

Genome Halving by Block Interchange

Antoine Thomas, Aïda Ouangraoua, Jean-Stéphane Varré

▶ **To cite this version:**

Antoine Thomas, Aïda Ouangraoua, Jean-Stéphane Varré. Genome Halving by Block Interchange. BIOINFORMATICS, Feb 2012, Lisboa, Portugal. pp.58-65. hal-00749026

HAL Id: hal-00749026
https://hal.science/hal-00749026

Submitted on 6 Nov 2012

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Single Tandem Halving by Block Interchange

Antoine Thomas, Aïda Ouangraoua, and Jean-Stéphane Varré

LIFL, UMR 8022 CNRS, Université Lille 1
INRIA Lille, Villeneuve d'Ascq, France

Abstract. We address the problem of finding the minimal number of block interchanges required to transform a duplicated unilinear genome into a single tandem duplicated unilinear genome. We provide a formula for the distance as well as a polynomial time algorithm for the sorting problem. This is the revised and extended version of [8].

1 INTRODUCTION

Genomic rearrangements are known to play a central role in the evolutionary history of the species. Several operations act on the genome, shaping the sequence of genes. A number of rearrangement operations to sort a genome into another, and evaluate the evolutionary distance between genomes, have been studied: reversals, transpositions, translocations, block interchanges, fusions, fissions, and more recently Double-Cut-and-Join (DCJ). In this paper, we focus on the *block interchange* operation, that consists in exchanging two intervals of a genome.

Block interchanges scenarios have been studied for the first time by Christie [1]. He proposed a $O(n^2)$ time algorithm for computing the minimum number of block interchanges for transforming a linear chromosome with unique gene content into another one. Lin *et al.* [5] proposed later the best algorithm to date in $O(\gamma n)$ where γ is the minimum number of block interchanges required for the transformation. Yancopoulos *et al.* [10] introduced the DCJ operation which consist in cutting the genomes in two points and joining the four resulting extremities in a different way. Interestingly, they noticed that a block interchange can be simulated by two consecutive DCJ operations.

Another very important feature in genome evolution is that genomes often undergo genome duplication events, both segmental and whole-genome duplications. For instance, a *tandem-duplication event* is a segmental duplication that duplicates a genomic sequence and results in a segment made of two consecutive occurrences of the genomic sequence, called a *single tandem duplicated segment*, in the genome. Genome duplication events are followed by other rearrangements events which result in a scrambled genome. The *Genome Halving problem* introduced by El-Mabrouk *et al.* [2] consists in finding the sequence of rearrangement events that allow one to go back from the scrambled genome to the original duplicated one.

Genome Halving has been studied under several models: reversals [2], translocation/reversals [3], breakpoints [7]. Most of the results led to polynomial time

algorithms. Particularly, the Genome Halving by DCJ was studied in [9,6], and in [6] some useful data structures were presented leading to a linear time algorithm for the Genome Halving by DCJ. Following these results on the Genome Halving by DCJ, a natural problem to consider is the Genome Halving by block interchange. However, as block interchanges cannot split chromosomes, the original duplicated genome in this case would rather be a *single tandem duplicated genome*, ie. we consider that the cause of duplicated content was a tandem duplication event on the whole genome.

In this paper, we study the Single Tandem Halving by block interchange on a duplicated genomic segment resulting from a tandem-duplication event, followed by block interchange events that have scrambled the gene content of the segment. This duplicated genomic segment is represented as a linear chromosome with duplicated gene content w.l.o.g, and we search for a parsimonous scenario of block interchange operations transforming the linear chromosome into a linear tandem duplicated chromosome. We answer *yes* to the question: *Does there exist a parsimonious sequence of block interchange operations, such that, replacing each block interchange by two consecutive DCJ operations yields a parsimonious sequence of DCJ operations ?*. Based on the adequate data structure to represent potential DCJ operations and their overlapping relations, we derive a quadratic time algorithm for the Single Tandem Halving by block interchanges. Very recently, Kováč *et al.* [4] addressed the problem of reincorporating the temporary circular chromosomes induced by DCJs immediately after their creation considering the Genome Halving. This problem is obviously related to the problem addressed in the present paper, but the aim and results are different. We are interested in linear genomes, not in multilinear ones, and we focus on pure block interchange scenarios that can be simulated by particular types of DCJ operations called *excisions* and *integrations*, whereas Kováč *et al.* focused on general DCJ scenarios simulating reversals, translocation, fusion, fissions along with excisions, integrations, and block interchanges.

Section 2 gives definitions. In Section 3, we first give a lower bound on the distance with helpful properties for the rest of the paper. In Section 4, we prove the analytical formula for the distance. We conclude in Section 5 with a quadratic time and space algorithm to obtain a parsimonious scenario.

2 PRELIMINARIES: DUPLICATED GENOMES, REARRANGEMENT, GENOME HALVING PROBLEMS

In this section we give the main definitions and notations used in the paper.

2.1 Duplicated Genomes

A genome is composed of genomic markers organized in linear or circular chromosomes. A linear chromosome is represented by an ordered sequence of unsigned integers, each standing for a marker, surrounded by two abstract markers $\circ$ at

each end indicating the telomeres. A circular chromosome is represented by a circularly ordered sequence of unsigned integers representing markers. For example, $(1\ \ 2\ \ 3)$ $(\circ\ \ 4\ \ 5\ \ 6\ \ 7\ \ \circ)$ is a genome constituted of one circular and one linear chromosome. Note that all genomes are considered *unsigned* in this paper w.l.o.g, because block interchange operations do not modify the signs of markers.

