Preprint
For final version
Journal of Computer and System Sciences **65** (2002) 587-609

EFFICIENT ALGORITHMS FOR MULTICHROMOSOMAL GENOME REARRANGEMENTS

GLENN TESLER

Abstract. Hannenhalli and Pevzner [5] gave a polynomial time algorithm for computing the minimum number of reversals, translocations, fissions, and fusions, that would transform one multichromosomal genome to another when both have the same set of genes without repeats. We fixed some problems with the construction: (1) They claim it can exhibit such a sequence of steps, but there was a gap in the construction. (2) Their construction had an asymmetry in the number of chromosomes in the two genomes, whereby forwards scenarios could have fissions but not fusions.

We also improved the speed by combining the algorithm with the algorithm of Bader, Moret, and Yan [2] that produces reversal scenarios for permutations in linear time.

1. Introduction

Hannenhalli and Pevzner [5] give a polynomial time algorithm `genomic_sort` for computing the distance between two multichromosomal genomes, where the distance is the minimum number of reversals, translocations, fissions, and fusions required to transform one genome to the other. An abridged version of that paper appears in [10, Ch. 10, pp. 214–226]. We have implemented this algorithm in full in a program GRIMM [12] available on the web [11], and are reporting additional details that are necessary to complete the algorithm:

(1) They say that their algorithm can exhibit an optimal sequence of transformation steps, *but they do not actually do this:* there is a gap in their reduction of the multichromosomal problem to the unichromosomal problem of "sorting by reversals" (where algorithms for efficient generation of such scenarios are known). It is sometimes necessary to reorder and flip certain chromosomes of *both* multichromosomal genomes to form the permutations used in the unichromosomal problem, but they do not reorder either one. They acknowledge flips are required in one genome [5, Lemma 2], but do not say when to do them, and they do not indicate that flips may be required in the other genome. Fixing all of this considerably complicates step 19 of their algorithm `genomic_sort`.

Due to this gap, the example of a rearrangement scenario that they provide ([5, p. 588], part (f)) was produced in an ad hoc fashion, and is not consistent with the "capping" produced by their algorithm (part (e)), as we will show in Section 7.2.

Key words and phrases. genome rearrangements, fusion, fission, translocation, reversal, inversion, breakpoint graph, genes, chromosomes, homology.

We will close the gap and prove the following improvement to their algorithm (see Sections 3–5):

Theorem 1. *Let $d = d(\Pi, \Gamma)$ denote the distance between two multichromosomal genomes, Π and Γ. There is a constructive algorithm to produce two permutations π^*, γ^* whose reversal distance is $d_{\text{rev}}(\pi^*, \gamma^*) = d$ or $d + 1$, such that optimal reversal scenarios between these permutations directly mimic optimal rearrangement scenarios between genomes Π and Γ. All of this takes polynomial time. When $d_{\text{rev}}(\pi^*, \gamma^*) = d + 1$, one reversal step mimics flipping a block of consecutive whole chromosomes, which does not count as an operation in a multichromosomal rearrangement scenario; there are examples when such a step is required.*

(2) Although the distance is symmetric ($d(\Pi, \Gamma) = d(\Gamma, \Pi)$), when the genomes have different numbers of chromosomes their algorithm requires that it be computed as $d(\Pi, \Gamma)$ where Π has fewer chromosomes than Γ. Thus, it may be necessary to swap the genomes to achieve this, and a rearrangement scenario derived from the resulting breakpoint graph would be backwards.

We determined and added the necessary steps to the procedure to compute the rearrangement distance regardless of which genome has more chromosomes, and adjusted their distance formula accordingly; see Section 2.4 for the formula and Section 4 for the proof.

(3) We combined this algorithm with the Bader, Moret, Yan [2] linear-time algorithm for computing reversal distance in unichromosomal genomes, thus reducing the time to compute distance to $O(n)$ and the time to compute a rearrangement scenario to $O(n^2)$ (where n is the total number of "markers" in the reduction: the number of genes plus twice the number of chromosomes in the genome with more chromosomes); see Section 3.2.

(4) We prove a heuristic for selecting good reversals based on breakpoints, in Section 6.2. The heuristic is not theoretically optimal for producing pairwise rearrangement scenarios, but is fast in practice, and generalizes to phylogenetic trees involving more than two genomes. It is used by MGR, a program for constructing phylogenetic trees (Bourque and Pevzner [3]).

2. Review of Notation and Terminology

Hannenhalli and Pevzner published algorithms for computing reversal distance and optimal reversal scenarios in unichromosomal genomes [6], and reversal distance in multichromosomal genomes [5]. These were later merged together into a unified treatment, and published in [10, Ch. 10]. We review the necessary terminology from these sources.

2.1. **Genes, chromosomes, genomes.** We represent genes by numbers $1, \ldots, N_g$, and indicate the orientation (strand) of each gene by a $\pm$ sign. A chromosome $\vec{a} = \langle a_1, \ldots, a_k \rangle$ is a sequence of signed numbers, and the *flip* of a chromosome is $-\vec{a} = \langle -a_k, \ldots, -a_1 \rangle$. In studies of rearrangements on unichromosomal genomes, several types of chromosomes have been considered ([2], [6], [7], [9, p. 208]) but only the first type below is biologically relevant for multichromosomal genomes:

(1) *Undirected linear chromosomes*: $\vec{a}$ and $-\vec{a}$ are regarded as equivalent.
(2) *Directed linear chromosomes*: $\vec{a}$ and $-\vec{a}$ are regarded as different.

(3) *Circular chromosomes* are equivalent under a dihedral action; all k circular shifts $\langle a_i, \ldots, a_k, a_1, \ldots, a_{i-1}\rangle$ of $\vec{a}$, and all k circular shifts of $-\vec{a}$, are regarded as equivalent.

In the remainder of this paper, we only consider multichromosomal genomes with undirected linear chromosomes. We regard a genome as a set $\Pi = \{\pi(1), \ldots, \pi(N_c)\}$ of N_c chromosomes partitioning genes $1, \ldots, N_g$, where $\pi(i) = \langle \pi(i)_1, \ldots, \pi(i)_{n_i}\rangle$ is the sequence of signed genes in the ith chromosome. Each gene $j = 1, \ldots, N_g$ occurs exactly once in the genome, either as j or as $-j$. All genomes in any problem will be defined on a common set of genes, since we do not consider insertions, deletions, or duplications.

We introduce additional markers called *caps*: $C_k = N_g + k$ for $k = 1, 2, \ldots, 2N_c$. These will serve as chromosome delimeters when we convert the genome into a single permutation. This gives a total of $n = N_g + 2N_c$ markers. A *capping* of a chromosome $\pi(i)$ is

$$\hat{\pi}(i) = \langle \pi(i)_0, \pi(i)_1, \ldots, \pi(i)_{n_i}, \pi(i)_{n_i+1}\rangle$$

where $\pi(i)_0, \pi(i)_{n_i+1}$ are signed caps, where the signs will be given in Lemma 2. $\pi(i)_0$ is called an *lcap* and $\pi(i)_{n_i+1}$ is called an *rcap*. A capping of a genome Π is

$$\hat{\Pi} = \{\hat{\pi}(1), \ldots, \hat{\pi}(N_c)\},$$

where each cap $C_1, \ldots, C_{2N_c}$ appears (with a suitable sign) exactly once. There are $(2N_c)!$ possible cappings. One capping is $\hat{\pi}(i) = \langle C_{2i-1}, \pi(i)_1, \ldots, \pi(i)_{n_i}, C_{2i}\rangle$.

A *concatenate* of $\hat{\Pi}$ is a signed permutation $\hat{\pi}$ of $1, 2, \ldots, n$, formed by choosing one of the $N_c!$ orderings and one of the 2^{N_c} flippings of the chromosomes, and concatenating them together; if we relabel the chromosomes after these choices, such a concatenation can be written

$$\begin{aligned} \hat{\pi} &= \hat{\pi}(1) + \cdots + \hat{\pi}(N_c) \\ &= \langle \pi(1)_0, \pi(1)_1, \ldots, \pi(1)_{n_1+1}, \ldots, \pi(N_c)_0, \pi(N_c)_1, \ldots, \pi(N_c)_{n_{N_c}+1}\rangle. \end{aligned}$$

Such a signed permutation may also be regarded as a directed linear chromosome. For an example, see Fig. 1(a). Clearly, $\hat{\Pi}$ can be recovered from $\hat{\pi}$ by scanning for caps from left to right, breaking after every other cap.

2.2. **Mimicking multichromosomal rearrangement operations by reversals on a single permutation.** The *reversal* $\rho(i, j)$ on a signed permutation $\pi = \langle \pi_1, \ldots, \pi_k\rangle$ (where $1 \le i \le j \le k$) is

$$\langle \pi_1, \ldots, \pi_{i-1},\ -\pi_j, \ldots, -\pi_i,\ \pi_{i+1}, \ldots, \pi_k\rangle\ .$$

We may also represent this as $\pi = \langle A, B, C\rangle$ and the reversal as $\langle A, -B, C\rangle$, where A, B, C are sequences and B is nonnull.

On multichromosomal genomes, we consider four operations: reversal, translocation, fission, fusion. When we represent a genome by a concatenate, these operations can be mimicked by reversals, but there are also trivial, non-optimal, and nonsensical operations mimicked by reversals. Let $\pi = \langle \pi_1, \ldots, \pi_k\rangle$ and $\sigma = \langle \sigma_1, \ldots, \sigma_m\rangle$ be two chromosomes (without caps).

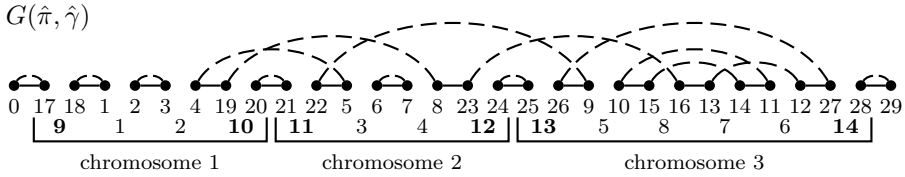
A reversal $\rho(i, j)$ on π is the same as for a signed permutation.

