Super Short Reversals on Both Gene Order and Intergenic Sizes

Andre Rodrigues Oliveira[1(✉)], Géraldine Jean[2], Guillaume Fertin[2], Ulisses Dias[3], and Zanoni Dias[1]

[1] Institute of Computing, University of Campinas, Campinas, Brazil
{andrero,zanoni}@ic.unicamp.br
[2] Laboratoire des Sciences du Numérique de Nantes, UMR CNRS 6004, University of Nantes, Nantes, France
{geraldine.jean,guillaume.fertin}@univ-nantes.fr
[3] School of Technology, University of Campinas, Limeira, Brazil
ulisses@ft.unicamp.br

Abstract. The evolutionary distance between two genomes can be estimated by computing the minimum length sequence of operations, called *genome rearrangements*, that transform one genome into another. Usually, a genome is modeled as an ordered sequence of (possibly signed) genes, and almost all the studies that have been undertaken in the genome rearrangement literature consist in shaping biological scenarios into mathematical models: for instance, allowing different genome rearrangements operations at the same time, adding constraints to these rearrangements (e.g., each rearrangement can affect at most a given number k of genes), considering that a rearrangement implies a cost depending on its length rather than a unit cost, etc. However, most of the works in the field have overlooked some important features inside genomes, such as the presence of sequences of nucleotides between genes, called *intergenic regions*. In this work, we investigate the problem of computing the distance between two genomes, taking into account both gene order and intergenic sizes; the genome rearrangement operation we consider here is a constrained type of reversals, called *super short reversals*, which affect up to two (consecutive) genes. We propose here three algorithms to solve the problem: a 3-approximation algorithm that applies to any instance, and two additional algorithms that apply only on specific types of genomes with respect to their gene order: the first one is an exact algorithm, while the second is a 2-approximation algorithm.

Keywords: Genome rearrangements · Intergenic regions
Super short reversals · Approximation algorithm

1 Introduction

Given two genomes $\mathcal{G}_1$ and $\mathcal{G}_2$, one way to estimate their evolutionary distance is to compute the minimum possible number of large scale events, called *genome*

R. Alves (Ed.): BSB 2018, LNBI 11228, pp. 14–25, 2018.
https://doi.org/10.1007/978-3-030-01722-4_2

rearrangements, that are needed to go from $\mathcal{G}_1$ to $\mathcal{G}_2$. The minimality requirement is dictated by the commonly accepted parsimony principle, while the allowed genome rearrangements depend on the model, i.e. on the classes of events that supposedly happen during evolution.

However, before one performs this task, it is necessary to model the input genomes. Almost all previous works have defined genomes as ordered sequences of elements, which are *genes*. Variants within this setting can occur: for instance, depending on the model, genes may be signed or unsigned, the sign of a gene representing the DNA strand it lies on. Besides, each gene may appear either once or several times in a genome: in the latter case, genomes are modeled as strings, while in the former case they are modeled as *permutations*.

Concerning genome rearrangements, the most commonly studied is *reversal*, which consists in taking a continuous sequence in the genome, reversing it, and putting it back at the same location (see e.g. [10] for one of the first studies of the problem). A more recent and general type of genome rearrangement is the DCJ (for Double-Cut and Join) [14]. One can also alternately define the rearrangement events in order to reflect specific biological scenarios. For example, in populations where the number of rearrangement events that affect a very large portion of the genes is known to be rare, we can restrict events to be applied over no more than k genes at the same time, for some predetermined value of k [5,8,9].

Since the mid-nineties, a very large amount of work has been done concerning algorithmic issues of computing distances between pairs of genomes, depending on the genome model and the allowed set of rearrangements. For instance, if one considers reversals in unsigned permutations, the problem is known to be NP-hard [4], while it is polynomial-time solvable in signed permutations [10]. We refer the reader to Fertin et al.'s book [7] for a survey of the algorithmics aspects of the subject.

As previously mentioned, almost all of these works have so far assumed that a genome is an ordered sequence of genes. However, it has recently been argued that this model could underestimate the "true" evolutionary distance, and that other genome features may require to be taken into account in the model in order to circumvent this problem [1,2].

Indeed, genomes carry more information than just their ordered sequences of genes, and in particular consecutive genes in a genome are separated by *intergenic regions*, which are DNA sequences between genes having different sizes (in terms of number of nucleotides).

This recently led some authors to model a genome as an ordered sequence of genes, together with an ordered list of its intergenic sizes, and to consider the problem of computing the DCJ distance, either in the case where insertions and deletions of nucleotides are forbidden [6], or allowed [3].

In this work, we also consider genomes as ordered sequences of genes together with their intergenic sizes, in the case where the gene sequence is an unsigned permutation and where the considered rearrangement operation is *super short reversal* (or SSR, i.e. a reversal of (gene) length at most two). In this context, our

goal is to determine the minimum number of SSRs that transform one genome into another.

Sorting by super short reversals and/or super short transpositions (i.e. transpositions of (gene) length at most two each) has been studied in linear and circular genomes, signed and unsigned, and in all cases the problem has been shown to be in P class [8,9,11–13].

