A|}{2} + \gamma(G)$.

For the genome in Fig. 11 we have $\gamma(G) = NE + 1 = 3$ and $c(G) = \frac{|A|}{2} + \gamma(G) = \frac{10}{2} + 3 = 8$. A completed graph $\mathcal{G}(\mathbf{V}, A, \Gamma)$ with 8 cycles is shown at Fig. 11c.[7] Below we provide a counterexample to Theorem 7.7 from [22].

Consider a circular genome $P = +a + b - a - b$ labelled as $+a_1 + b_1 - a_2 - b_2$. The genome P defines a partial graph $\mathcal{G}(\mathbf{V}, A)$ with a single natural graph of even size, implying $\gamma(G) = 1$. It follows from Theorem 7.7 in [22] that there exists a perfect duplicated genome Q such that the breakpoint graph $G = G(P, Q)$ consists of $\frac{|A|}{2} + \gamma(G) = 3$ cycles. However, the direct enumeration of all possible perfect duplicated genomes Q shows that there is no breakpoint graph $G(P, Q)$ with 3 cycles. There exist eight distinct labelled perfect duplicated genomes Q giving rise to eight breakpoint graphs $G(P, Q)$ shown in Fig. 12. All of them have less than 3 cycles. In the next section we explain what particular property of the genome $+a + b - a - b$ was not addressed properly in the El-Mabrouk–Sankoff analysis.

[7] While we do not explicitly consider $R \ominus R$ duplications shown in this Figure (see [22] for details), our counterexample works for both $R \oplus R$ and $R \ominus R$ duplications.

7 Classification Of Unichromosomal Duplicated Genomes

To introduce a new combinatorial invariant of duplicated genomes, consider labellings of vertices in the cycle defined by the duplicated rearranged genome P with numbers 0 and 1 (Fig. 13b). Every such labelling induces a two-digit labelling of the genes (edges): a label of each gene is formed by the labels of the incident vertices (Fig. 13c). A 01-labelling of the vertices is called *consistent* if for every pair of identical genes in P the label of one copy is inversion of the other. If there exist consistent labellings of genome P, we define the *parity index* of P as the number of genes labelled "01" modulo 2. Below we prove that the parity index is well-defined, i.e., the parity index is the same for all consistent labellings of a genome. It turns out that the El-Mabrouk–Sankoff theorem fails on genomes with the parity index 0.

We first outline the differences between the Genome Halving Problems for unichromosomal and multichromosomal genomes (compare the following three theorems to Theorems 3.1, 3.3, and 4.3).

Theorem 7.1 ([2]) *A graph H with black, gray, and obverse edges is a contracted breakpoint graph for some unichromosomal duplicated genomes iff*

- *each vertex in H is incident to two black edges, two gray edges, and two parallel obverse edges;*
- *H is connected with respect to black and obverse edges (*black-obverse connected*);*
- *H is connected with respect to gray and obverse edges (*gray-obverse connected*).*

Theorem 7.2 ([2]) *Let P and $R \oplus R$ be unichromosomal duplicated genomes and C be a black-gray cycle decomposition of the contracted breakpoint graph $G'(P, R \oplus R)$. Then there exists some labelling of P and either $R \oplus R$ or $2R$ that induces the cycle decomposition C.*

Theorem 7.3 ([2]) *For any duplicated genome P, there exists a perfect duplicated genome $R \oplus R$ such that $c_{max}(P, R \oplus R) = |P|/2 + b_e(P)$, and each paired component of $G'(P, R \oplus R)$ contains a single interedge.*

Although the maximal black-gray cycle decomposition of $G'(P, R \oplus R)$ may correspond to a breakpoint graph $G(P, 2R)$ (Fig. 5), we will prove below that there exists a breakpoint graph $G(P, R \oplus R)$ having "almost" the same number of black-gray cycle as $G(P, 2R)$ (Fig. 5d). Later we will classify all the cases where there exists a labelled genome $R' \oplus R'$ such that $c(P, R' \oplus R') = c(P, 2R)$.