A *translocation* transforms $\pi = \langle A, B\rangle$ and $\sigma = \langle C, D\rangle$ into $\langle A, D\rangle$ and $\langle C, B\rangle$. Certain translocations are given other names, however.

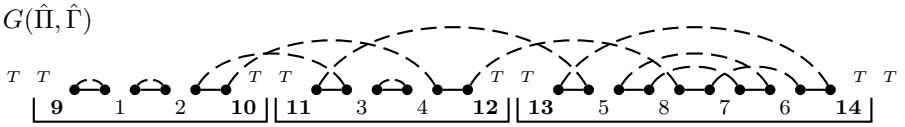
(a)

Genomes	$\Pi = \{\langle 1,2\rangle, \langle 3,4\rangle, \langle 5,8,7,6\rangle\}$	$\Gamma = \{\langle 1,2,3,4\rangle, \langle 5,6,7,8\rangle\}$
Cappings	$\hat{\Pi} = \{\langle \mathbf{9},1,2,\mathbf{10}\rangle, \langle \mathbf{11},3,4,\mathbf{12}\rangle, \langle \mathbf{13},5,8,7,6,\mathbf{14}\rangle\}$	$\hat{\Gamma} = \{\langle \mathbf{9},1,2,3,4,\mathbf{10}\rangle, \langle \mathbf{11},5,6,7,8,\mathbf{12}\rangle, \langle \mathbf{13},\mathbf{14}\rangle\}$
Concatenates	$\hat{\pi} = \langle \mathbf{9},1,2,\mathbf{10},\mathbf{11},3,4,\mathbf{12}, \mathbf{13},5,8,7,6,\mathbf{14}\rangle$	$\hat{\gamma} = \langle \mathbf{9},1,2,3,4,\mathbf{10},\mathbf{11},5, 6,7,8,\mathbf{12},\mathbf{13},\mathbf{14}\rangle$

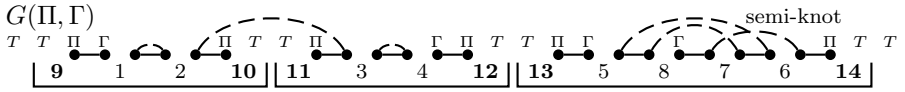
(b) $G(\hat{\pi}, \hat{\gamma})$



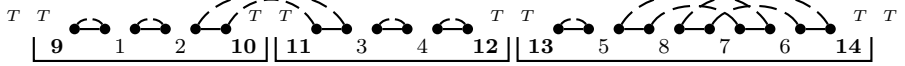
(c) $G(\hat{\Pi}, \hat{\Gamma})$



(d) $G(\Pi, \Gamma)$

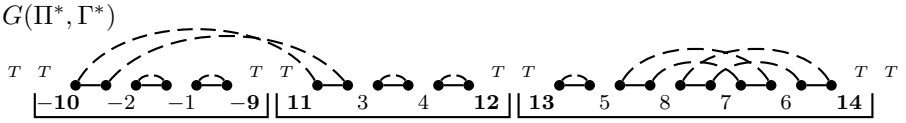


(e) Steps B2–B18

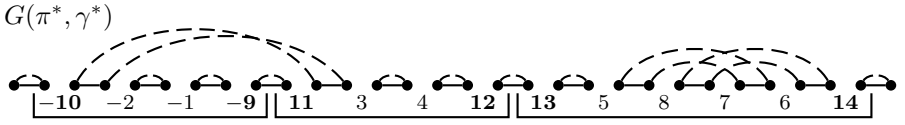


Cappings $\Pi^* = \hat{\Pi}$, $\Gamma^* = \{\langle 9,1,2,3,4,12\rangle, \langle 13,5,6,7,8,14\rangle, \langle 11,10\rangle\}$

(f) $G(\Pi^*, \Gamma^*)$



(g) $G(\pi^*, \gamma^*)$



Concatenates $\pi^* = \langle -\mathbf{10}, -2, -1, -\mathbf{9}, \mathbf{11}, 3, 4, \mathbf{12}, \mathbf{13}, 5, 8, 7, 6, \mathbf{14}\rangle$
$\gamma^* = \langle -\mathbf{10}, -\mathbf{11}, \mathbf{9}, 1, 2, 3, 4, \mathbf{12}, \mathbf{13}, 5, 6, 7, 8, \mathbf{14}\rangle$

(h) Scenario

0. π^*	$\langle -\mathbf{10}$	$-2\ -1\ -\mathbf{9}\rangle\ \langle \mathbf{11}\ 3\ 4\ \mathbf{12}\rangle\ \langle \mathbf{13}\ 5\ 8\ 7\ 6\ \mathbf{14}\rangle$
1. Fusion	$\langle -\mathbf{10}\ -\mathbf{11}\rangle$	$\langle \mathbf{9}\ 1\ 2\ 3\ 4\ \mathbf{12}\rangle\ \langle \mathbf{13}\ 5\ 8\ 7\ 6\ \mathbf{14}\rangle$
2. Reversal	$\langle -\mathbf{10}\ -\mathbf{11}\rangle$	$\langle \mathbf{9}\ 1\ 2\ 3\ 4\ \mathbf{12}\rangle\ \langle \mathbf{13}\ 5\ 8\ -6\ -7\ \mathbf{14}\rangle$
3. Reversal	$\langle -\mathbf{10}\ -\mathbf{11}\rangle$	$\langle \mathbf{9}\ 1\ 2\ 3\ 4\ \mathbf{12}\rangle\ \langle \mathbf{13}\ 5\ 6\ -8\ -7\ \mathbf{14}\rangle$
4. Reversal: γ^*	$\langle -\mathbf{10}\ -\mathbf{11}\rangle$	$\langle \mathbf{9}\ 1\ 2\ 3\ 4\ \mathbf{12}\rangle\ \langle \mathbf{13}\ 5\ 6\ 7\ 8\ \mathbf{14}\rangle$

FIGURE 1. An overview of the whole procedure on two genomes, with $N_g = 8$ genes, $N_c = 3$ chromosomes, $n = 14$ markers. Caps are in bold. Exposed vertices are marked T (tails), Π (Π-caps), Γ (Γ-tails). Graph parameters: $b = 9$, $c = 6$, $p_{\Pi\Pi} = 1$, $p_{\Gamma\Gamma} = 0$, $s = 1$, $rr = gr = fr = 0$. Distance: $d = 9 - 6 + 0 + 0 + \lceil \frac{1-0+0}{2} \rceil = 4$. Key differences from Hannenhalli-Pevzner algorithm: (a–d) $\hat{\Gamma}$ has $\nu = 1$ null, $\langle \mathbf{13}, \mathbf{14}\rangle$. (f) Step B19: Flip chromosome 1 to properly orient chromosomes 1, 2. All bonds are proper. (g) Step B19: Add tails to obtain $G(\pi^*, \gamma^*)$.

Reversal	⟨**8**	1	2	3	4	**9**⟩	⟨**10**	5	6	7	**11**⟩	⟨**12**	**13**⟩
Translocation	⟨**8**	1	−4	−3	−2	**9**⟩	⟨**10**	5	6	7	**11**⟩	⟨**12**	**13**⟩
Fission	⟨**8**	1	−6	−5	−**10**⟩	⟨−**9**	2	3	4	7	**11**⟩	⟨**12**	**13**⟩
Fusion	⟨**8**	1	−6	−**12**⟩	⟨−**11**	−7	−4	−3	−2	**9**⟩	⟨**10**	5	**13**⟩
Block flip	⟨**8**	**11**⟩	⟨**12**	6	−1	−7	−4	−3	−2	**9**⟩	⟨**10**	5	**13**⟩
Cap exchange	⟨**8**	**11**⟩	⟨−**13**	−5	−**10**⟩	⟨−**9**	2	3	4	7	1	−6	−**12**⟩
Cap exchange	⟨**8**	**13**⟩	⟨−**11**	−5	−**10**⟩	⟨−**9**	2	3	4	7	1	−6	−**12**⟩
Nonsense	⟨**8**	**13**⟩	⟨−**11**	6	−1	−7	−4	−3	−2	**9**⟩	⟨**10**	5	−**12**⟩
	⟨**8**	**13**⟩	1	−6	**11**⟩	−7	−4	−3	−2	**9**⟩	⟨**10**	5	−**12**⟩

FIGURE 2. Mimicking multichromosomal rearrangements by reversals. The segment to reverse is underlined. Genes are $1, \ldots, 7$ and caps are $\mathbf{8}, \ldots, \mathbf{13}$. For clarity, chromosome delimiters $\langle\rangle$ are shown. Null chromosomes are just a pair of caps, such as $\langle \mathbf{12\ 13} \rangle$. Note the fission illustrated is between nonadjacent chromosomes, which has the side effect of flipping intermediate chromosomes. The other interchromosomal operations are shown on adjacent chromosomes, but would also flip intermediate chromosomes if done on nonadjacent ones.

The *fusion* of π and σ is $\langle \pi_1, \ldots, \pi_k, \sigma_1, \ldots, \sigma_m \rangle$. It may be viewed as the translocation between $\langle \pi, \emptyset \rangle$ and $\langle \emptyset, \sigma \rangle$ resulting in $\langle \pi, \sigma \rangle$ and a null chromosome $\langle \emptyset, \emptyset \rangle$.

A *fission* on π results in $A = \langle \pi_1, \ldots, \pi_{i-1} \rangle$ and $B = \langle \pi_i, \ldots, \pi_k \rangle$; there is one fission for each $1 < i \leq k$. It may also be regarded as the translocation $\langle A, B \rangle, \langle \emptyset, \emptyset \rangle \to \langle A, \emptyset \rangle, \langle \emptyset, B \rangle$.

As shown in [5], all of these may be mimicked by reversals in a suitable capped concatenate of the chromosomes; see Fig. 2.

Let $\langle \pi_1, \ldots, \pi_n \rangle$ be a capped concatenate of genome Π, and $\langle \pi'_1, \ldots, \pi'_n \rangle$ be the result of a reversal $\rho(i, j)$, with $i \leq j$.