Definition 1. *A* totally duplicated genome *is a genome in which each marker appears twice.*

In a totally duplicated genome, two copies of a same marker are called paralogs. We distinguish paralogs by denoting one marker by x and its paralog by $\overline{x}$. By convention $\overline{\overline{x}} = x$. For example, the following genome is a totally duplicated genome: $(\circ\ \ 1\ \ \overline{1}\ \ 3\ \ 2\ \ 4\ \ 5\ \ 6\ \ \overline{6}\ \ 7\ \ \overline{3}\ \ 8\ \ \overline{2}\ \ \overline{4}\ \ \overline{5}\ \ 9\ \ \overline{8}\ \ \overline{7}\ \ \overline{9}\ \ \circ)$.

An *adjacency* in a genome is a pair of consecutive markers. For example, the genome $(\circ\ \ 1\ \ 2\ \ \circ)$ $(3\ \ 4\ \ 5)$ has six adjacencies, $(\circ\ \ 1)$, $(1\ \ 2)$, $(2\ \ \circ)$, and $(3\ \ 4)$, $(4\ \ 5)$, $(5\ \ 3)$. The linear or circular order of the markers in a chromosome naturally induces an order on the adjacencies that we denote by $<$. For example in the previous genome the order induced on the adjacencies is: $(\circ\ \ 1) < (1\ \ 2) < (2\ \ \circ)$, and $(3\ \ 4) < (4\ \ 5) < (5\ \ 3) < (3\ \ 4)$.

A *double-adjacency* in a genome G is an adjacency $(a\ \ b)$ such that $(\overline{a}\ \ \overline{b})$ is an adjacency of G as well. Note that a genome always has an even number of double-adjacencies. For example, the four double-adjacencies in the following genome are indicated by dots :

$$G = (\circ\ \ 1\ \ \overline{1}\ \ 3\ \ 2\ \cdot\ 4\ \cdot\ 5\ \ 6\ \ \overline{6}\ \ 7\ \ \overline{3}\ \ 8\ \ \overline{2}\ \cdot\ \overline{4}\ \cdot\ \overline{5}\ \ 9\ \ \overline{8}\ \ \overline{7}\ \ \overline{9}\ \ \circ)$$

A consecutive sequence of double-adjacencies can be rewritten as a single marker; this process is called *reduction*. For example, genome G can be reduced by rewriting $2 \cdot 4 \cdot 5$ and $\overline{2} \cdot \overline{4} \cdot \overline{5}$ as 10 and $\overline{10}$, yielding the following genome:

$$G^r = (\circ\ \ 1\ \ \overline{1}\ \ 3\ \ 10\ \ 6\ \ \overline{6}\ \ 7\ \ \overline{3}\ \ 8\ \ \overline{10}\ \ 9\ \ \overline{8}\ \ \overline{7}\ \ \overline{9}\ \ \circ)$$

Definition 2. *A* single tandem duplicated genome *is a totally duplicated genome composed of a single linear chromosome which can be reduced to a chromosome of the form* $(\circ\ \ x\ \ \overline{x}\ \ \circ)$.

In other words, a single tandem duplicated genome is composed of a single linear chromosome where all adjacencies, except the two containing the marker $\circ$ and the central adjacency, are double-adjacencies. For example, the genome $(\circ\ 1 \cdot 2 \cdot 3 \cdot 4\ \ \overline{1} \cdot \overline{2} \cdot \overline{3} \cdot \overline{4}\ \ \circ)$ is a single tandem duplicated genome that can be reduced to $(\circ\ \ 5\ \ \overline{5}\ \ \circ)$ by rewritting $1 \cdot 2 \cdot 3 \cdot 4$ and $\overline{1} \cdot \overline{2} \cdot \overline{3} \cdot \overline{4}$ as 5 and $\overline{5}$.

Definition 3. *A* perfectly duplicated genome *is a totally duplicated genome such that each adjacency is a double-adjacency.*

For example, the genome $(1\ \ 2\ \ \overline{1}\ \ \overline{2})\ (\circ\ \ 3\ \ 4\ \ \circ)\ (\circ\ \ \overline{3}\ \ \overline{4}\ \ \circ)$ is a perfectly duplicated genome composed of one single circular chromosome and two linear chromosomes.

In other words, a single tandem duplicated genome is the representation of a duplicated segment resulting from a tandem-duplication of a genomic sequence, and a perfectly duplicated genome represents the result of a whole-genome duplication event that has duplicated all chromosomes.

2.2 Rearrangements

A rearrangement operation on a given genome cuts a set of adjacencies of the genome called *breakpoints* and forms new adjacencies with the exposed extremities, while altering no other adjacency. In the sequel, the adjacencies cut by a rearrangement operation are indicated in the genome by the symbol $_{\blacktriangle}$.

An *interval* in a genome is a set of markers that appear consecutively in the genome. Given two different adjacencies $(a\ \ b)$ and $(c\ \ d)$ in a genome G such that $(a\ \ b) < (c\ \ d)$, $[b\ ;\ c]$ denotes the interval of G beginning with marker b and ending with marker c.

In this paper, we consider two types of rearrangement operations called *block interchange (BI)* and *double-cut-and-join (DCJ)*.