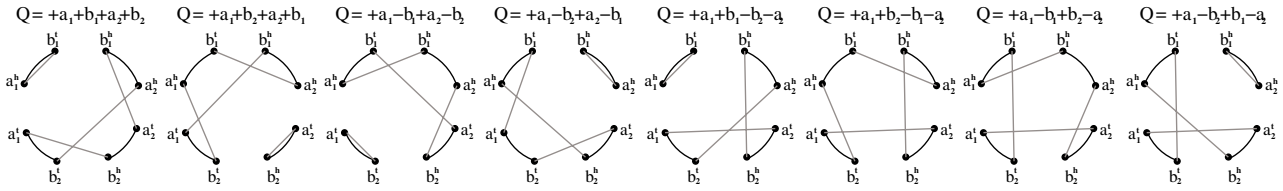

Figure 12: Breakpoint graphs of the circular genome $P = +a + b - a - b$ and all possible labellings of all possible perfect duplicated genomes Q (without loss of generality we assume that the labelling of P as $+a_1 + b_1 - a_2 - b_2$ is fixed). In terms of [22], the first four graphs correspond to $R \oplus R$ duplication pattern while the last four graphs correspond to to $R \ominus R$ duplication pattern.

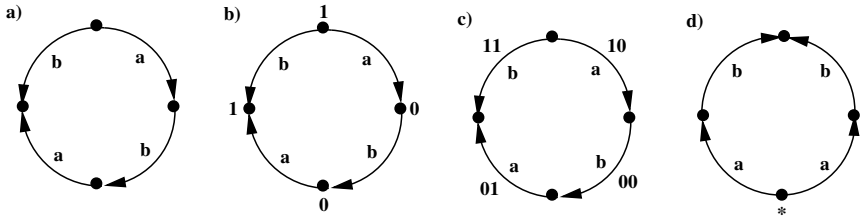

Figure 13: a) Circular genome $P = +a - b + a + b$ represented as a cycle with directed edges; b) 01-labelling of the vertices of the cycle defined by P; c) Induced labelling of the genes of P that is consistent; d) For some genomes consistent labellings do not exist: for genome $Q = +a + b - b - a$ the labels of both copies of gene a start with the same digit ("star") so they cannot be inversion of each other.

It is easy to see that $R \oplus R$ is equivalent to $2R$, implying that $G'(P, R \oplus R) = G'(P, 2R)$ for any duplicated genome P. But in difference from the breakpoint graph $G(P, R \oplus R)$ (for any labelling of P and $R \oplus R$) that contains a single gray-obverse cycle, the breakpoint graph $G(P, 2R)$ contains two gray-obverse cycles. The following theorem reveals the relationship between $G(P, R \oplus R)$ and $G(P, 2R)$.

Theorem 7.4 *For any labellings of the genomes P and $2R$, there exists a labelling of the genome $R \oplus R$ such that $|c(P, R \oplus R) - c(P, 2R)| \leq 1$. Moreover, if there are two gray edges (x, y) and $(\bar{x}, \bar{y})$ belonging to the same black-gray cycle in $G(P, 2R)$ then there exists a labelling of $R \oplus R$ with $c(P, R \oplus R) \geq c(P, 2R)$.*

Proof Let (x, y) be a gray edge in the breakpoint graph $G(P, 2R)$. Since the genome $2R$ is perfect duplicated there exists a gray edge $(\bar{x}, \bar{y})$ connecting counterparts of x and y. Define a graph H having the same vertices and edges as $G(P, 2R)$ except the gray edges (x, y) and $(\bar{x}, \bar{y})$ that are replaced with the gray edges $(x, \bar{y})$ and $(\bar{x}, y)$. Since the graph $G(P, 2R)$ consists of two gray-obverse cycles, the gray edges (x, y) and $(\bar{x}, \bar{y})$ belong to different gray-obverse cycles. Therefore, the graph H contains a single gray-obverse cycle (as well as a single black-obverse cycle inherited from $G(P, 2R)$). This implies that H is a breakpoint of the labelled genomes P and $R \oplus R$ (where the labelling of P is the same as in $G(P, 2R)$).