If π_i is an lcap but π_j is not an rcap, or if π_j is an rcap but π_i is not an lcap, the reversal is nonsensical; it leaves two left halves or two right halves of chromosomes, as shown in Fig. 2.

If π_i is an lcap in chromosome r and π_j is an rcap in chromosome s ($1 \leq r \leq s \leq N_c$), the reversal mimics flipping a block of chromosomes, to change the concatenate from $\hat{\pi}(1) + \cdots + \hat{\pi}(N_c)$ to

$$\hat{\pi}(1) + \cdots + \hat{\pi}(r-1) - \hat{\pi}(s) - \hat{\pi}(s-1) - \cdots - \hat{\pi}(r) + \hat{\pi}(s+1) + \cdots + \hat{\pi}(N_c).$$

Although this does not count as an operation in computing $d(\Pi, \Gamma)$, it is sometimes necessary to perform this operation when mimicking a multichromosomal rearrangement scenario by a permutation reversal scenario. This is because for any two nonnull chromosomes in a given concatenate, only two of the four fusions and only half of the translocations between them can be mimicked by a reversal. Flipping either chromosome allows the other half of these type of events to be mimicked. If nonoptimal concatenates are chosen for the mimicking procedure, this step will be required often, but with optimal concatenates, it will be required at most once.

If π_{i-1} is an lcap and π_{j+1} is an rcap, or π_i is an rcap and π_j is an lcap, the reversal changes the assignments of two caps. Call this a *cap exchange*. These operations are only necessary when nonoptimal cappings are chosen. We will find optimal cappings such that these operations are never used.

Reversals not covered by the above mimic valid rearrangement operations: if π_i and π_j are in the same chromosome, it is a reversal, and if they are in different chromosomes, it is a fission, fusion, or translocation.

Then, given any capped concatenates $\hat{\pi}, \hat{\gamma}$ of genomes Π, Γ, the number of steps in a scenario sorting $\hat{\pi}$ into $\hat{\gamma}$ by permitted reversals is

$$d(\Pi, \Gamma) \ + \ \#\text{ of block flips } + \ \#\text{ of cap exchanges.} \tag{1}$$

In all cases except the nonsensical ones, a reversal encompasses an even number of caps, of alternating types (lcap or rcap). The reversal turns each lcap it encompasses into an rcap and vice-versa, and also inverts the sign of each cap. This leads to the following conventions for the signs of lcaps and rcaps.

Lemma 2. *Let $\hat{\pi}$ be a concatenate of $\hat{\Pi}$ given by $\hat{\pi} = \hat{\pi}(1)+\cdots+\hat{\pi}(N_c)$ with capping $\hat{\pi}(i) = \langle C_{2i-1}, \pi(i)_1, \ldots, \pi(i)_{n_i}, C_{2i}\rangle$. Apply a sequence of permitted reversals to $\hat{\pi}$. Then the caps at every step have the following signs:*

(a) *Each lcap has the form $+C_j$ with j odd or $-C_j$ with j even, i.e., $(-1)^{j+1}C_j$.*
(b) *Each rcap has the form $-C_j$ with j odd or $+C_j$ with j even, i.e., $(-1)^j C_j$.*

2.3. **Breakpoint graph.** We review a series of graphs defined in [5]. See Fig. 1(a)–(d).

Let $\hat{\pi}$ be a signed permutation of $1, \ldots, n$. It may be transformed to an unsigned permutation $u(\hat{\pi}) = \langle \hat{\pi}_0, \ldots, \hat{\pi}_{2n+1}\rangle$ of $0, 1, \ldots, 2n, 2n+1$, by replacing each positive entry $+x$ with $2x-1, 2x$, each negative entry $-x$ with $2x, 2x-1$, and then prepending $\hat{\pi}_0 = 0$ and appending $\hat{\pi}_{2n+1} = 2n+1$.

Let Π and Γ be two genomes on the same N_g genes. They may have different numbers of chromosomes; add null chromosomes to the genome with fewer chromosomes so that they both have N_c chromosomes. (We can also accommodate null chromosomes in both genomes simultaneously.) Choose any cappings $\hat{\Pi}, \hat{\Gamma}$, and any concatenates $\hat{\pi}, \hat{\gamma}$, and transform them to unsigned permutations as described above.

The breakpoint graph $G(\hat{\pi}, \hat{\gamma})$ on $2n+2$ vertices $0, 1, \ldots, 2n+1$, is defined as follows. Arrange the vertices from left to right in the order $\hat{\pi}_0, \hat{\pi}_1, \ldots, \hat{\pi}_{2n}, \hat{\pi}_{2n+1}$. Form a black edge $\{\hat{\pi}_{2i}, \hat{\pi}_{2i+1}\}$ and a gray edge $\{\hat{\gamma}_{2i}, \hat{\gamma}_{2i+1}\}$, for $i = 0, \ldots, n$.

Next we define a graph $G(\hat{\Pi}, \hat{\Gamma})$ that depends only on the cappings $\hat{\Pi}, \hat{\Gamma}$, not on the concatenates $\hat{\pi}, \hat{\gamma}$. It is formed from $G(\hat{\pi}, \hat{\gamma})$ by deleting the edges arising from the rcap of one chromosome and the lcap of the next, in either $\hat{\pi}$ or $\hat{\gamma}$. Specifically, delete the vertices $0, 2N_g+1, 2N_g+4, 2N_g+5, 2N_g+8, 2N_g+9, \ldots, 2n-4, 2n-3, 2n, 2n+1$ and the black and gray edges incident on them. These vertices are called *tails*.

Finally, we define a graph $G(\Pi, \Gamma)$ that does not depend on the capping of Γ, by deleting from $G(\hat{\Pi}, \hat{\Gamma})$ the gray edges incident on vertices $2N_g+2, 2N_g+3, 2N_g+6, 2N_g+7, \ldots, 2n-2, 2n-1$. These vertices are called Π-*caps*. The vertex on the other end of the deleted gray edge is called a Γ-*tail*, unless the gray edge arises from a null chromosome of Γ, in which case both its ends are Π-caps and deletion does not introduce a Γ-tail. Note that the construction [5, p. 586] does not consider the

possibility of null chromosomes in Γ; see Section 4 for further details. (Also note that while the vertex labeling depends on the capping $\hat{\Pi}$, the graph does not.)

Let $\nu(\Gamma)$ be the number of null chromosomes in Γ. Then $G(\Pi, \Gamma)$ has $2(N_g + N_c)$ vertices, including $2N_c$ Π-caps and $2(N_c - \nu)$ Γ-tails, and has $b(\Pi, \Gamma) = N_g + N_c$ black edges and $N_g - N_c + \nu$ gray edges.

Each of the $(2N_c)!/(2^\nu \nu!)$ possible cappings $\hat{\Gamma}'$ corresponds to adding $2N_c - \nu$ gray edges to $G(\Pi, \Gamma)$, $2(N_c - \nu)$ of which join a Π-cap and a Γ-tail, and the remaining ν of which join two Π-caps.

Every vertex in $G(\Pi, \Gamma)$ has degree 1 or 2, so the graph consists of vertex-disjoint cycles and paths. Let $c(\Pi, \Gamma)$ be the total number of cycles and paths. Each path is classified as a $\Pi\Pi$-path, $\Gamma\Gamma$-path, or $\Pi\Gamma$-path, according as its endpoints are both Π-caps; both Γ-tails; or one of each. Let $p_{\Gamma\Gamma}(\Pi, \Gamma)$ be the number of $\Gamma\Gamma$ paths, and similarly for $p_{\Pi\Pi}$, $p_{\Pi\Gamma}$. In [5, p. 587], $p_{\Gamma\Gamma} = p_{\Pi\Pi}$ and this parameter is simply called p; however, we have extended the algorithm to handle the case when Π has more chromosomes than Γ, and this extension may cause $p_{\Gamma\Gamma} < p_{\Pi\Pi}$ (specifically, $p_{\Gamma\Gamma} = p_{\Pi\Pi} - \nu$).

2.4. **Hurdles and relatives.** We now review the definition of the *interleaving graph* of $G = G(\Pi, \Gamma)$. See Fig. 3.

Gray edges $\{\hat{\pi}_i, \hat{\pi}_j\}$ and $\{\hat{\pi}_k, \hat{\pi}_\ell\}$ are *interleaving* when the intervals $[i, j]$ and $[k, \ell]$ overlap, but neither interval contains the other. Cycles or paths CP_1, CP_2 *interleave* when there are interleaving edges $g_1 \in CP_1$, $g_2 \in CP_2$.

The *interleaving graph* $I(G)$ is a new graph, with one vertex for each path or cycle in G, excluding *adjacencies* (2-cycles of the breakpoint graph) and *bare edges* (paths consisting of a single black edge and no gray edges). $I(G)$ has an edge between each pair of vertices that correspond to interleaving elements (paths or cycles) of the breakpoint graph.

A gray edge $\{\hat{\gamma}_{2i}, \hat{\gamma}_{2i+1}\} = \{\hat{\pi}_j, \hat{\pi}_k\}$ in the breakpoint graph is *oriented* when $|k - j|$ is even and *unoriented* when $|k - j|$ is odd. It is *intrachromosomal* when $\hat{\pi}_j$, $\hat{\pi}_k$ arise from the same chromosome of $\hat{\Pi}$, and *interchromosomal* otherwise.

A connected component of the interleaving graph is *oriented* when any of its vertices corresponds to a path or cycle with an oriented edge in the breakpoint graph, and is *unoriented* otherwise. Similarly, it is *interchromosomal* if any of the corresponding edges of the breakpoint graph are interchromosomal, and is *intrachromosomal* otherwise.

The *extent* of a connected component K of the interleaving graph is $[i, j]$, where $\hat{\pi}_i$ and $\hat{\pi}_j$ are the leftmost and rightmost vertices of any paths or cycles of K in G. The component K is *real* when it is intrachromosomal and none of the vertices $\hat{\pi}_i, \hat{\pi}_{i+1}, \ldots, \hat{\pi}_j$ in G are Π-caps or Γ-tails.