This paper is organized as follows. In Sect. 2 we provide the notations that we will use throughout the paper, and we introduce new notions that will prove useful for studying the problem. In Sect. 3, we derive lower and upper bounds on the sought distance, which in turn will help us design three different algorithms: one applies to the general case, while the remaining two apply to specific classes of genomes. Section 4 concludes the paper.

2 Definitions

We can represent a genome $\mathcal{G}$ with n genes as an n-tuple. When there is no duplicated genes, the n-tuple is a permutation $\pi = (\pi_1\ \pi_2\ ...\ \pi_{n-1}\ \pi_n)$ with $\pi_i \in \{1, 2, ..., (n-1), n\}$, for $1 \leq i \leq n$, and $\pi_i = \pi_j$ if, and only if, $i = j$. We denote by ι the *identity permutation*, the permutation in which all elements are in ascending order. The *extended permutation* is obtained from π by adding two new elements: $\pi_0 = 0$ and $\pi_{n+1} = (n+1)$.

A genome $\mathcal{G}$, represented by a permutation π with n elements, has $m = n+1$ intergenic regions $r^\pi = (r_1^\pi, ..., r_m^\pi)$, with $r_j^\pi \geq 0$ for $1 \leq j \leq m$, such that the intergenic region r_i^π is located before element π_i, for $1 \leq i \leq n$, and the intergenic region r_m^π is situated right after element π_n.

A *reversal* $\rho(i, j, x, y)$ applied over a permutation π, with $1 \leq i \leq j \leq n$, $0 \leq x \leq r_i^\pi$, and $0 \leq y \leq r_{j+1}^\pi$, is an operation that (i) reverses the order of the elements in the subset of adjacent elements $\{\pi_i, ..., \pi_j\}$; (ii) reverses the order of intergenic regions in the subset of adjacent intergenic regions $\{r_{i+1}^\pi, ..., r_j^\pi\}$ when $j > i+2$; (iii) *cuts* two intergenic regions: after position x inside intergenic region r_i^π and after position y inside intergenic region r_{j+1}^π. This reversal results in the permutation π' such that $r_i^{\pi'} = x + y$ and $r_{j+1}^{\pi'} = (r_i^\pi - x) + (r_{j+1}^\pi - y)$.

A reversal $\rho(i, j, x, y)$ is also called a k-reversal, where $k = (j-i)+1$. A *super short reversal* is a 1-reversal or a 2-reversal, i.e., a reversal that affects only one or two elements of π.

Figure 1 shows a sequence of three super short reversals that transforms the permutation $\pi = (1\ 3\ 4\ 2\ 5)$ with $r^\pi = (3, 5, 2, 1, 2, 8)$ into $\iota = (1\ 2\ 3\ 4\ 5)$ with $r^\iota = (3, 2, 6, 4, 5, 1)$.

A pair of elements (π_i, π_j) from π is called an *inversion* if $\pi_i > \pi_j$ and $i < j$, with $\{i, j\} \in [1..n]$. We denote the number of inversions in a permutation π by $inv(\pi)$. For the example above, $inv(\pi) = 2$.

Given two permutations π and α of same size, representing genomes $\mathcal{G}_1$ and $\mathcal{G}_2$ respectively, we denote by $W_i(\pi, \alpha) = r_i^\pi - r_i^\alpha$ the *imbalance* between intergenic regions r_i^π and r_i^α, with $1 \leq i \leq m$.