If the gray edges (x, y) and $(\bar{x}, \bar{y})$ belong to the same black-gray cycle in $G(P, 2R)$ then this cycle may be split into two in H while the other black-gray cycles are not affected. Conversely, if the gray edges (x, y) and $(\bar{x}, \bar{y})$ belong to different black-gray cycles in $G(P, 2R)$ then these cycles may be joined into a single cycle in H. In either case the difference $|c(P, R \oplus R) - c(P, 2R)|$ does not exceed 1. □

We re-define the notion of parity of a genome P in terms of the de Bruijn graph $\hat{P}$. A genome P is called *singular* if all black cycles in $\hat{P}$ are even. For a non-singular genome P, define $parity(P) = \infty$. For a singular genome P, we clockwise label edges of each black cycle in $\hat{P}$ with alternating numbers $\{0, 1\}$ so that every two adjacent edges are labelled differently (Fig. 14a). Labels of black edges in cycle P classify obverse edges in P into two classes: *even* if its flanking black edges have the same labels, and *odd* if its flanking black edges have different labels (Fig. 14b). Let m_{even} and m_{odd} be the number of even/odd obverse edges in P correspondingly. Obviously, both m_{even} and m_{odd} are even numbers. We define $parity(P) = m_{odd}/2 \bmod 2$.

This definition of the parity index coincides with the one given in the beginning of this section. To establish a correspondence between them one can consider a genome P as a black-obverse cycle and contract each black edge into a single vertex that inherits the label from the black edge. Since every pair of adjacent black edges of $\hat{P}$ is labelled differently, every pair of counterpart vertices is labelled differently as well. This implies that two-digit labels of every pair of obverse edges are inversions of each other.

Theorem 7.5 *The parity index of a singular genome is well defined.*

Proof Let P be a singular genome. If $\hat{P}$ has k black cycles then there are 2^k different 01-labellings of its black edges (two possible labellings per cycle). Therefore, it is sufficient to show that a change of 01-labelling of a particular black cycle c does not affect $parity(P)$.

Let m^c_{even} and m^c_{odd} be the number of even/odd obverse edges in cycle P connecting black edges of c with black edges outside c. Since double obverse edges form matching in the de Bruijn graph $\hat{P}$, the total number of double obverse edges connecting c with other black cycles is even and, thus, $m^c_{even} + m^c_{odd}$ is a multiple of 4.

Change of 01-labelling of the black cycle c reverses the labels $0 \leftrightarrow 1$ in c. Reversed labelling of c does not change parity of obverse edges connecting two black edges in c (since both endpoint labels change) or two black edges outside of c (since neither of endpoint labels change). At the same time, each of $m^c_{even} + m^c_{odd}$ obverse edges connecting black edges in c with black edges outside c changes its parity (i.e., even edges become odd and vice

versa). Then m_{odd} changes into m'_{odd} equal to:

$$m_{odd} - m^c_{odd} + m^c_{even} = m_{odd} - (m^c_{odd} + m^c_{even}) + 2m^c_{even}$$

Since both $m^c_{odd} + m^c_{even}$ and $2m^c_{even}$ are multiples of 4, the parity of $m'_{odd}/2$ and $m_{odd}/2$ is the same implying that *parity*(P) is well defined. □

Our goal is to prove the following theorem:

Theorem 7.6 *For a duplicated genome P,*

$$\max_R c(P, R \oplus R) = \begin{cases} |P|/2 + b_e(P), & \text{if parity}(P) \neq 0 \\ |P|/2 + b_e(P) - 1, & \text{otherwise} \end{cases}$$

The proof of Theorem 7.6 is split into two cases depending on whether P is singular or non-singular.

Theorem 7.7 *For a non-singular genome P,* $\max_R c(P, R \oplus R) = |P|/2 + b_e(P)$.

Proof If P is a non-singular genome then $\hat{P}$ has an odd black cycle. According to Theorem 7.3 there exists a perfect duplicated genome $R \oplus R$ such that $c_{max}(P, R \oplus R) = |P|/2 + b_e(P)$. Theorem 7.2 ensures that the maximum cycle decomposition of the contracted breakpoint graph $G'(P, R \oplus R)$ is induced by a labelling of either $R \oplus R$ or $2R$. If it is $R \oplus R$ then the theorem holds. Otherwise, consider a paired component in $G'(P, R \oplus R)$ (which exists since $\hat{P}$ has an odd black cycle) and a single interedge e in it (Theorem 7.3). Let (x, y) and $(\bar{x}, \bar{y})$ be gray edges in $G(P, 2R)$ corresponding to the interedge e in $G'(P, 2R) = G'(P, R \oplus R)$. Since e is the only bridge between two different black cycles (Theorem 7.3) in $G'(P, R \oplus R)$, the gray edges (x, y) and $(\bar{x}, \bar{y})$ must belong to the same black-gray cycle in $G(P, 2R)$. Applying Theorem 7.4 to these gray edges we obtain a labelled genome $R \oplus R$ with $c(P, R \oplus R) = c(P, 2R) = |P|/2 + b_e(P)$. □