The set of unoriented components is denoted $\mathcal{U}(G)$. A *hurdle*, *greatest hurdle*, and *superhurdle*, are unoriented components satisfying additional conditions, and an interleaving graph is a *fortress* when the set of hurdles satisfies still more conditions; see [6]. On restricting the interleaving graph to the set of intrachromosomal unoriented components ($\mathcal{IU}(G)$) or to real unoriented components ($\mathcal{RU}(G)$), we obtain the generalizations of these terms shown in Table 1. Note that the number and orientation of interchromosomal components may depend on the concatenates used to construct the graph, but this is not so for the intrachromosomal components; thus $\mathcal{IU}(G)$ and $\mathcal{RU}(G)$ do not depend on the original choice of concatenates.

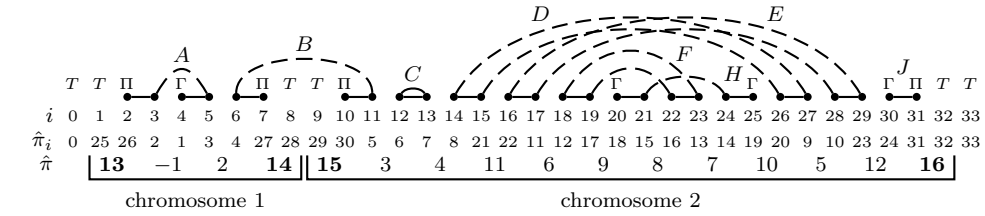


FIGURE 3. A graph $G(\Pi, \Gamma)$. A, J are $\Pi\Gamma$-paths, B is a $\Pi\Pi$-path, H is a $\Gamma\Gamma$-path, and C, D, E, F are cycles. Gray edge $\{\hat{\pi}_{21}, \hat{\pi}_{24}\} = \{15, 14\}$ interleaves with both $\{\hat{\pi}_{19}, \hat{\pi}_{22}\} = \{17, 16\}$ and $\{\hat{\pi}_{18}, \hat{\pi}_{23}\} = \{12, 13\}$, but no others. Thus, cycle F and path H interleave. Also, cycles D and E interleave. The interleaving graph has components $K_1 = \{A\}$, $K_2 = \{B\}$, $K_3 = \{D, E\}$, $K_4 = \{F, H\}$. Since C is an adjacency and J is a bare edge, they are not included in the interleaving graph. Gray edge $\{4, 5\}$ in B is interchromosomal unoriented and $\{2, 3\}$ in A is intrachromosomal oriented. All other gray edges are intrachromosomal unoriented. Component K_1 is intrachromosomal oriented, K_2 is interchromosomal unoriented, and K_3, K_4 are intrachromosomal unoriented, so $\mathcal{IU} = \{K_3, K_4\}$. None of the intrachromosomal components K_1, K_3, K_4 are real (so $\mathcal{RU} = \emptyset$); note that even though K_3 itself does not have Γ-caps, the Γ-caps $\hat{\pi}_{20} = 18$, $\hat{\pi}_{25} = 19$ are within its extent $[14, 29]$. Adjacency $\{C\}$ is real but is not regarded as a component.

Properties of $\mathcal{U}(G)$:			
hurdle	greatest hurdle	super-hurdle	fortress
Analogous properties of $\mathcal{IU}(G)$:			
knot	greatest knot	super-knot	fortress-of-knots
Analogous properties of $\mathcal{RU}(G)$:			
real-knot	greatest real-knot	super-real-knot	fortress-of-real-knots

TABLE 1. Hurdlemania

In addition, a *semi-knot* is a knot that isn't a real-knot, and whose extent does not encompass any $\Pi\Pi$- or $\Gamma\Gamma$-path. Since it's not real, it has at least one $\Pi\Gamma$-path. The number of semi-knots in $G(\Pi, \Gamma)$ is denoted $s(\Pi, \Gamma)$.

The construction of the interleaving graph of $G(\Pi, \Gamma)$, and the classification of its components, may be applied to other variations of the breakpoint graph.

A component is *simple* when it contains a $\Pi\Gamma$-path but is not a semi-knot. The graph $\overline{G} = \overline{G}(\Pi, \Gamma)$ is formed by closing all $\Pi\Gamma$-paths in simple components of $G(\Pi, \Gamma)$. Parameters *fr*, *gr*, *rr* are defined in [5, p. 589] in terms of the real-knots of $\overline{G}$. We need to elaborate on *gr* only: it is 1 if $\overline{G}$ has the greatest real-knot and $s(\Pi, \Gamma) > 0$, and is 0 otherwise.

The distance formula given in [5, Thm. 4] only applies when the number of chromosomes in Π is less or equal to the number in Γ. By changing their p to

`genomic_sort_B`(Π, Γ)

B1. Construct the graph $G = G(\Pi, \Gamma)$, and compute parameters $b(\Pi, \Gamma)$, $c(\Pi, \Gamma)$, $p_{\Gamma\Gamma}(\Pi, \Gamma)$, and $s = s(\Pi, \Gamma)$ (Section 3.2).
B2. Close all $\Pi\Gamma$-paths in simple components, and compute parameters $rr(\Pi, \Gamma)$, $fr(\Pi, \Gamma)$, $gr(\Pi, \Gamma)$.
B3. Compute the distance from these parameters (Equation (2)).
B4. Close all but one $\Pi\Gamma$-path in components with more than one $\Pi\Gamma$-path.
B5. **while** G contains a $\Gamma\Gamma$-path
B6. find an interchromosomal or an oriented edge g joining this $\Gamma\Gamma$-path with a $\Pi\Pi$-path;
B7. add edge g to the graph and close the resulting path (Section 3.1)
B8. Close all remaining $\Pi\Pi$-paths (Section 4).
B9. **while** $s > 2$
B10. join two $\Pi\Gamma$-paths from different semi-knots into one cycle, by adding the two $\Pi\Gamma$-edges; (Section 3.1)
B11. $s = s - 2$.
B12. **if** $s = 2$ and $gr = 0$,
B13. join two $\Pi\Gamma$-paths from different semi-knots into one cycle, by adding the two $\Pi\Gamma$-edges;
B14. $s = s - 2$
B15. **else**
B16. close all s remaining $\Pi\Gamma$-paths;
B17. $s = 0$.
B18. Determine optimal cappings Π^*, Γ^* implied by G.
B19. Determine optimal concatenates π^*, γ^* from the graph (Section 5).
B20. Sort permutation π^* into γ^* by reversals, but interpret each step as a multichromosomal rearrangement transforming Π into Γ (Section 6.1).

FIGURE 4. Genomic sorting algorithm.

$p_{\Gamma\Gamma}$, we obtain a distance formula that is always valid, regardless of how many chromosomes are in each genome, as we will prove in Section 4:

$$d(\Pi, \Gamma) = b(\Pi, \Gamma) - c(\Pi, \Gamma) + p_{\Gamma\Gamma}(\Pi, \Gamma) + rr(\Pi, \Gamma) + \left\lceil \frac{s(\Pi, \Gamma) - gr(\Pi, \Gamma) + fr(\Pi, \Gamma)}{2} \right\rceil . \tag{2}$$

3. The new algorithm

We refer to the steps of `genomic_sort` in [5] as A1–A21. Our new algorithm `genomic_sort_B` is shown in Fig. 4. If only the genomic distance is required, stop at step B3. If the optimal capping is required but not the optimal concatenates, stop at step B18.

3.1. **Joining and closing paths, simplified.** Several steps of `genomic_sort` add an edge to the graph to join two paths into a larger path. The result is always a $\Pi\Gamma$-path with an oriented or interchromosomal edge, and a subsequent iteration of

the main loop of their algorithm closes that path (step A17). We simplify this by adding two edges simultaneously to join these paths into a cycle in a single loop iteration.

The first such steps (A5–A6) join a ΠΠ-path with a ΓΓ-path. The resulting paths never interact with any other path in the main loop, so we separate this out into its own loop (B5–B7). It is also rephrased to account for the new distinction between $p_{\Pi\Pi}$ and $p_{\Gamma\Gamma}$.

The other path joining steps (steps A8 and A13) join two ΠΓ-paths. They proved that at least one of the two possible ΠΓ-edges connecting them is oriented or interchromosomal, and they test the edges to add such an edge first. The other edge is guaranteed to be added in a later iteration. Since the order that they are added does not affect the final output, we remove this test and just add them both at once (steps B10 and B13).

3.2. **Adaptation of the Bader-Moret-Yan algorithm to multichromosomal genomes.** An algorithm was presented in [2] to compute the connected components of the interleaving graph. They implemented it in the file `invdist.c` of `GRAPPA` [1]. We modified it to account for paths (instead of just cycles), deleted tails, and bare edges. The resulting procedure `form_components` runs in time $\Theta(n)$. It identifies the components and computes and stores certain structural information about them (including their leftmost vertex). Say there are cc connected components.

They subsequently determine which components are unoriented (i.e., the set $\mathcal{U}$), and set flags for each component to indicate this. We adapted this to classify components by membership in $\mathcal{U}$, $\mathcal{IU}$, and $\mathcal{RU}$. The resulting procedure `classify_components` runs in time $\Theta(n)$.

Next, they classify and count the number of hurdles, superhurdles, greatest hurdles, and fortresses, by analyzing the stored structural information about the connected components that have been marked as members of $\mathcal{U}$; denote this step `classify_hurdles`($\mathcal{U}$). By modifying this to instead check the new flags for membership in $\mathcal{IU}$ or $\mathcal{RU}$, all the analogous terms in Table 1 can be classified and counted in time $\Theta(cc)$. Combining the results of the calls on $\mathcal{IU}$ and on $\mathcal{RU}$ gives the remaining parameters in the distance formula and other parameters in the algorithm. Thus, we may perform steps B1–B2 of our algorithm as indicated in Fig. 5.