A *block interchange* (BI) on a genome G is a rearrangement operation that acts on four adjacencies in G, $(a\ \ b) < (c\ \ d) \leq (u\ \ v) < (x\ \ y)$ such that the intervals $[b\ ;\ c]$ and $[v\ ;\ x]$ do not overlap, swapping the intervals $[b\ ;\ c]$ and $[v\ ;\ x]$. For example, the following block interchange acting on adjacencies $(\overline{1}\ \ 2) < (6\ \ \overline{6}) < (\overline{3}\ \ 8) < (\overline{8}\ \ \overline{7})$ consists in swapping the intervals $[2, 6]$ and $[8, \overline{8}]$.

$$(\circ\ \ 1\ \ \overline{1}\ _{\blacktriangle}\ \mathbf{2}\ \ \mathbf{3}\ \ \mathbf{\overline{2}}\ \ \mathbf{4}\ \ \mathbf{5}\ \ \mathbf{6}\ _{\blacktriangle}\ \overline{6}\ \ 7\ \ \overline{3}\ _{\blacktriangle}\ \ \mathbf{8}\ \ \mathbf{\overline{4}}\ \ \mathbf{9}\ \ \mathbf{\overline{5}}\ \ \mathbf{\overline{8}}\ _{\blacktriangle}\ \overline{7}\ \ \overline{9}\ \ \circ)$$

$$\downarrow$$

$$(\circ\ \ 1\ \ \overline{1}\ \ \mathbf{8}\ \ \mathbf{\overline{4}}\ \ \mathbf{9}\ \ \mathbf{\overline{5}}\ \ \mathbf{\overline{8}}\ \ \overline{6}\ \ 7\ \ \overline{3}\ \ \mathbf{2}\ \ \mathbf{3}\ \ \mathbf{\overline{2}}\ \ \mathbf{4}\ \ \mathbf{5}\ \ \mathbf{6}\ \ \overline{7}\ \ \overline{9}\ \ \circ)$$

A *double-cut-and-join* (DCJ) operation on a genome G cuts two different adjacencies in G and glues pairs of the four exposed extremities to form two new adjacencies. Here, we focus on two types of DCJ operations called *excision* and *integration*.

An *excision* is a DCJ operation acting on a single chromosome by extracting an interval from it, making this interval a circular chromosome, and making the remainder a single chromosome.For example, the following excision extracts the circular chromosome $(2\ \ 3\ \ 4)$:

$$(\circ\ 1\ _{\blacktriangle}\ \mathbf{2}\ \ \mathbf{3}\ \ \mathbf{4}\ _{\blacktriangle}\ 5\ \ 6\ \circ) \rightarrow (\mathbf{2}\ \ \mathbf{3}\ \ \mathbf{4})(\circ 1\ \ 5\ \ 6\ \circ)$$

An *integration* is the inverse of an excision; it is a DCJ operation that acts on two chromosomes, one being a circular chromosome, to produce a single chromosome. For example, the following operation is an integration of the circular chromosome $(2\ \ 3\ \ 4)$:

$$(\mathbf{2}\ _{\blacktriangle}\ \mathbf{3}\ \ \mathbf{4})(\circ 1\ \ 5\ \ 6\ _{\blacktriangle}\ \circ) \rightarrow (\circ\ 1\ \ 5\ \ 6\ \ \mathbf{3}\ \ \mathbf{4}\ \ \mathbf{2}\ \circ)$$

We now give an obvious, but very useful, property linking BI operations to DCJ operations.

Property 1. A single BI operation on a linear chromosome is equivalent to two DCJ operations: an excision followed by an integration.

Proof. Let $(\circ\ \ 1\ \ U\ \ 2\ \ V\ \ 3\ \ \circ)$ be a genome, U and V the two intervals that are to be swapped by a block interchange operation, 1 2 and 3 the intervals constituting the rest of the genome (note that each of them may be empty).

The first DCJ operation is the excision that produces the adjacency $(1\ \ V)$ by extracting and circularizing the interval $[U\ ;\ 2]$:

$$(\circ\ \ 1\ _{\blacktriangle}\ U\ \ 2\ _{\blacktriangle}\ V\ \ 3\ \ \circ) \rightarrow (\circ\ \ 1\ \ V\ \ 3\ \ \circ)(U\ \ 2\ )$$

The second DCJ operation is the integration that produces the adjacency $(U\ \ 3)$ by reintegrating the circular chromosome $(U\ \ 2)$ in the appropriate way:

$$(\circ\ \ 1\ \ V\ _{\blacktriangle}\ 3\ \ \circ)(U\ \ 2\ _{\blacktriangle}) \rightarrow (\circ\ \ 1\ \ V\ \ 2\ \ U\ \ 3\ \ \circ).$$

A *rearrangement scenario* between two genomes A and B is a sequence of rearrangement operations allowing one to transform A into B.

Definition 4. *A* BI (resp. DCJ) scenario *is a rearrangement scenario composed of BI (resp. DCJ) operations.*

The length of a rearrangement scenario is the number of rearrangement operations composing the scenario.

Definition 5. *The* BI (resp. DCJ) distance *between two genomes A and B, denoted by $d_{BI}(A, B)$ (resp. $d_{DCJ}(A, B)$), is the minimal length of a BI (resp. DCJ) scenario between A and B.*

2.3 Genome Halving

We now state the genome halving problem considered in this paper.

Definition 6. *Given a totally duplicated genome G composed of a single linear chromosome, the* BI halving problem *consists in finding a single tandem duplicated genome H such that the BI distance between G and H is minimal.*

In order to solve the BI halving problem, we use some results on the *DCJ halving problem* that were stated in [6] as a starting point. However, unlike the BI halving problem, the aim of the DCJ halving problem is to find a perfectly duplicated genome instead of a single tandem duplicated genome.

Definition 7. *Given a totally duplicated genome G, the* DCJ genome halving problem *consists in finding a perfectly duplicated genome H such that the DCJ distance between G and H is minimal.*

The BI and DCJ genome halving problems lead to two definitions of *halving distances*: the *BI halving distance* (resp. *DCJ halving distance*) of a totally duplicated genome G is the minimum BI (resp. DCJ) distance between G and any single tandem duplicated genome (resp. any perfectly duplicated genome) ; we denote it by $d^t_{BI}(G)$ (resp. $d^p_{DCJ}(G)$).