For a singular genome P, we first fix some alternating 01-labelling of black edges in every black cycle of $\hat{P}$. The labelling of edges imposes a labelling of vertices of any breakpoint graph $G(P, Q)$ (for any genome Q) so that each vertex inherits a label from an incident black edge. Note that every pair of counterpart vertices get different labels as their incident black edges are adjacent in $\hat{P}$. A labelling of vertices of $G(P, Q)$ is called *uniform* if endpoints of every gray edge have identical labels (i.e, every gray edge is even). We will need the following theorem:

Theorem 7.8 *Let P be a singular genome and Q be a perfect duplicated genome with $c(P, Q) = |P|/2 + b_e(P)$. Then every alternating 01-labelling of $\hat{P}$ imposes a uniform labelling on vertices of $G(P, Q)$.*

While the definition of the breakpoint graph does not explicitly specify the counterpart edges, one can derive them for $G(P, Q)$ in Theorem 7.8 from the vertex labels. Also, it is easy to see that gray and counterpart edges in $G(P, Q)$ form cycles of length 4 as soon as Q is a perfect duplicated genome. We take a liberty to restate the condition $c(P, Q) = |P|/2 + b_e(P)$ as $c_{bg}(G) = n + c_{bc}(G)$, where $c_{bg}(G)$ is the number of black-gray edges in G, n is the number of unique genes in P and $c_{bc}(G)$ is the number of black-counterpart cycles in G. Also, every alternating 01-labelling of $\hat{P}$ corresponds to an alternating labelling of black edges within black-counterpart cycles. This leads to the following reformulation of Theorem 7.8:

Theorem 7.9 *Let H be a graph on $4n$ vertices consisting of three perfect matchings: black, gray, and counterpart such that (i) gray and counterpart matchings form cycles of length 4 and (ii) $c_{bg}(H) = n + c_{bc}(H)$. Then every alternating 01-labelling of black edges within black-counterpart cycles imposes a uniform labelling on vertices of H.*

Proof The proof is done by induction on n. If $n = 1$ then the graph H consists of a gray-counterpart cycle with two black edges parallel to the gray edges, and the theorem holds. Assume that the theorem holds for graphs with less than $4n$ vertices.

Since H has $2n$ black edges and $c_{bg}(H) = n + c_{bc}(H) > n$, the pigeonhole principle implies that there exists a trivial black-gray cycle c_1 in H. Let $e_1 = (x, y)$ be a gray edge in the cycle c_1 (thus, e_1 is even) and let (x, z) and (y, t) be adjacent counterpart edges. Then there is a gray edge $e_2 = (z, t)$ belonging to the same gray-counterpart cycle as e_1. Let c_2 be a black-gray cycle c_2 containing the gray edge e_2.

If the cycle c_2 is trivial, then the endpoints of e_2 have identical labels. In this case we define a new graph H' as the graph H without vertices x, y, z, t and all incident edges. It is easy to see that H' is a graph on $4(n-1)$ vertices satisfying the conditions of the theorem. Indeed, the number of black-gray cycles in H' is reduced by 2 and the number of black-counterpart cycles is reduced by 1 (as compared to H), i.e., $c_{bg}(H') = c_{bg}(H) - 2$ and $c_{bc}(H') = c_{bc}(H) - 1$. Therefore, $c_{bg}(H') = (n-1) + c_{bc}(H')$. By the induction assumption, every alternating 01-labelling of H' imposes a uniform labelling on vertices of H'. It implies that every alternating 01-labelling of H imposes a uniform labelling on vertices of H.