Although the classification of components changes as edges are added in steps B4–B16, there is no need to call these routines again because each added edge changes the classification of components in a known fashion (see [5, Thm. 4] and our extension of that in Section 4, Case 5). No new semi-knots are created, and as they are destroyed, we maintain a count, s. (Also, the number of remaining semi-knots equals the number of remaining ΠΓ-paths in the reorganized code, steps B5–B17.)

The test for the greatest real-knot (A10/B12) simply uses the value of gr from step B2. This is because steps B3–B11 neither create nor destroy components in $\mathcal{RU}$. Each edge added in these steps connects two Π-caps or a Π-cap and a Γ-tail. Any component having either of these within its extent was not real to begin with, and is either oriented or interchromosomal after the addition of the edge.

In step B19, it may be necessary to form and classify the components again, perhaps multiple times, because the number and orientation of interchromosomal components can change. This will be described in Section 5.

B1. Construct the graph $G = G(\Pi, \Gamma)$.
Compute parameters $b(\Pi, \Gamma)$, $c(\Pi, \Gamma)$, $p(\Pi, \Gamma)$.
`form_components`(G)
`classify_components`(G)
`classify_hurdles`$(\mathcal{IU}(G))$
`classify_hurdles`$(\mathcal{RU}(G))$
Combine the results to compute parameter $s(\Pi, \Gamma)$ and to identify simple components.

B2. Form $\overline{G}$ by closing all $\Pi\Gamma$-paths in simple components.
`classify_components`$(\overline{G})$
`classify_hurdles`$(\mathcal{IU}(\overline{G}))$
`classify_hurdles`$(\mathcal{RU}(\overline{G}))$
Combine the results to determine parameters $rr(\Pi, \Gamma)$, $fr(\Pi, \Gamma)$, $gr(\Pi, \Gamma)$.

FIGURE 5. Adaptation of BMY algorithm for multichromosomal genomes.

4. When Γ has fewer chromosomes than Π

The original construction of $G(\Pi, \Gamma)$ [5, p. 586] assumes that Π has no more chromosomes than Γ, and then says to pad Π with null chromosomes so that they both have the same number of chromosomes. However, that construction breaks down without that assumption: if Γ has fewer chromosomes and we pad it with nulls, then when we delete a gray edge corresponding to a null in Γ, the construction leaves unresolved how to classify the vertices of the edge into Π-caps and Γ-tails. We have said both vertices should be classified as Π-caps in this case. This causes the parameters $p_{\Gamma\Gamma}, p_{\Pi\Pi}$ to be unequal (instead of equal, as they were in [5]), so the distance formula was changed to (2). It also requires rephrasing steps A5–A6 (see B5–B7), and introducing a new step B8.

We have done all this to make the construction truly symmetric, regardless of which genome has more chromosomes. We now explain how to adjust the proofs in [5] to account for these changes.

On page 587, they observe that every cycle in $G(\hat{\Pi}, \hat{\Gamma})$ containing a $\Pi\Pi$-path contains at least one more path, so that $c(\hat{\Pi}, \hat{\Gamma}) \le c(\Pi, \Gamma) - p(\Pi, \Gamma)$ (where their p means $p_{\Pi\Pi}$); that is false if these Π-caps arose in the new way we allow. However, it may be corrected: every cycle in $G(\hat{\Pi}, \hat{\Gamma})$ containing a $\Gamma\Gamma$-path contains at least one more path, so that $c(\hat{\Pi}, \hat{\Gamma}) \le c(\Pi, \Gamma) - p_{\Gamma\Gamma}(\Pi, \Gamma)$. All further references they make to parameter p must be changed to $p_{\Gamma\Gamma}$.

Theorems 3–4 of [5] consider how each graph parameter b, c, p, r, s, fr, gr changes as gray edges $g_1, g_2, \ldots$ are added to the graph $G_0 = G(\Pi, \Gamma)$, and give various upper and lower bounds on the distance, culminating in a proof of the distance formula and the capping algorithm. Let G_i be G_0 plus edges $g_1, \ldots, g_i$. Let $c_i = c(G_i)$, $\Delta c = c_i - c_{i-1}$, and similarly for the other parameters. Let

$$\begin{aligned}
\Delta_i^{(3)} &= \left(c_i - p_{\Gamma\Gamma i} - r_i - \left\lceil \tfrac{s_i}{2} \right\rceil\right) - \left(c_{i-1} - p_{\Gamma\Gamma i-1} - r_{i-1} - \left\lceil \tfrac{s_{i-1}}{2} \right\rceil\right) \\
\Delta_i^{(4)} &= \left(c_i - p_{\Gamma\Gamma i} - r_i - \left\lceil \tfrac{s_i - gr_i + fr_i}{2} \right\rceil\right) \\
&\quad - \left(c_{i-1} - p_{\Gamma\Gamma i-1} - r_{i-1} - \left\lceil \tfrac{s_{i-1} - gr_{i-1} + fr_{i-1}}{2} \right\rceil\right)
\end{aligned}$$

be the parameters that were considered in [5], Theorems 3 and 4. The proofs of both theorems have four cases, depending on what kinds of paths and vertices are being joined. In all four cases, g_i connects a Π-cap with a Γ-tail, and it is necessary to prove $\Delta_i^{(3)} \leq 0$ and $\Delta_i^{(4)} \leq 0$. In both theorems, we add additional cases for g_i connecting two Π-caps. Our "case 5" describes the only new type of edge we actually use in the construction (in step B8), but the other new cases are necessary to prove the validity of the distance formula (2).

Case 5: Edge g_i closes a $\Pi\Pi\Pi$-path P. We will show $\Delta c_i = \Delta p_{\Gamma\Gamma i} = \Delta r_i = \Delta s_i = \Delta\, gr_i = \Delta\, fr_i = 0$, so that $\Delta_i^{(3)} = 0$ and $\Delta_i^{(4)} = 0$.

Proof: Clearly $\Delta c_i = \Delta p_{\Gamma\Gamma i} = 0$. (Note $\Delta p_{\Pi\Pi\Pi i} = -1$, which is why that parameter is not the correct one to use.)

Closing P does not create or destroy components, does not affect whether any component is interchromosomal or intrachromosomal, and does not affect whether any component is oriented or unoriented. Thus, no components are added to, or removed from, $\mathcal{IU}$.

However, closing P may reclassify a non-real component as real, so we must consider the possibility that a component $K \notin \mathcal{RU}(G_{i-1})$ is moved to $K \in \mathcal{RU}(G_i)$. We will show that this cannot happen, which implies $\mathcal{IU}$ and $\mathcal{RU}$ are unchanged and $\Delta r_i = \Delta s_i = \Delta\, gr_i = \Delta\, fr_i = 0$.

Suppose a component is reclassified on addition of edge g_i: $K \notin \mathcal{RU}(G_{i-1})$ but $K \in \mathcal{RU}(G_i)$. This requires that K be intrachromosomal unoriented, and P is wholly within the extent of K. Then P connects two Π-caps at the ends of the same chromosome, K is the component containing P, and there are no Γ-tails in this chromosome in G_i. Further, there were no Γ-tails in this chromosome in G_0: if there had been, the previously added edges that removed them were interchromosomal.

Say that in $u(\hat{\pi})$, this chromosome is

$$\hat{\pi}_{2i+1}, \hat{\pi}_{2i+2}, \ldots, \hat{\pi}_{2j-1}, \hat{\pi}_{2j} \;=\; x, a_1, \ldots, a_k, y$$

where x, y are Π-caps and the rest are not. Since there are no interchromosomal edges among these, $u(\hat{\gamma})$ has the form $\ldots, v, a_{i_1}, \ldots, a_{i_k}, w, \ldots$, where v, w are Π-caps and the a's are permuted from how they appear in $\hat{\pi}$.

If $k > 0$, the edge (v, a_{i_1}) was deleted to leave a Γ-tail at a_{i_1} in G_0. However, there are no Γ-tails in this chromosome, so $k = 0$ and this chromosome of Π is null. In capped form it is just (x, y), and P and K are just the bare edge (x, y). It is not in $\mathcal{RU}(G_{i-1})$. Closing it turns it into an adjacency, which is still not in $\mathcal{RU}(G_i)$, contradicting the assumption that it is. (Bare edges and adjacencies are specifically excluded from the interleaving graph.)

Case 6: Edge g_i connects two Π-caps in different $\Pi\Pi\Pi$-paths. Then $\Delta c_i = -1$ and $\Delta p_{\Gamma\Gamma i} = \Delta r_i = \Delta s_i = \Delta\, gr_i = \Delta\, fr_i = 0$, so that $\Delta_i^{(3)} = -1 \leq 0$ and $\Delta_i^{(4)} = -1 \leq 0$.

Case 7: Edge g_i connects two Π-caps, one in a $\Pi\Pi\Pi$-path P_1, the other in a $\Pi\Gamma$-path P_2.

The two paths are merged into one $\Pi\Gamma$-path P_3, so $\Delta c_i = -1$ and $\Delta p_{\Gamma\Gamma i} = 0$. No real components were created, destroyed, or affected, so $\Delta r_i = \Delta\, fr_i = 0$.

Each chromosome has two Π-caps but P_1 and P_2 have three, so the resulting component with P_3 is interchromosomal (hence not real). If P_2 was in a semi-knot, that semi-knot is now destroyed, so $\Delta s = -1$ and $\Delta\, gr = 0$ or -1; otherwise $\Delta s = \Delta\, gr = 0$.

`proper_flip_left`(G)

1. `form_components`(G)
2. `classify_components`(G)
3. Determine all distinct chromosomes $i_1, i_2, \ldots, i_k$ that contain the leftmost vertex of one or more interchromosomal unoriented components.
4. Flip chromosomes $i_1, i_2, \ldots, i_k$.

FIGURE 6. Algorithm to find a proper flipping of a graph.

Then $\Delta_i^{(3)} = 0$ or -1 and $\Delta_i^{(4)} = 0$ or -1.

Case 8: Edge g_i connects two Π-caps in different $\Pi\Gamma$-paths, P_1 and P_2.

The two paths are merged into one $\Gamma\Gamma$-path, so $\Delta c_i = -1$ and $\Delta p_{\Gamma\Gamma\, i} = +1$. No real components were created, destroyed, or affected, so $\Delta r_i = \Delta\, fr_i = 0$.