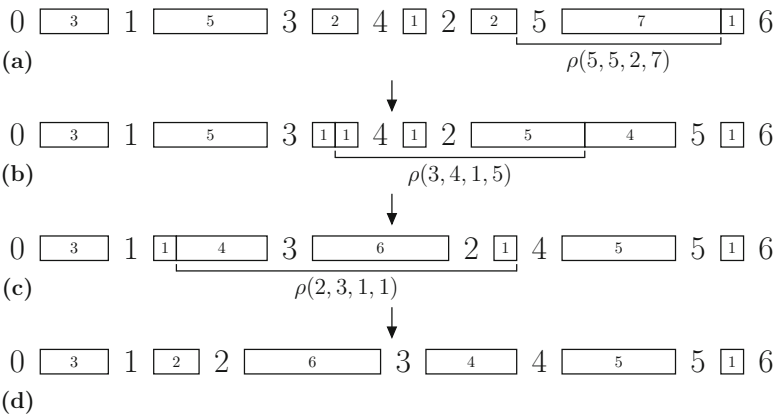

Fig. 1. A sequence of super short reversals that transforms $\pi = (1\ 3\ 4\ 2\ 5)$, with $r^{\pi} = (3, 5, 2, 1, 2, 8)$ into $\iota = (1\ 2\ 3\ 4\ 5)$, with $r^{\iota} = (3, 2, 6, 4, 5, 1)$. Intergenic regions are represented by rectangles, whose dimensions vary according to their sizes. The 1-reversal $\rho(5, 5, 2, 7)$ applied in **(a)** transforms π into $\pi' = \pi$, and it cuts π after position 2 at r_5^{π} and after position 7 at r_6^{π}, resulting in $r_5^{\pi'} = 9$, $r_6^{\pi'} = 1$, and $r^{\pi'} = (3, 5, 2, 1, 9, 1)$. The 2-reversal $\rho(3, 4, 1, 5)$ applied in **(b)** transforms π' into $\pi'' = (1\ 3\ 2\ 4\ 5)$, and it cuts π' after position 1 at $r_3^{\pi'}$ and after position 5 at $r_5^{\pi'}$, resulting in $r_3^{\pi''} = 6$, $r_5^{\pi''} = 5$, and $r^{\pi''} = (3, 5, 6, 1, 5, 1)$. Finally, the 2-reversal $\rho(2, 3, 1, 1)$ applied in **(c)** transforms π'' into ι, as shown in **(d)**.

Given two permutations π and α of same size and same total sum of the intergenic region lengths, let $S_j(\pi, \alpha) = \sum_{i=1}^{j} W_i(\pi, \alpha)$ be the cumulative sum of imbalances between intergenic regions of π and α from position 1 to j, with $1 \leq j \leq m$. Since π and α have same total sum of the intergenic region lengths, $S_m(\pi, \alpha) = 0$.

From now on, we will consider that (i) the target permutation α is such that $\alpha = \iota$; (ii) π and ι have the same number of elements; and (iii) the number of nucleotides inside intergenic regions of r^{π} equals the number of nucleotides inside intergenic regions of r^{ι}. By doing this, we can compute the *sorting distance* of π, denoted by $d(\pi)$, that consists in finding the minimum number of super short reversals that sorts π and transforms r^{π} into r^{ι}.

The *intergenic graph* of π with respect to the target permutation ι, denoted by $I(\pi, \iota) = (V, E)$, is such that V is composed by the set of intergenic regions r^{π} and the set of elements from the extended permutation π. Besides, the edge $e = (r_i^{\pi}, r_{i+2}^{\pi}) \in E$ if there is a $j \neq i$ such that (π_i, π_j) or (π_j, π_{i+1}) is an inversion, with $1 \leq i \leq n-1$ and $1 \leq j \leq n$.

A *component* c is a minimal set of consecutive elements from V in which: (i) the sum of imbalances of its intergenic regions with respect to r^{ι} is equal to zero; and (ii) any two intergenic regions that are connected to each other by an edge must belong to the same component.

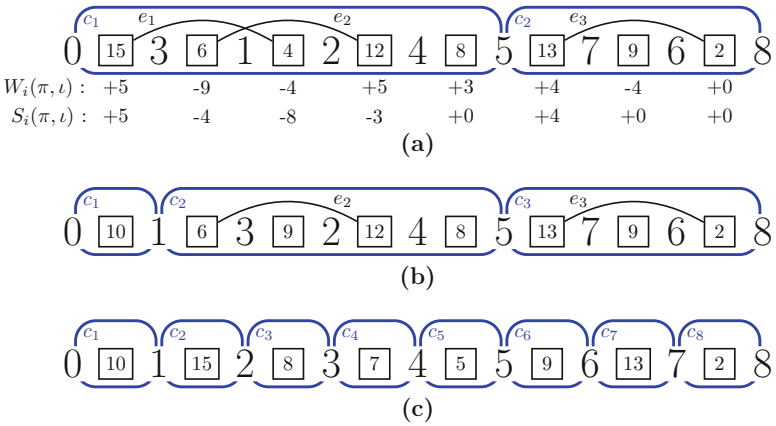

Fig. 2. Intergenic graphs $I(\pi,\iota)$ in **(a)**, $I(\pi',\iota)$ in **(b)**, and $I(\iota,\iota)$ in **(c)**, with $\pi = (3\ 1\ 2\ 4\ 5\ 7\ 6)$, $r^{\pi} = (15,6,4,12,8,13,9,2)$, $\pi' = (1\ 3\ 2\ 4\ 5\ 7\ 6)$, $r^{\pi'} = (10,6,9,12,8,13,9,2)$, $\iota = (1\ 2\ 3\ 4\ 5\ 6\ 7)$, and $r^{\iota} = (10,15,8,7,5,9,13,2)$. Black squares represent intergenic regions, and the number inside it indicate their sizes. Rounded rectangles in blue represent components. Note that in **(a)** there are three edges in $I(\pi,\iota)$, and $C(I(\pi,\iota)) = 2$. We also have in **(a)** all values for $S_i(\pi,\iota)$ and $W_i(\pi,\iota)$, with $1 \leq i \leq 8$. The permutation π' is the result of applying $\rho(1,2,8,2)$ to π. In **(b)** we can see that $I(\pi',\iota)$ has one more component than $I(\pi,\iota)$, and the edge e_1 was removed. In **(c)** we can see that when we reach the target permutation the number of components is equal to the number of intergenic regions in ι (i.e., $C(I(\iota,\iota)) = m = 8$).