If the cycle c_2 is not trivial, let (u, z) and (t, v) be black edges adjacent to e_2. These black edges are neighbors of the black edge (x, y) on a black-counterpart cycle (passing through the vertices u, z, x, y, t, v), so they have the same label l which different from the label of (x, y). Therefore, the endpoints of the gray edge e_2 have identical labels. We define a new graph H' as the graph H with vertices x, y, z, t and all incident edges removed but with a single black edge (u, v) labelled l added (Fig. 14c). The graph H' has $4(n-1)$ vertices, $c_{bc}(H') = c_{bc}(H)$ black-counterpart cycles, and $c_{bg}(H') = c_{bg}(H) - 1$ black-gray cycles, thus, $c_{bg}(H') = n - 1 + c_{bc}(H')$ and the induction applies. □

To complete the proof of Theorem 7.6 we need one more theorem:

Theorem 7.10 *For a singular genome P and a perfect duplicated genome Q with $c(P, Q) = |P|/2 + b_e(P)$,*

- *$Q = R \oplus R$ iff parity$(P) = 1$;*
- *$Q = 2R$ iff parity$(P) = 0$.*

Proof According to Theorem 7.2, the graph $G(P, Q)$ has either a single gray-obverse cycle (case $Q = R \oplus R$) or two symmetric gray-obverse cycles (case $Q = 2R$). Theorem 7.8 implies that all gray edges in $G(P, Q)$ are even (i.e., have identically labelled endpoints) for every alternating 01-labelling of black edges of P.

Case 1: Graph $G(P, Q)$ has a single gray-obverse cycle c. Consider an arbitrary vertex v in $G(P, Q)$ and its counterpart $\overline{v}$. Vertices v and $\overline{v}$ break c into two paths: c' (from v to $\overline{v}$) and c'' (from $\overline{v}$ to v). For every path (cycle) c denote c_{odd} as the number of odd obverse edges in c. Note that obverse edges are evenly divided between c' and c'', i.e., for every pair of obverse edges connecting counterpart vertices, one edge belongs to c' and the other

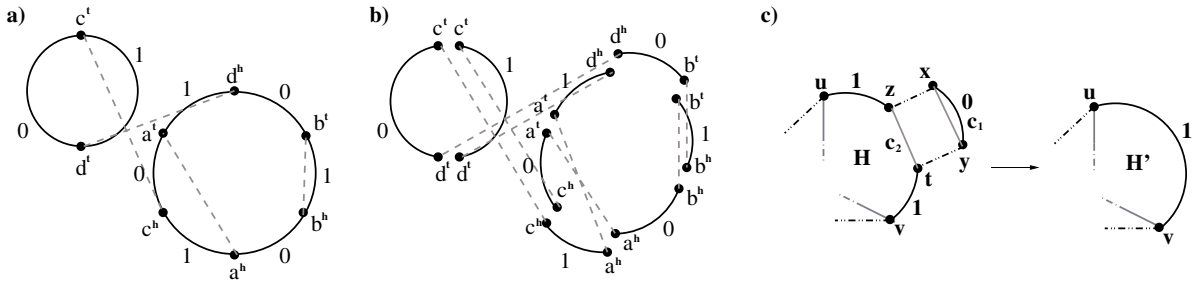

Figure 14: For the genome $P = +a - b - b - d + c - a - d + c$, a) 01-labelling of the de Bruijn graph $\hat{P}$; b) induced labelling of black-obverse cycle P with $m_{odd} = 4$ and $m_{even} = 4$; c) transformation of the graph H into H' by removing vertices x, y, z, t and incident edges and adding a black edge (u, v) labeled the same as (u, x) and (v, t).

edge belongs to c''. Therefore, $c'_{odd} = c''_{odd}$. Note that start (vertex v) and end (vertex $\overline{v}$) vertices of path c' are labelled differently. Since the total number of odd edges is odd for every path with differently labelled ends and since all gray edges are even (Theorem 7.8), the total number of odd obverse edges in the path c' is odd. Therefore, $c_{odd}/2 = c'_{odd}$ is odd implying that $parity(P) = 1$.