3 LOWERBOUND FOR THE BI HALVING DISTANCE

In this section we give a lowerbound on the BI halving distance of a totally duplicated genome. We use a data structure representing the genome called the *natural graph* introduced in [6].

Definition 8. [6] *The natural graph of a totally duplicated genome G, denoted by $\mathtt{NG}(G)$, is the graph whose vertices are the* adjacencies *of G, and for any marker u there is one edge between $(u \quad v)$ and $(\overline{u} \quad w)$, and one edge between $(x \quad u)$ and $(y \quad \overline{u})$.*

Note that the number of edges in the natural graph of a genome G containing n distinct markers, each one present in two copies, is always $2n$. Moreover, since every vertex has degree one or two, then the natural graph consists only of cycles and paths. For example, the natural graph of genome $G = (\circ\ 1\ \overline{2}\ \overline{1}\ \overline{4}\ 3\ 4\ \overline{3}\ 2\ \circ)$ is depicted in Fig. 1.

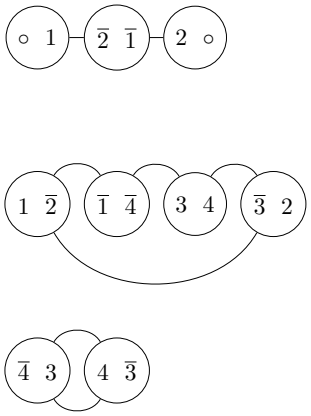


Fig. 1. The natural graph of genome $G = (\circ\ 1\ \ \overline{2}\ \ \overline{1}\ \ \underline{\overline{4}\ \ 3}\ \ \underline{4\ \ \overline{3}}\ \ 2\ \ \circ)$; it is composed of one path and two cycles.

Definition 9. *Given an integer k, a* $k-$cycle *(resp.* $k-$path*) in the natural graph of a totally duplicated genome is a cycle (resp. path) that contains k edges. If k is even, the cycle (resp. path) is called* even*, and* odd *otherwise.*

Based on the natural graph, a formula for the DCJ halving distance was given in [6]. Given a totally duplicated genome G such that the number of even cycles and the number of odd paths in $\mathtt{NG}(G)$ are respectively denoted by EC and OP, the DCJ halving distance of G is:

$$d^p_{DCJ}(G) = n - \mathrm{EC} - \left\lfloor \frac{\mathrm{OP}}{2} \right\rfloor$$

In the case of the BI halving distance, some peculiar properties of the natural graph need to be stated, allowing one to simplify the formula of the DCJ halving distance, and leading to a lowerbound on the BI halving distance.

In the following properties, we assume that G is a genome composed of a single linear chromosome containing n distinct markers, each one present in two copies in G.

Property 2. The natural graph $\mathtt{NG}(G)$ contains only even cycles and paths:

1. All cycles in the natural graph $\mathtt{NG}(G)$ are even.
2. The natural graph $\mathtt{NG}(G)$ contains only one path, and this path is even.

Proof. First, if $(a \quad x)$ is a vertex of the graph that belongs to a cycle C, then there exists an edge between $(a \quad x)$ and a vertex $(\overline{a} \quad y)$. These two adjacencies are the only two containing a copy of the marker a at the first position. So, if we consider the set of all the first markers in all adjacencies contained in the cycle C, then each marker in this set is present exactly twice. Therefore, the cycle C is an even cycle.

Secondly, the graph contains exactly two vertices (adjacencies) containing the marker $\circ$ which are both necessarily ends of a path in $\mathtt{NG}(G)$. Thus there can be only one path in the graph. Since the number of edges in the graph is even and all cycles are even, then the single path is also even.

We now give a lowerbound on the minimum length of DCJ scenario transforming G into a single tandem duplicated genome.

Lemma 1. *Let $d^t_{DCJ}(G)$ be the minimum DCJ distance between G and any single tandem duplicated genome. If $\mathtt{NG}(G)$ contains C cycles then a lowerbound on $d^t_{DCJ}(G)$ is given by:*

$$d^t_{DCJ}(G) \geq n - C - 1$$

Proof. First, since all cycles of $\mathtt{NG}(G)$ are even and $\mathtt{NG}(G)$ contains no odd path, then, from the DCJ halving distance formula, the DCJ halving distance of G is $d^p_{DCJ}(G) = n - C$.

Now, since any single tandem duplicated genome can be transformed into a perfectly duplicated genome with one DCJ, then $d^t_{DCJ} + 1 \geq d^p_{DCJ}$. Therefore, we have $d^t_{DCJ} \geq d^p_{DCJ} - 1 \geq n - C - 1$.

We are now ready to state a lowerbound on the BI halving distance of a totally duplicated genome G.

Theorem 1. *If $\mathtt{NG}(G)$ contains C cycles, then a lowerbound on the BI halving distance is given by:*

$$d_{BI}^{t}(G) \geq \left\lfloor \frac{n-C}{2} \right\rfloor$$

Proof. We denote by $\ell(S)$ the length of a rearrangement scenario S. Let S_{BI} be a BI scenario transforming G into a single tandem duplicated genome. From property 1, we have that S_{BI} is equivalent to a DCJ scenario S_{DCJ} such that $\ell(S_{DCJ}) = 2 * \ell(S_{BI})$. Now, suppose that $\ell(S_{BI}) < \lfloor \frac{n-C}{2} \rfloor$, then $\ell(S_{BI}) \leq \lfloor \frac{n-C}{2} \rfloor - 1 \leq \lceil \frac{n-C-1}{2} \rceil - 1$.