If P_1 or P_2 were in semi-knots, edge g_i destroyed them, giving $\Delta s = -1$ or -2 and $\Delta\, gr = 0$ or -1. If neither path was in a semi-knot, then $\Delta s = \Delta\, gr = 0$.

Then $\Delta_i^{(3)} = -1$ or -2 and $\Delta_i^{(4)} = -1$ or -2.

5. From optimal cappings to optimal concatenates

The procedure `genomic_sort`, steps A1–A19, produced a new capping of Γ to prove the distance formula [5, Thm. 4]. However, to compute the distance without building a proof certificate (i.e., capping), it is only necessary to perform steps A1–A2. It *is* possible to extend that procedure to algorithmically produce an optimal rearrangement scenario between two genomes, but they do not actually give the connection between the capping and the scenario; our added step B19 does this, and we explain it now.

5.1. **Proper flipping.** Hannenhalli and Pevzner's reduction of the multichromosomal rearrangement problem to the unichromosomal rearrangement problem in [5] assumes all interchromosomal components can be made oriented.

A chromosome $\pi(i)$ of Π is *properly flipped* in a graph $G = G(\hat{\pi}, \hat{\gamma})$ if every interchromosomal edge originating from it belongs to an oriented component of G [5, p. 585]. The graph G is properly flipped if all chromosomes are properly flipped. We extend these definitions to apply as well to the graphs $G(\hat{\Pi}, \hat{\Gamma})$ in which cycles among the tails have been removed.

In [5, Lemma 1], they prove the following:

Lemma 3. *If a chromosome $\pi(i)$ is not properly flipped in G, then it is properly flipped in the graph G' obtained by flipping that chromosome. Moreover, every properly flipped chromosome in G remains properly flipped in G'.*

We require an extension of this, whose proof is the same as the original proof.

Lemma 4. *Lemma 3 also applies to graphs $G(\hat{\Pi}, \hat{\Gamma})$ in which cycles among the tails have been removed.*

This leads to a procedure `proper_flip_left` for obtaining a proper flipping (Fig. 6).

Theorem 5. *Algorithm* `proper_flip_left` *results in a properly flipped graph, and takes time $O(n)$.*

Proof: Let $i_1 < i_2 < \ldots < i_k$ be the indices of the chromosomes that contain the leftmost vertex of one or more interchromosomal unoriented components of G.

For convenience, set $i_{k+1} = N_c + 1$. Let G_j be the result of flipping chromosomes $i_1, \ldots, i_j$.

Let $0 \leq j \leq k$. We claim that in G_j, chromosomes $1, 2, \ldots, i_{j+1} - 1$ are properly flipped and (when $j < k$) chromosome i_{j+1} is not. This is true for G_0. Assume it is true for G_{j-1}; then chromosomes $1, 2, \ldots, i_j - 1$ of G_{j-1} are properly flipped but i_j is not. By Lemma 4, flipping chromosome i_j will properly orient chromosome i_j and will keep chromosomes $1, \ldots, i_j - 1$ properly oriented. Now consider chromosome r in G_j, with $i_j < r < i_{j+1}$. None of the chromosome flips $i_1, \ldots, i_j$ affected interchromosomal components with leftmost vertex in chromosome r, because all these flips occurred to the left of the chromosome. So any interchromosomal component incident with chromosome r has its leftmost vertex in a smaller numbered chromosome, and hence is oriented. So chromosome r is properly oriented. Similarly (when $j < k$), chromosome i_{j+1} still has the leftmost vertex of an interchromosomal unoriented component, and so is not properly flipped.

The final result, G_k, is properly oriented.

Steps 1 and 2 take time $O(n)$ each. Step 3 takes time $O(cc)$. Step 4 takes time $O((n_{i_1} + 2) + \cdots + (n_{i_k} + 2))$. The total is $O(n)$.

5.2. **Proper bonding.** In Section 2.2, we noted that in any concatenate, only half the possible fusions and translocations between a given pair of chromosomes can be mimicked by a reversal; flipping one of the chromosomes permits mimicking the other half. Mimicking a sequence of multichromosomal rearrangement operations by reversals potentially requires numerous chromosome flips; recall Equation (1). We will show that the capping produced at step B18 can be used to form concatenates (π^*, γ^*) in which an optimal reversal scenario from π^* to γ^* or vice-versa includes at most one such flip.

The set of *(internal) bonds* of concatenate $\hat{\pi}$ is defined as

$$\left\{(\pi(1)_{n_1+1}, \pi(2)_0), \ldots, (\pi(N_c - 1)_{n_{N_c-1}+1}, \pi(N_c)_0)\right\} .$$

The *external bonds* are $(0, \pi(1)_0)$ and $(\pi(N_c)_{n_{N_c}+1}, n+1)$. For example, in Fig. 1(a), the internal bonds of $\hat{\pi}$ are $\{(10, 11), (12, 13)\}$, and the external bonds are $(0, 9)$ and $(14, 15)$. (Note that we work with signed entries of $\hat{\pi}$, not unsigned entries of $u(\hat{\pi})$.)

A bond (a, b) in $\hat{\gamma}$ is a *proper bond* when either (a, b) or $(-b, -a)$ is a bond in $\hat{\pi}$. We will show it is possible to form concatenates π^*, γ^* with the cappings from step B18, such that these conditions are satisfied:

- P1. $G(\pi^*, \gamma^*)$ is properly oriented.
- P2. Either
  - (a) All internal bonds in γ^* are proper relative to π^*, and π^*, γ^* both start with the same cap and both end with the same cap (i.e., both external bonds are proper); or,
  - (b) there is one improper internal bond, and one improper external bond.

Take a capping Π^*, Γ^* from step B18. Hannenhalli and Pevzner [5, Thm. 2] prove that there is a reversal scenario between suitable concatenates π^* and γ^* that mimics an optimal rearrangement scenario between Π and Γ. In terms of Equation (1), it involves $d(\Pi, \Gamma)$ reversals, translocations, fissions, and fusions; a number of block chromosome flips; and no cap exchanges or nonsensical reversals. Their proof is not fully constructive, however. We will give a fully constructive way to do this using bonds.

They say to form concatenates with $G(\pi^*, \gamma^*)$ properly flipped and $b_{\text{tail}} - c_{\text{tail}}$ minimal, without saying what values are possible or indicating how this might be done. Their parameters are $b_{\text{tail}} = N_c - 1$ black edges among tails, and c_{tail} cycles among tails; these don't account for the leftmost and rightmost tails, so we define $b_{\text{tail}} = N_c + 1$ and adjust c_{tail}. We will give a construction that guarantees $b_{\text{tail}} - c_{\text{tail}}$ is either 0 (giving case P2a above) or 1 (giving case P2b). In the former case, a reversal scenario mimicking an optimal rearrangement scenario will not have any block chromosome flips; in the latter, it will have exactly one flip. We will also prove that this latter case is sometimes unavoidable.

First, we give the most general procedure to produce concatenates whose bonds are all proper bonds, without regard to whether they are properly flipped. Then we adapt it to the additional requirements given above.

The input is a list of the pairs of caps bounding the chromosomes of Π^*, and a similar list for Γ^*. There are N_c pairs in each list.

At each stage, we take two chromosome blocks A, B in Π^* and replace them by the single block $A + B$. A, B, or both, may be flipped from how they were considered at an earlier step. We do a related operation in Γ^*.

When we form a concatenation $A + B$ in Π^*, where the rcap of A is a and the lcap of B is b, we must simultaneously form a concatenation $A' + B'$ in Γ^*, where the rcap of A' is a and the lcap of B' is b. If a and b are in different chromosome blocks of Γ^*, this is possible (and may require flipping chromosome blocks in Γ^*), and we say the concatenation $A + B$ is *legal*. However, if a single block of Γ^* has b and a (or $-a$ and $-b$) as its caps, this is not possible, and we say the concatenation $A + B$ is *illegal*.

Example 6. Suppose Π and Γ have 100 genes and 4 chromosomes each, and the capping at step B18 is this (genes are not shown):

$$\begin{array}{rcl}
\hat{\pi}(1) & = & 101 \cdots 102 \\
\hat{\pi}(2) & = & 103 \cdots 104 \\
\hat{\pi}(3) & = & 105 \cdots 106 \\
\hat{\pi}(4) & = & 107 \cdots 108 \\
\hline
\hat{\gamma}(1) & = & -108 \cdots -107 \\
\hat{\gamma}(2) & = & 103 \cdots -105 \\
\hat{\gamma}(3) & = & 101 \cdots 106 \\
\hat{\gamma}(4) & = & -102 \cdots 104
\end{array}$$

(1) The concatenation $\hat{\pi}(1) - \hat{\pi}(2)$ is illegal because it would form a bond $(102, -104)$, and the block $\hat{\gamma}(4)$ has these (negated) as caps. All five other concatenates $\hat{\pi}(1) \pm \hat{\pi}(j)$ are legal. Let's form $\hat{\pi}(1) + \hat{\pi}(2)$. This creates the bond $(102, 103)$ in $\hat{\pi}$, so we must create the same bond in γ^* by forming $-\hat{\gamma}(4) + \hat{\gamma}(2)$:

$$\begin{array}{rcl}
\hat{\pi}(1) + \hat{\pi}(2) & = & 101 \cdots 102, 103 \cdots 104 \;=\; 101 \cdots 104 \\
\hat{\pi}(3) & = & 105 \cdots 106 \\
\hat{\pi}(4) & = & 107 \cdots 108 \\
\hline
-\hat{\gamma}(4) + \hat{\gamma}(2) & = & -104 \cdots 102, 103 \cdots -105 \;=\; -104 \cdots -105 \\
\hat{\gamma}(1) & = & -108 \cdots -107 \\
\hat{\gamma}(3) & = & 101 \cdots 106
\end{array}$$