A component always starts and finishes with elements from π. Besides, the first component starts with the element π_0, and the last component ends with the element π_{n+1}. Consecutive components share exactly one element from π, i.e., the last element π_i of a component is the first element of its adjacent component to the right. A component with one intergenic region is called *trivial*. The number of intergenic regions in a component c is denoted by $r(c)$. The number of components in a permutation π is denoted by $C(I(\pi,\iota))$. Figure 2 shows three examples of intergenic graphs.

3 Sorting Permutations by Super Short Reversals

In this section we analyze the version of the problem when only super short reversals (i.e., 1-reversals and 2-reversals) are allowed to sort a permutation on both order and intergenic regions. First, we show that any 1-reversal can increase the number of components by no more than one unit. After that, we state that if a component c of an intergenic graph $I(\pi,\iota)$ with $r(c) > 1$ has no edges (i.e., there is no inversions inside c), then it is always possible to split c into two components with a 1-reversal.

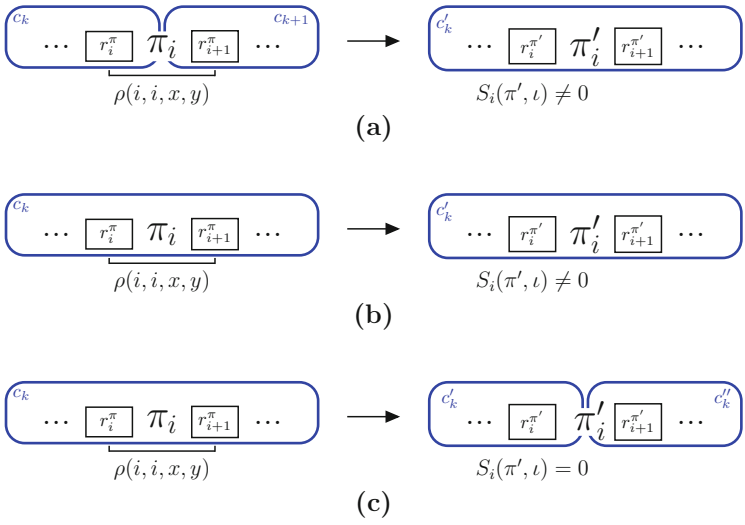

Fig. 3. Example of intergenic graphs for all possible values of $C(I(\pi', \iota))$ with respect to $C(I(\pi, \iota))$, where π' is the resulting permutation after applying a 1-reversal $\rho(i, i, x, y)$ to π. If the 1-reversal is applied over two components at the same time and $x+y \neq r_i^\pi$, then $C(I(\pi', \iota)) = C(I(\pi, \iota)) - 1$, as shown in **(a)**. If the 1-reversal is applied over one component, then either $C(I(\pi', \iota)) = C(I(\pi, \iota))$, if $x+y \neq r_i^\pi - S_i(\pi, \iota)$, or $C(I(\pi', \iota)) = C(I(\pi, \iota)) + 1$, if $x + y = r_i^\pi - S_i(\pi, \iota)$, as shown in **(b)** and **(c)** respectively.

Lemma 1. *Given a permutation π and a target permutation ι, let π' be the resulting permutation from π after applying a 1-reversal. It follows that $C(I(\pi, \iota)) - 1 \leq C(I(\pi', \iota)) \leq C(I(\pi, \iota)) + 1$.*

Proof. If a 1-reversal $\rho(i, i, x, y)$, applied over intergenic regions r_i^π and r_{r+1}^π, is applied over two different components in $I(\pi, \iota) = (V, E)$, then r_i^π is the last element of the first component, so $S_i(\pi, \iota) = 0$ and the graph $I(\pi', \iota) = (V', E')$, where π' is the resulting permutation, is such that $C(I(\pi', \iota)) = C(I(\pi, \iota)) - 1$ if $x + y \neq r_i^\pi$, as shown in Fig. 3(a). Let us consider now that this 1-reversal is applied over intergenic regions of a same component c.

First note that, since 1-reversals does not remove inversions from π, the intergenic graph $I(\pi', \iota)$ has $E' = E$. If $(r_i^{\pi'}, r_{i+2}^{\pi'}) \in E'$ (for $0 < i < n$), or $(r_{i-1}^{\pi'}, r_{i+1}^{\pi'}) \in E'$ (for $0 < i \leq n$), then $C(I(\pi', \iota)) = C(I(\pi, \iota))$. Otherwise, we have two cases to consider: $C(I(\pi', \iota)) = C(I(\pi, \iota))$, if $S_i(\pi', \iota) \neq 0$ (as shown in Fig. 3(b)); and $C(I(\pi', \iota)) = C(I(\pi, \iota)) + 1$ if $S_i(\pi', \iota) = 0$ (as shown in Fig. 3(c)). □

Lemma 2. *If a component c of an intergenic graph $I(\pi, \iota)$ with $r(c) \geq 2$ contains no edges, then there is always a pair of consecutive intergenic regions to which we can apply a 1-reversal that splits c into two components c' and c'' such that $r(c') + r(c'') = r(c)$.*

Proof. Let p_i be the index in r^π of the i-th intergenic region inside component c. The last intergenic region of c is at position $p_{r(c)}$. By definition of component, and since c contains no edges, for any $p_1 \leq j < p_{r(c)}$ we have that $S_j(\pi, \iota) \neq 0$. Note that since $r(c) > 1$ we have that $S_{p_1}(\pi, \iota) = W_{p_1}(\pi, \iota) \neq 0$.