Case 2: Graph $G(P, Q)$ has two gray-obverse cycles c' and c''. Note that obverse edges are evenly divided between c' and c'', i.e., for every pair of obverse edges connecting counterpart vertices, one edge belongs to c' and the other edge belongs to c''. Therefore, $c'_{odd} = c''_{odd}$. Since the total number of odd edges in every cycle is even and since all gray edges are even (Theorem 7.8), the total number of odd obverse edges in every cycle is even. Since c'_{odd} is even, the overall number of odd obverse edges is a multiple of 4 implying that $parity(P) = 0$. □

For a singular genome P with $parity(P) = 1$ Theorem 7.10 implies Theorem 7.6 while for a singular genome P with $parity(P) = 0$ it implies that there is no genome R for which $c(P, R \oplus R) = |P|/2 + b_e(P)$. In the latter case, there exists a genome R and a labelling of P and $2R$ for which $c(P, 2R) = |P|/2 + b_e(P)$ (Theorem 7.3). The genome $2R$ can be transformed into a labelled genome $R \oplus R$ with $c(P, R \oplus R) = c(P, 2R) - 1 = |P|/2 + b_e(P) - 1$ (Theorem 7.4). This completes the proof of Theorem 7.6.

The classification of circular genomes leads to the following algorithm for Genome Halving Problem[8]:

1. For a given duplicated genome P, find a perfect duplicated genome $R \oplus R$ such that $c_{max}(P, R \oplus R) = |P|/2 + b_e(P)$ (Theorem 7.3) and decompose $G'(P, R \oplus R)$ into maximum number of black-gray cycles [2].
2. Find a labelling of the genomes P and Q ($Q = R \oplus R$ or $Q = 2R$) and a breakpoint graph $G(P, Q)$ inducing the maximum black-gray cycle decomposition of $G'(P, R \oplus R)$ (Theorem 7.2).
3. If $Q = R \oplus R$ then output the breakpoint graph $G(P, R \oplus R)$.
4. If $Q = 2R$ and P is non-singular then there is a paired component in $G'(P, R \oplus R)$ with a single interedge (Theorem 7.3) that corresponds to two gray edges (x, y) and $(\bar{x}, \bar{y})$ in $G(P, 2R)$. Find a labelling of the genome $R \oplus R$ for which $c(P, R \oplus R) = c(P, 2R)$ (Theorems 7.4 and 7.7) and output $G(P, R \oplus R)$.
5. If $Q = 2R$ and P is singular then $parity(P) = 0$ (Theorem 7.10). Find a labelling of the genome $R \oplus R$ for which $c(P, R \oplus R) = c(P, 2R) - 1$ (Theorem 7.4) and output $G(P, R \oplus R)$.

The first two steps of the Genome Halving algorithm can be implemented in $O(|P|^2)$ time (see [2]) while the remaining steps fit this time bound as well. In practice, our Genome Halving software takes less than a second to halve a "random" duplicated genome with 1000 unique genes with a standard Intel Pentium III 900MHz CPU.

[8] The algorithm below outputs the breakpoint graph $G(P, R \oplus R)$ (in addition to the pre-duplicated genome R). This allows one to reconstruct a sequence of reversals transforming $R \oplus R$ into P with the reversal distance algorithm.

Acknowledgements

We are grateful to Mohan Paturi and Dekel Tsur for many insightful comments.

References

[1] M. Alekseyev and P. Pevzner. Multi-Break Rearrangements and Breakpoint Re-use in Chromosomal Evolution. (unpublished manuscript).

[2] M. Alekseyev and P. Pevzner. Colored de Bruijn graphs and Genome Halving Problem. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 3, 2006. (in press).

[3] 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.*, 8:483–491, 2001.

[4] M. Bader and E. Ohlebusch. Sorting by weighted reversals, transpositions, and inverted transpositions. *Proceedings of the 10th Conference on Research in Computational Molecular Biology (RECOMB)*, pages 563–577, 2006.

[5] V. Bafna and P. A. Pevzner. Genome rearrangement and sorting by reversals. *SIAM Journal on Computing*, 25:272–289, 1996.

[6] V. Bafna and P. A. Pevzner. Sorting permutations by transpositions. *SIAM J. Discrete Math.*, 11:224–240, 1998.

[7] E. Belda, A. Moya, and F. J. Silva. Genome rearrangement distances and gene order phylogeny in γ-proteobacteria. *Mol. Biol. Evol.*, 22:1456–1467, 2005.

[8] A. Bergeron. A very elementary presentation of the Hannenhalli–Pevzner theory. *Lecture Notes in Computer Science*, 2089:106–117, 2001.

[9] A. Bergeron, J. Mixtacki, and J. Stoye. Reversal distance without hurdles and fortresses. *Lecture Notes in Computer Science*, 3109:388–399, 2004.