(2) The concatenation $(\hat{\pi}(1)+\hat{\pi}(2))+\hat{\pi}(3)$ is illegal because it forms the bond $(104, 105)$, but these (negated) are the caps of a block in Γ^*. The other three concatenates $(\hat{\pi}(1)+\hat{\pi}(2))\pm\hat{\pi}(j)$ are legal; we choose $(\hat{\pi}(1)+\hat{\pi}(2))-\hat{\pi}(3)$. This creates a bond $(104, -106)$, inducing the concatenate $-(-\hat{\gamma}(4)+\hat{\gamma}(2))-\hat{\gamma}(3)$ in Γ^*.

$$\begin{array}{rcl} \hat{\pi}(1)+\hat{\pi}(2)-\hat{\pi}(3) & = & 101\cdots-105 \\ \hat{\pi}(4) & = & 107\cdots 108 \\ \hline\hline -\hat{\gamma}(2)+\hat{\gamma}(4)-\hat{\gamma}(3) & = & 105\cdots-101 \\ \hat{\gamma}(1) & = & -108\cdots-107 \end{array}$$

(3) Both concatenations $(\hat{\pi}(1)+\hat{\pi}(2)-\hat{\pi}(3))\pm\hat{\pi}(4)$ are legal. (Whenever a block in Π^* and a block in Γ^* have the same caps (up to sign), all single step concatenations involving that block will be legal.) If we do $(\hat{\pi}(1)+\hat{\pi}(2)-\hat{\pi}(3))+\hat{\pi}(4)$, we form the bond $(-105, 107)$ which induces the concatenation $(\hat{\gamma}(3)-\hat{\gamma}(4)+\hat{\gamma}(2))-\hat{\gamma}(1)$.

$$\begin{array}{rcl} \hat{\pi}(1)+\hat{\pi}(2)-\hat{\pi}(3)+\hat{\pi}(4) & = & 101\cdots 108 \\ \hline\hline \hat{\gamma}(3)-\hat{\gamma}(4)+\hat{\gamma}(2)-\hat{\gamma}(1) & = & 101\cdots 108 \end{array}$$

Note that it is sometimes necessary to flip this final concatenation to get π^*, γ^* to start and end with the same caps.

Theorem 7. *Step B19: Algorithm* `form_optimal_concatenate` *(Fig. 7) forms concatenates π^*, γ^* of the cappings Π^*, Γ^* so that conditions P1–P2 are satisfied. The time is $O(n \cdot N_c)$, and the average time is $O(n \cdot \ln(N_c))$.*

Proof: Condition P2: At the start of iteration i, we have the concatenate $\hat{\pi}(i)+\cdots+\hat{\pi}(N_c)$. Steps 6 and 8 do not alter any interchromosomal components whose leftmost vertex is in chromosomes $i, i+1, \ldots, N_c$; thus, step 9 does not flip any of these chromosomes, so this concatenate and the bonds in it are unaltered.

When $i > 2$, all bonds formed are legal: there is at most one illegal bond that can be prepended to $\hat{\pi}(i)$, and when it would be formed, step 6 moves a different chromosome before $\hat{\pi}(i)$. Both its caps can form a legal bond with $\hat{\pi}(i)$, so after flipping it if necessary in step 9, the bond formed in step 10 is legal.

When $i = 2$, we try to form a legal bond, but we will fail if doing so results in an improper orientation.

Condition P1: Steps 2 and 9 guarantee that $G(\Pi^*, \Gamma^*)$ is properly flipped. However, $G(\pi^*, \gamma^*)$ also includes cycles among the tails. If all the bonds are proper, the tail cycles are all adjacencies, so they do not introduce new unoriented interchromosomal components. (The internal bonds give adjacencies for the tails between chromosomes; since π^* and γ^* start with the same gene, the leading tails form an adjacency, and since they end with the same gene, the trailing tail is an adjacency.)

Otherwise, there is one improper bond, and $\pi^* = \langle a, \cdots, b, c, \cdots d\rangle$, where (b, c) is the improper bond between the first two chromosomes. Γ has two fragments. The two ways of concatenating them so they start with a are $\gamma^* = \langle a, \cdots, d, -b\cdots-c\rangle$ and $\gamma^* = \langle a, \cdots, d, c\cdots b\rangle$. All the tail cycles are adjacencies, except for one cycle C involving the tails between the first two chromosomes, and the tails following the last chromosome; see Fig. 8. There must be interchromosomal gray edges g originating in chromosome 1; otherwise, at this stage, the first chromosomes of π^* and γ^* would be the same genes in permuted order, and with the same caps, so

`form_optimal_concatenate`$(G, \hat{\pi}, \hat{\gamma})$

1. Initialize the list of block caps of Γ to be the pairs of caps on the chromosomes of Γ.
2. $G =$ `proper_flip_left`(G)
3. **for** $(i = N_c;\ i \geq 2;\ i = i - 1)$ {
4. **if** the bond from $\hat{\pi}(i-1)$ to $\hat{\pi}(i) + \cdots + \hat{\pi}(N_c)$ is illegal {
5. **if** $(i > 2)$
6. $\langle \hat{\pi}(i-2), \hat{\pi}(i-1) \rangle = \langle -\hat{\pi}(i-1), -\hat{\pi}(i-2) \rangle$ (and make corresponding changes to G)
7. **else**
8. $\hat{\pi}(i-1) = -\hat{\pi}(i-1)$ (and make corresponding changes to G)
9. $G =$ `proper_flip_left`(G)
}
10. Form the bond $\hat{\pi}(i-1) + (\hat{\pi}(i) + \cdots + \hat{\pi}(N_c))$. Update the list of bonds and block caps in Γ^* (if step 8 occurred this iteration, and this is not possible, skip it).
}
11. $\pi^* = \hat{\pi}(1) + \cdots + \hat{\pi}(N_c)$
12. **if** there are no improper bonds
13. form the concatenate γ^* starting with the same cap as π^* and with the same internal bonds
14. **else** // 1 improper bond. Other bonds concatenate Γ^* into 2 fragments.
15. Concatenate the two blocks of Γ^* together so that γ^* and π^* start with the same cap.

FIGURE 7. Algorithm to form optimal concatenates of genomes.

a proper bonding would be possible. All such g belong to oriented components of $G(\Pi^*, \Gamma^*)$ (since it is properly flipped), and C is merged with these into an interchromosomal oriented component in $G(\pi^*, \gamma^*)$.

Running time: Steps 1, 2, 11–15 take time $O(n)$. The worst case for the main loop is the low-probability event that we do steps 5–9 on all $N_c - 1$ iterations, giving a time bound $O((N_c - 1) \cdot n)$. However, at most one cap out of the $2(i-1)$ caps on $\hat{\pi}(1), \ldots, \hat{\pi}(i-1)$ is illegal to prepend to $\hat{\pi}(i)$, so there is only a $1/(2(i-1))$ chance of having to do steps 5–9, giving average time $O((\frac{1}{2} + \frac{1}{4} + \cdots + \frac{1}{2(N_c-1)})n) = O(n \cdot \ln(N_c))$.

We are hopeful that the time may be improved to $O(n)$ by making versions of the BMY algorithms and of `proper_flip_left` that work just one chromosome at a time instead of examining all chromosomes: after step 6, one of the four ways to separately flip or not flip chromosomes $i-2$ and $i-1$, makes it properly flipped.

6. Optimal scenarios

6.1. **Mimicking a rearrangement scenario by a reversal scenario.** There are several algorithms for producing optimal reversal scenarios between a pair of

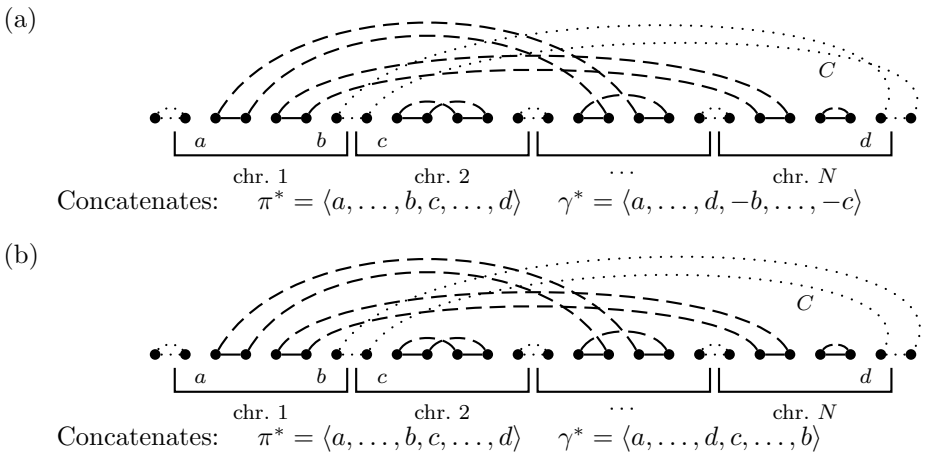


FIGURE 8. The two ways to add tail cycle C for an improper bond between chromosomes 1 and 2. The other bonds are proper, and their tail cycles are adjacencies. All tail cycles are shown dotted.

permutations. This includes the original $O(n^5)$ and $O(n^4)$ algorithms of Hannenhalli and Pevzner [6], the $O(n^2\alpha(n))$ algorithm of Berman and Hannenhalli, and an $O(n^2)$ algorithm of Kaplan, Shamir, and Tarjan [7]. These are easily adapted to produce a multichromosomal rearrangement scenario by interpreting the reversals as described in Section 2.2. Some of these algorithms use the breakpoint graph $G(\hat{\pi}, \hat{\gamma})$; rather than create it from scratch, they can use the graph from the end of step B19.

In adapting a reversal scenario algorithm to produce a multichromosomal rearrangement scenario, there is a restriction that must be obeyed: a reversal starts at an lcap if, and only if, it ends at an rcap. The $O(n^4)$, $O(n^2\alpha(n))$, and $O(n^2)$ algorithms named above do this without any additional modifications, because they select reversals based on the orientations of edges and components. Cycles among tails are all interchromosomal, hence are adjacencies (which they will not reverse) or oriented (in which case they choose edges connecting two tails or two nontails).

This concludes the proof of Theorem 1, except for the existence of genomes in which all optimal scenarios require one block chromosome flip, which will be done in Section 7.1.