This implies $\ell(S_{DCJ}) \leq 2\lceil \frac{n-C-1}{2} \rceil - 2 \leq n - C - 2 < n - C - 1$. Thus, from Lemma 1 we have $\ell(S_{DCJ}) < d_{DCJ}^{t}$ which contradicts the fact that d_{DCJ}^{t} is the minimal number of DCJ operations required to transform G into a single tandem duplicated genome.

In conclusion, we always have $d_{BI}^{t}(G) \geq \lfloor \frac{n-C}{2} \rfloor$.

4 FORMULA FOR THE BI HALVING DISTANCE

In this section, we show that the BI halving distance of a totally duplicated genome G with n distinct markers such that $\mathtt{NG}(G)$ contains C cycles is exactly:

$$d_{BI}^{t}(G) = \left\lfloor \frac{n-C}{2} \right\rfloor$$

In other words, we show that enforcing the constraint that successive couples of consecutive DCJ operations have to be equivalent to BI operations does not change the distance even though it obviously restricts the DCJ that can be performed at each step of the scenario.

In the following, G denotes a totally duplicated genome G constisting in a single linear chromosome with n distinct markers after the reduction process, and such that $\mathtt{NG}(G)$ contains C cycles. We begin by recalling some useful definitions and properties of the DCJ operations that allow one to decrease the DCJ halving distance by 1 in the resulting genome.

Definition 10. *A DCJ operation on G producing genome G' is* sorting *if it decreases the DCJ halving distance by* 1*: $d_{DCJ}^{p}(G') = d_{DCJ}^{p}(G) - 1 = n - C - 1$.*

Since the number of distinct markers in G' is n and $d_{DCJ}^{p}(G') = n - C - 1$, then $\mathtt{NG}(G')$ contains $C + 1$ cycles. In other words, a DCJ operation is sorting if it increases the number of cycles in $\mathtt{NG}(G)$ by 1.

Given $(u \;\; v)$ an adjacency of G that is not a double-adjacency, we denote by $DCJ(u \;\; v)$ the DCJ operation that cuts adjacencies $(\overline{u} \;\; x)$ and $(y \;\; \overline{v})$ to form adjacencies $(\overline{u} \;\; \overline{v})$ and $(y \;\; x)$, making $(u \;\; v)$ a double-adjacency.

Property 3. Let $(u \;\; v)$ be an adjacency of G that is not a double-adjacency, $DCJ(u \;\; v)$ is a sorting DCJ operation.

Proof. $DCJ(u \;\; v)$ increases the number of cycles in $\mathtt{NG}(G)$ by 1, by creating a new cycle composed of adjacencies $(u \;\; v)$ and $(\overline{u} \;\; \overline{v})$.

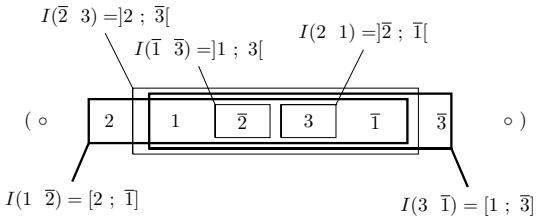


Fig. 2. $\mathcal{I}(G) = \{]\overline{2} \; ; \; \overline{1}[\, , [\mathbf{2} \; ; \; \overline{\mathbf{1}}] \, , \,]2 \; ; \; \overline{3}[\, , \, [\mathbf{1} \; ; \; \overline{\mathbf{3}}] \, , \,]1 \; ; \; 3[\}$, the set of intervals of $G = (\circ \;\; 2 \;\; 1 \;\; \overline{2} \;\; 3 \;\; \overline{1} \;\; \overline{3} \;\; \circ)$ depicted as boxes. The two boxes with thick lines represent two overlapping intervals of $\mathcal{I}(G)$ inducing a BI which exchanges 2 and $\overline{3}$.

Definition 11. *Let $(u \;\; v)$, $(\overline{u} \;\; x)$, and $(y \;\; \overline{v})$ be adjacencies of G. The* interval *of the adjacency $(u \;\; v)$, denoted by $I(u \;\; v)$ is either:*

- *the interval $[x \; ; \; y]$ if $(\overline{u} \;\; x) < (y \;\; \overline{v})$. In this case, we denote it by $]\overline{u} \; ; \; \overline{v}[$, or*
- *the interval $[\overline{v} \; ; \; \overline{u}]$ if $(y \;\; \overline{v}) < (\overline{u} \;\; x)$.*

For example, the intervals of the adjacencies in genome $(\circ \;\; 2 \;\; 1 \;\; \overline{2} \;\; 3 \;\; \overline{1} \;\; \overline{3} \;\; \circ)$ are depicted in Fig 2. Note that, given an adjacency $(u \;\; v)$ of G, if $(u \;\; v)$ is a double-adjacency then the interval $I(u \;\; v)$ is empty, otherwise $DCJ(u \;\; v)$ is the excision operation that extracts the interval $I(u \;\; v)$ to make it circular, thus producing the adjacency $(\overline{u} \;\; \overline{v})$.

Two intervals $I(a \;\; b)$ and $I(x \;\; y)$ are said to be *overlapping* if their intersection is non-empty, and none of the intervals is included in the other. It is easy to see, following Property 1, that given two adjacencies $(a \;\; b)$ and $(x \;\; y)$ of G such that $I(a \;\; b)$ and $I(x \;\; y)$ are non-empty intervals, the successive application of $DCJ(a \;\; b)$ and $DCJ(x \;\; y)$ is equivalent to a BI operation if and only if $I(a \;\; b)$ and $I(x \;\; y)$ are overlapping. Note that in this case neither $(a \;\; b)$, nor $(x \;\; y)$ can be double-adjacencies in G since their intervals are non-empty. Figure 2 shows an example of two overlapping intervals.