If $S_{p_1}(\pi, \iota) > 0$, let k be the index of element from π located right after $r^\pi_{p_1}$. Apply the reversal $\rho(k, k, r^\iota_{p_1}, 0)$. Otherwise, we have that $S_{p_1}(\pi, \iota) < 0$, and we need to find two intergenic regions $r^\pi_{p_i}$ and $r^\pi_{p_{i+1}}$ for $1 \leq i < r(c)$ such that $S_{p_i}(\pi, \iota) < 0$ and $S_{p_{i+1}}(\pi, \iota) \geq 0$. Since, by definition of component, $S_{p_{r(c)}} = 0$, such a pair always exists. So, apply the reversal $\rho(p_i, p_i, r^\pi_{p_i}, -S_{p_i}(\pi, \iota))$.

In both cases, the resulting permutation π' has $S_{p_i}(\pi', \iota) = 0$, $S_{p_{i+1}}(\pi', \iota) = S_{p_{i+1}}(\pi, \iota) + S_{p_i}(\pi, \iota)$, and for any $i + 2 \leq j \leq r(c)$ we have that $S_{p_j}(\pi', \iota) = S_{p_j}(\pi, \iota)$ so, as before, all intergenic regions from $r^{\pi'}_{p_{i+1}}$ to $r^{\pi'}_{p_{r(c)}}$ must be in the same component.

This 1-reversal splits c into two components: c' with all intergenic regions in positions p_1 to p_i, and c'' with all intergenic regions in positions p_{i+1} to $p_{r(c)}$, and the lemma follows. □

Now we state that any 2-reversal can increase the number of components by no more than two units.

Lemma 3. *Given a permutation π and a target permutation ι, let π' be the resulting permutation from π after applying a 2-reversal. We have that $C(I(\pi, \iota)) - 2 \leq C(I(\pi', \iota)) \leq C(I(\pi, \iota)) + 2$.*

Proof. If a 2-reversal is applied over intergenic regions of two different components then we are necessarily creating a new inversion, and the graph $I(\pi', \iota) = (V', E')$, where π' is the resulting permutation, has $C(I(\pi', \iota)) = C(I(\pi, \iota)) - 2$ (as shown in Fig. 4(a)) or $C(I(\pi', \iota)) = C(I(\pi, \iota)) - 1$ (as shown in Fig. 4(b)). Let us consider now that the operation is applied over intergenic regions of a same component c.

Suppose that we apply an operation that exchanges elements π_i and π_{i+1}, with $1 \leq i < n - 1$. If the resulting permutation π' is such that $(r^{\pi'}_i, r^{\pi'}_{i+2}) \in E'$ then $C(I(\pi', \iota)) = C(I(\pi, \iota))$. Otherwise, we have three cases to consider: $C(I(\pi', \iota)) = C(I(\pi, \iota))$, if $S_i(\pi', \iota) \neq 0$ and $S_{i+1}(\pi', \iota) \neq 0$ (as shown in Fig. 4(c)); $C(I(\pi', \iota)) = C(I(\pi, \iota)) + 1$ if either $S_i(\pi', \iota) = 0$ or $S_{i+1}(\pi', \iota) = 0$ (as shown in Fig. 4(d)); and $C(I(\pi', \iota)) = C(I(\pi, \iota)) + 2$ otherwise (as shown in Fig. 4(e)). □

Using Lemmas 1, 2, and 3 we show in the following two lemmas the minimum and maximum number of super short reversals needed to transform π into ι and r^π into r^ι.

Lemma 4. *Given a genome $\mathcal{G}_1$, let π be its corresponding permutation with $r^\pi = (r^\pi_1, ..., r^\pi_m)$ intergenic regions. We have that $d(\pi) \geq \max(\frac{m - C(I(\pi,\iota))}{2}, inv(\pi))$, where ι is the corresponding permutation of the target genome $\mathcal{G}_2$.*

Proof. In order to sort π we need to remove all inversions, and since a 2-reversal can remove only one inversion, we necessarily have that $d(\pi) \geq inv(\pi)$. Besides,