[10] G. Bourque, P. A. Pevzner, and G. Tesler. Reconstructing the genomic architecture of ancestral mammals: lessons from human, mouse, and rat genomes. *Genome Research*, 14:507–516, 2004.

[11] G. Bourque, Y. Yacef, and N. El-Mabrouk. Maximizing synteny blocks to identify ancestral homologs. *Lecture Notes in Bioinformatics*, 3678:21–34, 2005.

[12] G. Bourque1, E. M. Zdobnov, P. Bork, P. A. Pevzner, and G. Tesler. Comparative architectures of mammalian and chicken genomes reveal highly variable rates of genomic rearrangements across different lineages. *Genome Research*, 15:98–110, 2005.

[13] X. Chen, J. Zheng, P. Nan Z. Fu, Y. Zhong, S. Lonardi, and T. Jiang. Computing the assignment of orthologous genes via genome rearrangement. *Proceedings of Asia Pacific Bioinformatics Conference*, pages 363–378, 2005.

[14] D. A. Christie. *Genome Rearrangement Problems*. PhD thesis, University of Glasgow, 1999.

[15] A. Christoffels, E. G. L. Koh, J. Chia, S. Brenner, S. Aparicio, and B. Venkatesh. Fugu genome analysis provides evidence for a whole-genome duplication early during the evolution of ray-finned fishes. *Mol. Biol. Evol.*, 21(6):1146–1151, 2004.

[16] P. Dehal and J. L. Boore. Two rounds of genome duplication in the ancestral vertebrate genome. *PLoS Biology*, 3(10):e314, 2005.

[17] F. S. Dietrich et al. The *Ashbya gossypii* Genome as a Tool for Mapping the Ancient *Saccharomyces cerevisiae* Genome. *Science*, 304:304–307, 2004.

[18] N. El-Mabrouk. Genome Rearrangement by Reversals and Insertions/Deletions of Contiguous Segments. *Lecture Notes in Computer Science*, 1848:222–234, 2000.

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

[20] N. El-Mabrouk, J. H. Nadeau, and D. Sankoff. Genome Halving. *Lecture Notes in Computer Science*, 1448:235–250, 1998.

[21] N. El-Mabrouk and D. Sankoff. On the reconstruction of ancient doubled circular genomes. *Genome Informatics*, 10:83–93, 1999.

[22] N. El-Mabrouk and D. Sankoff. The Reconstruction of Doubled Genomes. *SIAM Journal on Computing*, 32:754–792, 2003.

[23] I. Elias and T. Hartman. A 1.375-Approximation Algorithm for Sorting by Transpositions. *Lecture Notes in Computer Science*, 3692:204–214, 2005.

[24] Q. P. Gu, S. Peng, and H. Sudborough. A 2-approximation algorithm for genome rearrangements by reversals and transpositions. *Theoret. Comput. Sci.*, 210:327–339, 1999.

[25] R. Guyot and B. Keller. Ancestral genome duplication in rice. *Genome*, 47:610–614, 2004.

[26] S. Hannenhalli and P. Pevzner. Transforming men into mouse (polynomial algorithm for genomic distance problem). *Proceedings of the 36th Annual Symposium on Foundations of Computer Science*, pages 581–592, 1995.

[27] S. Hannenhalli and P. Pevzner. Transforming cabbage into turnip (polynomial algorithm for sorting signed permutations by reversals). *Journal of the ACM*, 46:1–27, 1999.

[28] T. Hartman. A simpler 1.5-approximation algorithm for sorting by transpositions. *Lecture Notes in Computer Science*, 2676:156–169, 2003.

[29] T. Hartman and R. Sharan. A 1.5-approximation algorithm for sorting by transpositions and transreversals. *Lecture Notes in Computer Science*, 3240:50–61, 2004.

[30] O. Jaillon et al. Genome duplication in the teleost fish *Tetraodon nigroviridis* reveals the early vertebrate protokaryotype. *Nature*, 431:946–957, 2004.

[31] H. Kaplan, R. Shamir, and R.Tarjan. Faster and simpler algorithm for sorting signed permutations by reversals. *SIAM Journal on Computing*, 29:880–892, 1999.