The following property states precisely in which case the successive application of $DCJ(a \;\; b)$ and $DCJ(x \;\; y)$ decreases the DCJ halving distance by 2, meaning that both DCJ operations are sorting.

Property 4. Given two adjacencies $(a \;\; b)$ and $(x \;\; y)$ of G, such that $I(a \;\; b)$ and $I(x \;\; y)$ are overlapping, the successive application of $DCJ(a \;\; b)$ and $DCJ(x \;\; y)$ decreases the DCJ halving distance by 2 if and only if $x \neq \overline{a}$ and $y \neq \overline{b}$.

Proof. If $x \neq \overline{a}$ and $y \neq \overline{b}$, then the successive application of $DCJ(a\ \ b)$ and $DCJ(x\ \ y)$ increases the number of cycles in $\mathtt{NG}(G)$ by 2, by creating two new 2-cycles. Otherwise, $DCJ(a\ \ b)$ first creates a new cycle that is then destroyed by $DCJ(x\ \ y)$.

We denote by $\mathcal{I}(G)$, the set of intervals of all the adjacencies of G that do not contain marker $\circ$.

Remark 1. Note that, if G contains n distinct markers, then there are $2n-1$ adjacencies in G that do not contain marker $\circ$, defining $2n-1$ intervals in $\mathcal{I}(G)$.

Definition 12. *Two intervals $I(a\ \ b)$ and $I(x\ \ y)$ of $\mathcal{I}(G)$ are said to be* compatible *if they are overlapping and $x \neq \overline{a}$ and $y \neq \overline{b}$.*

In the following, we prove the BI halving distance formula by showing that if genome G contains more than three distinct markers, $n > 3$, then there exist two compatible intervals in $\mathcal{I}(G)$, and if $n = 2$ or $n = 3$ then $d^t_{BI}(G) = 1$ and $2 \leq d^p_{DCJ}(G) \leq 3$. This means that there exists a BI halving scenario S such that all BI operations in S, possibly excluding the last one, are equivalent to two successive sorting DCJ operations.

From now on, until the end of the section, $(a\ \ b)$ is an adjacency of G that is not a double-adjacency, A is a genome consisting in a linear chromosome $\mathfrak{L}$ and a circular chromosome $\mathfrak{C}$, obtained by applying the *sorting DCJ*, $DCJ(a\ \ b)$, on G.

If there exists an interval $I(x\ \ y)$ in $\mathcal{I}(G)$ compatible with $I(a\ \ b)$, then applying $DCJ(x\ \ y)$ on A consists in the integration of the circular chromosome $\mathfrak{C}$ into the linear chromosome $\mathfrak{L}$ such that the adjacency $(\overline{x}\ \ \overline{y})$ is formed. Such an *integration* can only be performed by cutting an adjacency $(\overline{x}\ \ u)$ in $\mathfrak{C}$ and an adjacency $(v\ \ \overline{y})$ in $\mathfrak{L}$ (or inversely) to produce adjacencies $(\overline{x}\ \ \overline{y})$ and $(v\ \ u)$. This means that there must be an adjacency $(x\ \ y)$ in either $\mathfrak{C}$ or $\mathfrak{L}$ such that $\overline{x}$ is in $\mathfrak{C}$ and $\overline{y}$ in $\mathfrak{L}$ or inversely. Hence, we have the following property :

Property 5. $\mathfrak{C}$ *cannot* be reintegrated into $\mathfrak{L}$ by applying a sorting DCJ, $DCJ(x\ \ y)$, on A if and only if either:

(1) for any adjacency $(x\ \ y)$ in $\mathfrak{C}$ (resp. $\mathfrak{L}$), markers $\overline{x}$ and $\overline{y}$ are in $\mathfrak{L}$ (resp. $\mathfrak{C}$), or
(2) for any adjacency $(x\ \ y)$ in $\mathfrak{C}$ (resp. $\mathfrak{L}$), markers $\overline{x}$ and $\overline{y}$ are also in $\mathfrak{C}$ (resp. $\mathfrak{L}$).

Proof. If there exists no adjacency $(x\ \ y)$ in A such that $\overline{x}$ is in $\mathfrak{C}$ and $\overline{y}$ in $\mathfrak{L}$ or inversely, then A necessarily satisfies either (1), or (2).

Definition 13. *An interval $I(a\ \ b)$ in $\mathcal{I}(G)$ is called* interval of type 1 *(resp.* interval of type 2*) if $DCJ(a\ \ b)$ produces a genome A satisfying configuration* (1) *(resp. configuration* (2)*) described in Property 5.*

For example, in genome $(\circ\ 2\ 1\ \overline{1}\ 3\ \overline{2}\ \overline{3}\ \circ)$, $I(\overline{1}\ 3)$ is of type 1 as $DCJ(\overline{1}\ 3)$ produces genome $(\circ\ 2\ 1\ \overline{3}\ \circ)$ $(\overline{1}\ 3\ \overline{2})$; $I(\overline{2}\ \overline{3})$ is of type 2 as $DCJ(\overline{2}\ \overline{3})$ produces genome $(\circ\ 2\ 3\ \overline{2}\ \overline{3}\ \circ)$ $(1\ \overline{1})$.

Now we give the maximum numbers of intervals of type 1 and type 2 that can be contained in genome G.