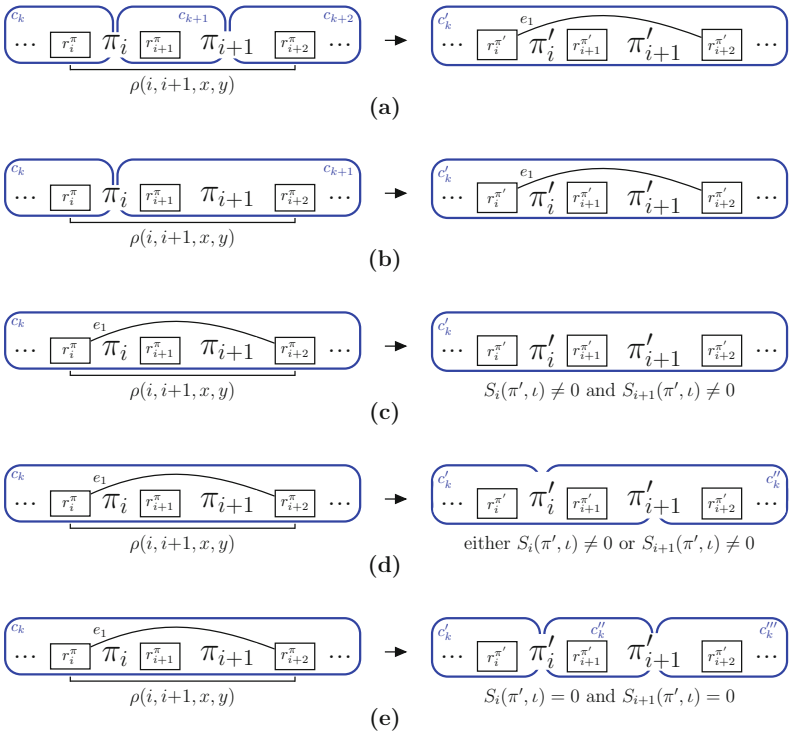

Fig. 4. Example of intergenic graphs for all possible values of $C(I(\pi', \iota))$ with respect to $C(I(\pi, \iota))$ where π' is the resulting permutation after applying a 2-reversal to π. When the 2-reversal is applied over two components at the same time then either $C(I(\pi', \iota)) = C(I(\pi, \iota)) - 2$, as shown in **(a)**, or $C(I(\pi', \iota)) = C(I(\pi, \iota)) - 1$, as shown in **(b)**. Otherwise, we have that either $C(I(\pi', \iota)) = C(I(\pi, \iota))$, if $S_i(\pi', \iota) \neq 0$ and $S_{i+1}(\pi', \iota) \neq 0$ as shown in **(c)**, or $C(I(\pi', \iota)) = C(I(\pi, \iota)) + 1$, if $e_1 \notin I(\pi', \iota)$ and either $S_i(\pi', \iota) \neq 0$ or $S_{i+1}(\pi', \iota) \neq 0$ as shown in **(d)**, or $C(I(\pi', \iota)) = C(I(\pi, \iota)) + 2$, if $e_1 \notin I(\pi', \iota)$, $S_i(\pi', \iota) = 0$ and $S_{i+1}(\pi', \iota) = 0$ as shown in **(e)**.

by Lemmas 1 and 3, we can increase the number of components by at most two with a super short reversal, so to reach m trivial components we need at least $\frac{m-C(I(\pi,\iota))}{2}$ super short reversals. Thus, $d(\pi) \geq \max(\frac{m-C(I(\pi,\iota))}{2}, inv(\pi))$. □

Lemma 5. *Given a genome $\mathcal{G}_1$, let π be its corresponding permutation with $r^\pi = (r_1^\pi, ..., r_m^\pi)$ intergenic regions. We have that $d(\pi) \leq inv(\pi)+m-C(I(\pi, \iota))$, where ι is the corresponding permutation of the target genome $\mathcal{G}_2$.*

Proof. Suppose that first we remove all inversions of π with $inv(\pi)$ 2-reversals of type $\rho(i, i+1, r_i^\pi, 0)$ i.e., without exchanging its intergenic regions. Let π' be the resulting permutation, with $r^{\pi'} = r^\pi$. The number of components in π cannot be smaller than $C(I(\pi, \iota))$ since each 2-reversal removing an inversion is applied inside a same component. Let us suppose then that π' has $k' \geq C(I(\pi, \iota))$

components. By Lemma 2, we can go from k' to m components using $m - k'$ 1-reversals, which results in no more than $m-C(I(\pi,\iota))$ 1-reversals, and the lemma follows. □

Finally, using Lemmas 4 and 5, we prove that it is possible to obtain a solution 3-approximable for this problem.

Theorem 6. *Given a genome $\mathcal{G}_1$ with its corresponding permutation π, and a target genome $\mathcal{G}_2$ with its corresponding permutation ι, the value of $d(\pi)$ is 3-approximable.*

Proof. Let us represent $\mathcal{G}_1$ by a permutation π with $r^\pi = (r_1^\pi, ..., r_m^\pi)$ intergenic regions, $inv(\pi)$ inversions, and let $k = C(I(\pi,\iota))$. If $\frac{m-k}{2} > inv(\pi)$ then, by Lemma 4, $d(\pi) \geq \frac{m-k}{2}$, and, by Lemma 5, $d(\pi) \leq m-k+inv(\pi) \leq m-k+\frac{m-k}{2} \leq 3\frac{m-k}{2}$. Otherwise, $\frac{m-k}{2} < inv(\pi)$, so $m - k < 2inv(\pi)$. By Lemma 4, $d(\pi) \geq inv(\pi)$, and, by Lemma 5, $d(\pi) \leq m-k+inv(\pi) \leq 2inv(\pi)+inv(\pi) \leq 3inv(\pi)$, and the lemma follows. □

Although Theorem 6 states that this problem is 3-approximable, it is possible to sort any permutation π and transform r^π into r^ι optimally if $\pi_1 = n$ and $\pi_n = 1$, as shown in the following lemma.