[32] H. Kaplan and E. Verbin. Sorting signed permutations by reversals, revisited. *J. Comput. Syst. Sci.*, 70(3):321–341, 2005.

[33] M. Kellis, B. W. Birren, and E. S. Lander. Proof and evolutionary analysis of ancient genome duplication in the yeast *Saccharomyces cerevisiae*. *Nature*, 428:617–624, 2004.

[34] G. H. Lin and G. Xue. Signed genome rearrangements by reversals and transpositions: models and approximations. *Theoret. Comput. Sci.*, 259:513–531, 2001.

[35] Y. C. Lin, C. L. Lu, H. Y. Chang, and C. Y. Tang. An Efficient Algorithm for Sorting by Block-Interchanges and Its Application to the Evolution of Vibrio Species. *J. Comput. Biol.*, 12:102–112, 2005.

[36] A. Meyer and Y. Van de Peer. From 2R to 3R: evidence for a fish-specific genome duplication (FSGD). *BioEssays*, 27(9):937–945, 2005.

[37] W. J. Murphy, G. Bourque, G. Tesler, P. Pevzner, and S. J. O'Brien. Reconstructing the genomic architecture of mammalian ancestors using multispecies comparative maps. *Human Genomics*, 1:30–40, 2003.

[38] S. Ohno. *Evolution by gene duplication*. Springer, Berlin, 1970.

[39] M. Ozery-Flato and R. Shamir. Two Notes on Genome Rearrangement. *Journal of Bioinformatics and Computational Biology*, 1:71–94, 2003.

[40] P. Pevzner, H. Tang, and G. Tesler. De Novo Repeat Classification and Fragment Assembly. *Genome Research*, 14:1786–1796, 2004.

[41] P. Pevzner and G. Tesler. Genome Rearrangements in Mammalian Evolution: Lessons from Human and Mouse Genomes. *Genome Research*, 13:37–45, 2003.

[42] P. A. Pevzner. *Computational Molecular Biology: An Algorithmic Approach*. The MIT Press, Cambridge, 2000.

[43] P. A. Pevzner and G. Tesler. Human and mouse genomic sequences reveal extensive breakpoint reuse in mammalian evolution. *Proceedings of the National Academy of Sciences*, 100:7672–7677, 2003.

[44] A. J. Radcliffe, A. D. Scott, and E. L. Wilmer. Reversals and Transpositions Over Finite Alphabets. *SIAM J. Discrete Math.*, 19:224–244, 2005.

[45] M. Robinson-Rechavi, O. Marchand, H. Escriva, and V. Laudet. An ancestral whole-genome duplication may not have been responsible for the abundance of duplicated fish genes. *Curr Biol.*, 11(12):458–459, 2001.

[46] D. Sankoff. Genome rearrangement with gene families. *Bioinformatics*, 15:909–917, 1999.

[47] K. M. Swenson, M. Marron, J. V. Earnest-DeYoung, and B. M. E. Moret. Approximating the true evolutionary distance between two genomes. *Proc. 7th Workshop on Algorithm Engineering & Experiments (ALENEX)*, pages 121–129, 2005.

[48] K. M. Swenson, N. D. Pattengale, and B. M. E. Moret. A framework for orthology assignment from gene rearrangement data. *Lecture Notes in Bioinformatics*, 3678:153–166, 2005.

[49] E. Tannier and M. F. Sagot. Sorting by reversals in subquadratic time. *Lecture Notes in Computer Science*, pages 1–13, 2004.

[50] G. Tesler. Efficient algorithms for multichromosomal genome rearrangements. *J. Comput. Syst. Sci.*, 65:587–609, 2002.

[51] M. E. Walter, Z. Dias, and J. Meidanis. Reversal and transposition distance of linear chromosomes. *String Processing and Information Retrieval: A South American Symposium (SPIRE)*, pages 96–102, 1998.

[52] M. E. Walter, L. Reginaldo, A. F. Curado, and A. G. Oliveira. Working on the Problem of Sorting by Transpositions on Genome Rearrangements. *Lecture Notes in Computer Science*, 2676:372–383, 2003.

[53] S. Yancopoulos, O. Attie, and R. Friedberg. Efficient sorting of genomic permutations by translocation, inversion and block interchange. *Bioinformatics*, 21:3340–3346, 2005.