Lemma 2. *The maximum number of intervals of type 1 in $\mathcal{I}(G)$ is 2.*

Proof. First, note that there cannot be two intervals I and J of $\mathcal{I}(G)$ such that $I \neq J$, and both I and J are of type 1. Now, if I is an interval of type 1, there can be at most two different adjacencies $(x\ \ y)$ and $(u\ \ v)$ such that $I(x\ \ y) = I(u\ \ v) = I$. In this case G necessarily has a chromosome of the form $(\ldots\ \overline{x}\ \overline{v}\ \ldots\ \overline{u}\ \overline{y}\ \ldots)$ or $(\ldots\ \overline{u}\ \overline{y}\ \ldots\ \overline{x}\ \overline{v}\ \ldots)$. Therefore, there are at most two intervals of type 1 in $\mathcal{I}(G)$.

Lemma 3. *The maximum number of intervals of type 2 in $\mathcal{I}(G)$ is n.*

Proof. First, note that for two adjacencies $(x\ \ y)$ and $(\overline{x}\ \ z)$ in G that do not contain marker $\circ$, if $(x\ \ y)$ is of type 2 then $(\overline{x}\ \ z)$ cannot be of type 2. Now, there is only one marker u such that $(\overline{u}\ \ \circ)$ is an adjacency of G. Let $(u\ \ v)$ be the adjacency of G having u as first marker, then at most half of the intervals in $\mathcal{I}(G) - \{I(u\ \ v)\}$ can be of type 2. Therefore, there are at most n intervals of type 2 in $\mathcal{I}(G)$.

Theorem 2. *If $\mathtt{NG}(G)$ contains C cycles, then the BI halving distance of G is given by:*

$$d^t_{BI}(G) = \left\lfloor \frac{n - C}{2} \right\rfloor$$

Proof. Since there are $2n-1$ intervals in $\mathcal{I}(G)$, and at most $n+2$ are of type 1 or 2, then if G is a genome containing more than three distinct markers $n > 3$, then $2n-1 > n+2$ and there exist two compatible intervals in $\mathcal{I}(G)$ inducing a BI operation that decreases the DCJ distance by 2.

Next, we show that if $n = 2$ or $n = 3$, then $d^t_{BI}(G) = 1$ and $2 \leq d^p_{DCJ}(G) \leq 3$.

If $n = 2$, then the genome can be written, either as $(\circ\ a\ b\ \overline{b}\ \overline{a}\ \circ)$, in which case a BI can swap a and b to produce a single tandem duplicated genome, or as $(\circ\ a\ \overline{a}\ b\ \overline{b}\ \circ)$, in which case a BI can swap a and $\overline{a}\ b$ to produce a single tandem duplicated genome.

If $n = 3$, then the genome has two double-adjacencies to be constructed, of the form $(\overline{a}\ \ \overline{b})$, $(\overline{x}\ \ \overline{y})$, with $(a\ \ b)$ and $(x\ \ y)$ being two adjacencies already present in the genome such that $b = x$ or $b = \overline{x}$ and a and y are distinct markers. One can rewrite $(a\ \ b)$ and $(x\ \ y)$ as single markers since they will not be splitted, which makes a genome with 4 markers such that at most 2 are misplaced. Then, a single BI can produce a single tandem duplicated genome.

Now, it is easy to see to see that if $n = 2$ or $n = 3$, then $d^p_{DCJ}(G) = n - C \leq 3$. Finally, if $n = 2$ or $n = 3$, then $d^p_{DCJ}(G) \geq 2$, otherwise we would have

$d^p_{DCJ}(G) = 1$ which would imply, as G consists in a single linear chromosome, $d^t_{BI}(G) = 0$. In conclusion, if $n > 3$ then there exist two compatible intervals in $\mathcal{I}(G)$, otherwise if $n = 2$ or $n = 3$, then $d^t_{BI}(G) = 1$ and $2 \leq d^p_{DCJ}(G) \leq 3$. Therefore $d^t_{BI} = \lfloor \frac{d^p_{DCJ}}{2} \rfloor = \lfloor \frac{n-C}{2} \rfloor$.

5 SORTING ALGORITHM

In Section 4, we showed that if a genome G contains more than three distinct markers after reduction then there exist two compatible intervals in $\mathcal{I}(G)$ inducing a BI to perform. If G contains two or three distinct markers then the BI to perform can be trivially computed. Thus the main concern of this section is to describe an efficient algorithm for finding compatible intervals when $n > 3$.

As in Section 4, in the following, G denotes a genome consisting of n distinct markers after reduction. It is easy to show that the set of intervals $\mathcal{I}(G)$ can be built in $O(n)$ time and space complexity.

We now show that finding 2 compatible intervals in $\mathcal{I}(G)$ can be done in $O(n)$ time and space complexity.

Property 6. If $n > 3$, then all the smallest intervals in $\mathcal{I}(G)$ that are not of type 2 admit compatible intervals.

Proof. Let J be a smallest interval that is not of type 2 in $\mathcal{I}(G)$. As J is not of type 2, then J has compatible intervals if J is not of type 1.

Let us suppose that J is of type 1, then for any adjacency $(a\ b)$ such that markers a and b are not in J, $\overline{a}$ and $\overline{b}$ are in J, and then $I(a\ b)$ is strictly included in J and $I(a\ b)$ can't be of type 2. Such adjacency does exist as there are $n > 3$ markers not included in J. Therefore J cannot be a smallest interval that is not of type 2.

We are now ready to give the algorithm for sorting a duplicated genome G into a single tandem duplicated genome with $\lfloor \frac{n-C}{2} \rfloor$ BI operations.

Algorithm 1 Reconstruction of a single tandem duplicated genome

```
while G contains more than 3 markers do
    Construct I(G)
    Pick a smallest interval I(a  b) that is not of type 2 in I(G)
    Find an interval I(x  y) in I(G) compatible with I(a  b)
    Perform the BI  equivalent to DCJ(a  b) followed by DCJ(x  y)
    Reduce G
end while
if G contains 2 or 3 markers then
    Find the last BI operation and perform it
end if
```

Theorem 3. *Algorithm 1 reconstruct a single tandem duplicated genome with a BI scenario of length* $\lfloor \frac{n-C}{2} \rfloor$ *in* $O(n^2)$ *time and space complexity.*

Proof. Building $\mathcal{I}(G)$ and finding two compatible intervals can be done in $O(n)$ time and space complexity. It follows that the while loop in the algorithm can be computed in $O(n^2)$ time and space complexity.