Lemma 7. *If a permutation π is such that $\pi_1 = n$ and $\pi_n = 1$, with $n > 1$, then $d(\pi) = inv(\pi) + \varphi(\pi)$, where $\varphi(\pi) = 1$, if the sum of imbalances of intergenic regions in odd positions of r^π differs from zero, and $\varphi(\pi) = 0$, otherwise.*

Proof. By Lemma 4, we have that $d(\pi) \geq inv(\pi)$. Besides, since only 2-reversals remove inversions, and since 2-reversals exchange nucleotides between intergenic regions of same parity only, then $d(\pi) \geq inv(\pi) + \varphi(\pi)$, with $\varphi(\pi) = 1$, if the cumulative sum of imbalances of intergenic regions in odd positions, denoted by $S_{odd}(\pi,\iota)$, differs from zero (in this case we will need at least one 1-reversal to exchange nucleotides between an odd and an even intergenic region), and $\varphi(\pi) = 0$ otherwise. Consider the following procedure, divided into four steps:

(i) Remove any inversion between elements in positions 2 to $(n-1)$ with 2-reversals of type $\rho(i, i+1, r_i^\pi, 0)$, and let $\pi' = (n\ 2\ ...\ (n-1)\ 1)$ be the resulting permutation. Note that $r^{\pi'} = r^\pi$, and π' has $(2n-3)$ inversions which means that $inv(\pi) - 2n + 3$ 2-reversals were applied.

(ii) Take the element $\pi'_1 = n$ to position $n-1$ by a sequence of $(n-2)$ 2-reversals of type $\rho(i, i+1, 0, 0)$, for $1 \leq i \leq n-2$, and let $\pi'' = (2\ 3\ ...\ n\ 1)$ be the resulting permutation. After this sequence is applied, all intergenic nucleotides are in the last three intergenic regions $r_{n-1}^{\pi'}$, $r_n^{\pi''}$ and $r_{n+1}^{\pi''}$ only, and $inv(\pi'') = n-1$.

(iii) Let $a = S_{odd}(\pi'',\iota)$, if n is odd, and $a = -S_{odd}(\pi,\iota)$ otherwise, and let $b = W_{n+1}(\pi'',\iota)$. If $b \geq 0$ (resp. $b < 0$) apply the 2-reversal $\rho(n-1, n, r_{n-1}^{\pi''}, b)$ balancing r_{n+1} (resp. if $a \neq 0$, apply the 1-reversal $\rho(n-1, n-1, x, y)$ with $x = r_{n-1}^{\pi''}$ and $y = a$ if $a > 0$; $x = r_{n-1}^{\pi''} + a$ and $y = 0$ otherwise), and, if $a \neq 0$, apply $\rho(n-1, n-1, x, y)$, with $x = r_{n-1}^{\pi''} + b$ and $y = a$ if $a > 0$; $x = r_{n-1}^{\pi''} + b + a$

and $y = 0$ otherwise (resp. apply by the 2-reversal $\rho(n{-}1, n, x + y + b, 0)$ balancing r_{n+1}). We applied $1{+}\varphi(\pi)$ operations here. Let $\pi''' = (2\ \ldots\ 1\ n)$ be the resulting permutation, with $(n{-}2)$ inversions and two components: one with all intergenic regions $r_i^{\pi'''}$, for $1 \leq i \leq n$, and one with the intergenic region $r_{n+1}^{\pi'''}$ only.

(iv) Move element 1 from position $(n{-}1)$ to position 1 by a sequence of reversals $\rho(i, i+1, 0, k - r'_{i+2})$ such that k is the length of the intergenic region that the current 2-reversal is cutting in the right. We will apply $(n-2)$ 2-reversals, removing the same amount of inversions. This step goes from 2 to $2 + (n-1) = m$ components since each 2-reversal here creates a new component, except for the last one that creates two new components.

Summing up, we apply $inv(\pi) - 2n + 3$ reversals in (i), $n - 2$ reversals in (ii), $1 + \varphi(\pi)$ reversals in (iii), and $n - 2$ reversals in (iv), which gives us exactly the minimum amount of $(inv(\pi) + \varphi(\pi))$ operations. □

We can use Lemma 7 to obtain a 2-approximation algorithm for permutations π with $n \geq 9$ elements and $inv(\pi) \geq 4n$, as explained in the next lemma.

Lemma 8. *If a permutation π with $n \geq 9$ elements has $inv(\pi) \geq 4n$ then the value of $d(\pi)$ is 2-approximable.*

Proof. Suppose that we have a permutation π with $n \geq 9$ such that $inv(\pi) \geq 4n$. By Lemma 4, we have that $d(\pi) \geq inv(\pi)$. Consider the following procedure, divided into three steps:

(i) Apply a sequence of k super short reversals that moves the element n on π to position 1, without exchanging any intergenic region (i.e., any super short reversal $\rho(i, i{+}1, x, y)$ applied here has $x = r_i^\pi$ and $y = 0$, keeping r^π intact). Let π' be the resulting permutation. Since π has n elements, we have that $k < n$ and $inv(\pi') < inv(\pi) + n$, regardless of the position of element n in π.

(ii) Apply a sequence of k' super short reversals in a similar way as above that moves element 1 from π' to position n. Let π'' be the resulting permutation. Since π' has n elements, and since element 1 cannot be at position 1 in π' ($\pi'_1 = n$), it follows that $k' < n - 1$ and $inv(\pi'') < inv(\pi') + n - 1 < inv(\pi) + 2n - 1$, regardless of the position of element 1 in π'.

(iii) Use the algorithm presented in Lemma 7 to sort π''.

Note that the first two steps apply $(k + k') < (2n - 1)$ super short reversals, and Step (iii) applies up to $inv(\pi) + 2n$ super short reversals, so the procedure above applies z super short reversals such that $z \leq 2n{-}1{+}inv(\pi){+}2n = inv(\pi){+}4n{-}1$. Since $inv(\pi) \geq 4n$, we have that $z \leq 2inv(\pi)$, and the lemma follows. □

4 Conclusion

In this paper, we analyzed the minimum number of super short reversals needed to sort a permutation π and transform its intergenic regions r^{π} according to the set of intergenic regions r^{ι} of the target genome represented by ι. We defined some bounds that allowed us to state three different algorithms: a more general that guarantees an approximation factor of 3; an exact algorithm for any permutation π with $n > 1$ elements such that $\pi_1 = n$ and $\pi_n = 1$; and a more specific one that sorts any permutation π with $n \geq 9$ elements such that $inv(\pi) \geq 4n$ with an approximation factor of 2. We intend to investigate the problem using super short transpositions instead of super short reversals, as well as using these operations together on signed permutations. We will also study the complexity of all these variants of the problem.

Acknowledgments. This work was supported by the National Council for Scientific and Technological Development - CNPq (grants 400487/2016-0, 425340/2016-3, and 140466/2018-5), the São Paulo Research Foundation - FAPESP (grants 2013/08293-7, 2015/ 11937-9, 2017/12646-3, 2017/16246-0, and 2017/16871-1), the Brazilian Federal Agency for the Support and Evaluation of Graduate Education - CAPES, and the CAPES/COFECUB program (grant 831/15).

References

1. Biller, P., Guéguen, L., Knibbe, C., Tannier, E.: Breaking good: accounting for fragility of genomic regions in rearrangement distance estimation. Genome Biol. Evol. **8**(5), 1427–1439 (2016)
2. Biller, P., Knibbe, C., Beslon, G., Tannier, E.: Comparative genomics on artificial life. In: Beckmann, A., Bienvenu, L., Jonoska, N. (eds.) CiE 2016. LNCS, vol. 9709, pp. 35–44. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-40189-8_4
3. Bulteau, L., Fertin, G., Tannier, E.: Genome rearrangements with indels in intergenes restrict the scenario space. BMC Bioinform. **17**(S14), 225–231 (2016)
4. Caprara, A.: Sorting permutations by reversals and eulerian cycle decompositions. SIAM J. Discret. Math. **12**(1), 91–110 (1999)
5. Chen, T., Skiena, S.S.: Sorting with fixed-length reversals. Discret. Appl. Math. **71**(1–3), 269–295 (1996)
6. Fertin, G., Jean, G., Tannier, E.: Algorithms for computing the double cut and join distance on both gene order and intergenic sizes. Algorithms Mol. Biol. **12**(16), 1–11 (2017)
7. Fertin, G., Labarre, A., Rusu, I., Tannier, E., Vialette, S.: Combinatorics of Genome Rearrangements. Computational Molecular Biology. The MIT Press, London (2009)
8. Galvão, G.R., Baudet, C., Dias, Z.: Sorting circular permutations by super short reversals. IEEE/ACM Trans. Comput. Biol. Bioinform. **14**(3), 620–633 (2017)
9. Galvão, G.R., Lee, O., Dias, Z.: Sorting signed permutations by short operations. Algorithms Mol. Biol. **10**(12), 1–17 (2015)
10. Hannenhalli, S., Pevzner, P.A.: Transforming men into mice (polynomial algorithm for genomic distance problem). In: Proceedings of the 36th Annual Symposium on Foundations of Computer Science (FOCS 1995), pp. 581–592. IEEE Computer Society Press, Washington, DC (1995)

11. Jerrum, M.R.: The complexity of finding minimum-length generator sequences. Theor. Comput. Sci. **36**(2–3), 265–289 (1985)
12. Knuth, D.E.: The art of Computer Programming: Fundamental Algorithms. Addison-Wesley, Reading (1973)
13. Oliveira, A.R., Fertin, G., Dias, U., Dias, Z.: Sorting signed circular permutations by super short operations. Algorithms Mol. Biol. **13**(13), 1–16 (2018)
14. Yancopoulos, S., Attie, O., Friedberg, R.: Efficient sorting of genomic permutations by translocation, inversion and block interchange. Bioinformatics **21**(16), 3340–3346 (2005)