Finding and performing the last BI operation when $2 \leq n \leq 3$ can be done in constant time and space complexity.

Moreover, all BI operations, possibly excluding the last one, are computed as pairs of sorting DCJ operations, which ensures that the length of the scenario is $\lfloor \frac{n-C}{2} \rfloor$.

An example of scenario is shown in figure 3.

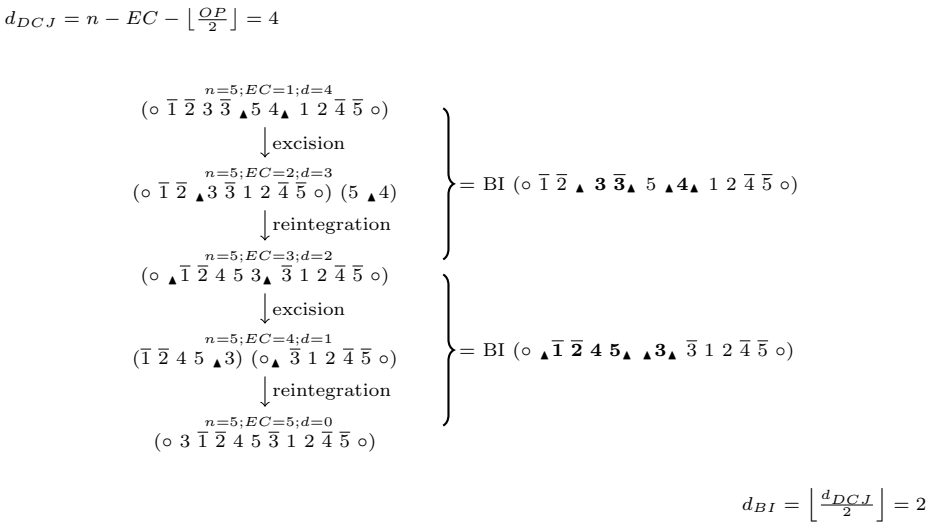


Fig. 3. A BI scenario computed by algorithm 1.

6 CONCLUSION

In this paper, we introduced the single tandem halving problem. We used the DCJ model to simulate BI operations and we showed that it is always possible to choose two consecutive sorting DCJ operations such that they are equivalent to a BI operation. This is an interesting result as it shows that restricting the scope of allowed DCJ operations under the constraint of performing only BI doesn't affect our halving distance. This also means our BI scenarios are in fact optimal DCJ scenarios. We thus provided a quadratic time and space algorithm to obtain a most parsimonious scenario for the BI halving problem. One direction for further

studies of variants of the BI halving problem is to consider multichromosomal genomes. A further extension of these results will be a generalization to the more general DCJ model, without the orientation constraints on the genome that the BI model implied, as well as ways of reconstructing more than a single tandem.

References

1. David A. Christie. Sorting permutations by block-interchanges. *Inf. Process. Lett.*, 60(4):165–169, 1996.
2. Nadia El-Mabrouk, Joseph H. Nadeau, and David Sankoff. Genome halving. In Martin Farach-Colton, editor, *Proceedings of CPM'98*, volume 1448 of *Lecture Notes in Computer Science*, pages 235–250. Springer, 1998.
3. Nadia El-Mabrouk and David Sankoff. The reconstruction of doubled genomes. *SIAM J. Comput.*, 32(3):754–792, 2003.
4. Jakub Kovác, Marília D. V. Braga, and Jens Stoye. The problem of chromosome reincorporation in DCJ sorting and halving. In Eric Tannier, editor, *RECOMB-CG*, volume 6398 of *Lecture Notes in Computer Science*, pages 13–24. Springer, 2010.
5. Ying Chih Lin, Chin Lung Lu, Hwan-You Chang, and Chuan Yi Tang. An efficient algorithm for sorting by block-interchanges and its application to the evolution of vibrio species. *Journal of Computational Biology*, 12(1):102–112, 2005.
6. Julia Mixtacki. Genome halving under DCJ revisited. In Xiaodong Hu and Jie Wang, editors, *Proceedings of COCOON'08*, volume 5092 of *Lecture Notes in Computer Science*, pages 276–286. Springer, 2008.
7. Eric Tannier, Chunfang Zheng, and David Sankoff. Multichromosomal genome median and halving problems. In Keith A. Crandall and Jens Lagergren, editors, *Proceedings of WABI'08*, volume 5251 of *Lecture Notes in Computer Science*, pages 1–13. Springer, 2008.
8. Antoine Thomas, Aïda Ouangraoua, and Jean-Stéphane Varré. Genome halving by block interchange. In BIOINFORMATICS'2012, Third International Conference on Bioinformatics Models, Methods and Algorithms, Vilamoura, Portugal, 1-4 February, 2012.
9. Robert Warren and David Sankoff. Genome halving with double cut and join. In Alvis Brazma, Satoru Miyano, and Tatsuya Akutsu, editors, *Proceedings of APBC'08*, volume 6 of *Advances in Bioinformatics and Computational Biology*, pages 231–240. Imperial College Press, 2008.
10. Sophia Yancopoulos, Oliver Attie, and Richard Friedberg. Efficient sorting of genomic permutations by translocation, inversion and block interchange. *Bioinformatics*, 21(16):3340–3346, 2005.