6.2. **Breakpoint heuristics for optimal scenarios and trees.** Although the algorithms just named can quickly select good reversals for pairwise genomic rearrangement scenarios, selection of good reversals is NP-hard for even the simplest phylogenetic trees [4]. We have integrated the algorithms in this paper into Guillaume Bourque's program MGR for constructing phylogenetic trees. Here we prove the validity of a heuristic for selecting good rearrangements in these types of problems (Bourque and Pevzner [3, p. 29]). This generalizes a result of Kececioglu and Sankoff [8, Thm. 3] for sorting a signed permutation by reversals.

Let $G = \{\Pi_1, \ldots, \Pi_m\}$ be a set of genomes, either multichromosomal, or unichromosomal with circular, directed linear, or undirected linear chromosomes. A *phylogenetic tree* T on G is a tree whose vertices are genomes on a common set of

genes, and whose leaves are the genomes in G. The *score* of T is the sum of the appropriate distance $d(\Pi, \Gamma)$, taken over all edges (Π, Γ) of T. The *optimal score* of a phylogenetic tree on G is the minimum score among all phylogenetic trees.

A *conserved adjacency* (x, y) of G is a pair of genes such that every genome in G contains either (x, y) or $(-y, -x)$ consecutively. In multichromosomal genomes, these must be consecutive within the same chromosome; no caps or concatenates are being considered. Let $A(\Pi_1, \ldots, \Pi_m)$ denote the set of all conserved adjacencies. A *conserved strip* $(x_1, \ldots, x_k)$ is a sequence of genes such that every genome contains either it or $(-x_k, \ldots, -x_1)$ consecutively. It is comprised of $k-1$ conserved adjacencies.

Theorem 8. *(a) Between any two genomes (Π, Γ), there is an optimal reversal or rearrangement scenario in which the pairs in $A(\Pi, \Gamma)$ are adjacent at every step.*

(b) For a set of genomes $G = \{\Pi_1, \ldots, \Pi_m\}$, there is an optimal phylogenetic tree in which the pairs in $A(\Pi_1, \ldots, \Pi_m)$ are adjacencies in every node, and an optimal rearrangement scenario of form (a) exists on each edge.

Proof: Part (a) is a special case of part (b), so we prove the latter.

Let $A = A(\Pi_1, \ldots, \Pi_m)$. Let $(x, y) \in A$. Let Π_i' be Π_i with $\pm y$ deleted, and $G' = \{\Pi_1', \ldots, \Pi_m'\}$. Any phylogenetic tree T on G can be turned into a tree $\alpha(T)$ on G' by discarding $\pm y$ from every genome. We have $\text{score}(\alpha(T)) \leq \text{score}(T)$ because reversals only on $\pm y$ no longer count, but all other rearrangements do.

Conversely, take any tree T' on G'. In every genome in T', replace x by x, y and $-x$ by $-y, -x$, to form a tree $\beta(T')$. Form a rearrangement scenario on an edge of $\beta(T')$ by taking a scenario on the corresponding edge of T'; keep the same starting and ending genes for each, except rearrangements ending at x should be extended to end at y, and those starting at $-x$ should be extended to start at $-y$. Thus, $\text{score}(\beta(T')) \leq \text{score}(T')$ (because this does not preclude the possibility of alternate scenarios with smaller scores).

Combining these, all genomes in $\beta(\alpha(T))$ preserve the adjacency (x, y), and $\text{score}(\beta(\alpha(T))) \leq \text{score}(T)$.

Let $\alpha_{(x,y)}$, $\beta_{(x,y)}$ denote the above constructions for a specific (x, y). Let $S = (x_1, \ldots, x_k)$ be a conserved strip of G, and

$$\phi_S(T) = \beta_{(x_{k-1},x_k)}\alpha_{(x_{k-1},x_k)} \cdots \beta_{(x_1,x_2)}\alpha_{(x_1,x_2)}(T).$$

All genomes in $\phi_S(T)$ preserve the strip S.

Let $\pm S_1, \ldots, \pm S_m$ be all maximal conserved strips of G, and $\phi(T) = \phi_{S_1} \cdots \phi_{S_m}(T)$. Each leaf Π_i of T is unchanged in $\phi(T)$, and $\text{score}(\phi(T)) \leq \text{score}(T)$. If T is an optimal tree, then $\text{score}(T) \leq \text{score}(\phi(T))$, so these are equal. Then $\phi(T)$ is an optimal tree of form (b).

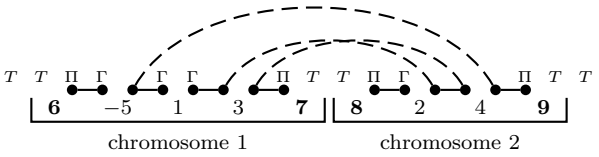
7. Examples

7.1. **Scenario with mandatory flip.** We will prove that all optimal reversal scenarios mimicking a rearrangement scenario from $\Pi = \{\langle -5, 1, 3\rangle, \langle 2, 4\rangle\}$ to $\Gamma = \{\langle 1\rangle, \langle 2, 3, 4, 5\rangle\}$ include a step mimicking flipping a chromosome. We have $d(\Pi, \Gamma) = 3$; see Figs. 9–10.

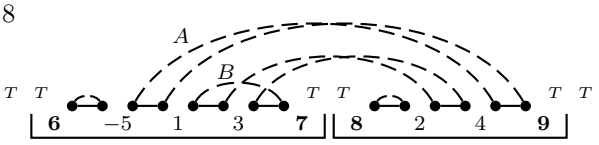
Since our construction allows for the possibility of null chromosomes, we assume both genomes have been padded with null chromosomes to a total of $N_c \geq 2$ chromosomes. Once Π has been capped, there are $(2N_c)!$ ways to cap Γ. There are

(a) Genomes $\Pi = \{\langle -5, 1, 3\rangle, \langle 2, 4\rangle\}$ $\Gamma = \{\langle 1\rangle, \langle 2, 3, 4, 5\rangle\}$
Cappings $\hat{\Pi} = \{\langle \mathbf{6}, -5, 1, 3, \mathbf{7}\rangle, \langle \mathbf{8}, 2, 4, \mathbf{9}\rangle\}$ $\hat{\Gamma} = \{\langle \mathbf{6}, 1, \mathbf{7}\rangle, \langle \mathbf{8}, 2, 3, 4, 5, \mathbf{9}\rangle\}$
Concatenates $\hat{\pi} = \langle \mathbf{6}, -5, 1, 3, \mathbf{7}, \mathbf{8}, 2, 4, \mathbf{9}\rangle$ $\hat{\gamma} = \langle \mathbf{6}, 1, \mathbf{7}, \mathbf{8}, 2, 3, 4, 5, \mathbf{9}\rangle$

(b) $G(\Pi, \Gamma)$

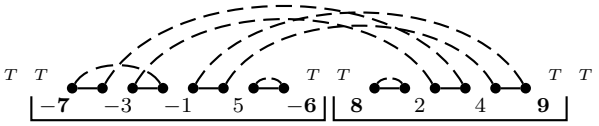


(c) Steps B2–B18

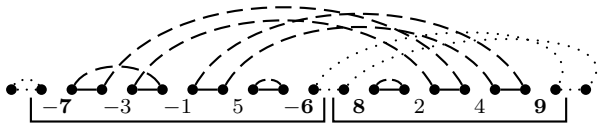


Capping $\Pi^* = \hat{\Pi}$, $\Gamma^* = \{\langle -\mathbf{7}, -1, \mathbf{9}\rangle, \langle \mathbf{8}, 2, 3, 4, 5, -\mathbf{6}\rangle\}$

(d) $G(\Pi^*, \Gamma^*)$

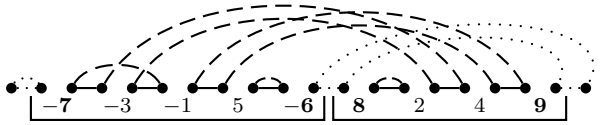


(e) $G(\pi^*, \gamma^*)$



Concatenates: $\pi^* = \langle -\mathbf{7}, -3, -1, 5, -\mathbf{6}, \mathbf{8}, 2, 4, \mathbf{9}\rangle$
$\gamma^* = \langle -\mathbf{7}, -1, \mathbf{9}, \mathbf{6}, -5, -4, -3, -2, -\mathbf{8}\rangle$

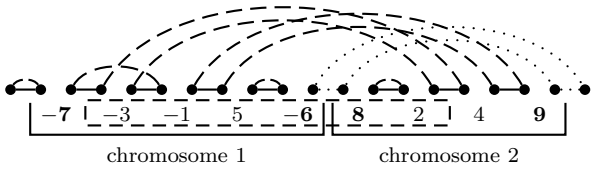
(f) $G(\pi^*, \gamma^{**})$



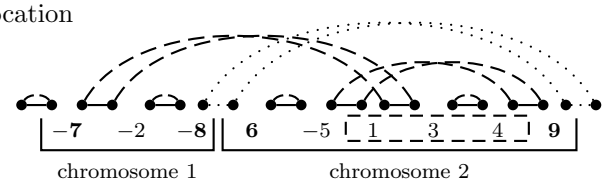
Concatenates: $\pi^* = \langle -\mathbf{7}, -3, -1, 5, -\mathbf{6}, \mathbf{8}, 2, 4, \mathbf{9}\rangle$
$\gamma^{**} = \langle -\mathbf{7}, -1, \mathbf{9}, \mathbf{8}, 2, 3, 4, 5, -\mathbf{6}\rangle$

FIGURE 9. (a) Input. (b) Graph $G(\Pi, \Gamma)$. Parameters $b = 7$, $c = 4$, $p_{\Gamma\Gamma} = s = rr = gr = fr = 0$ give distance $d = 7 - 4 + 0 + 0 + \lceil \frac{0-0+0}{2} \rceil = 3$. (c) Up to Step B18. The graph is properly bonded (with $\gamma^* = \langle \mathbf{6}, -5, -4, -3, -2, -\mathbf{8}, -\mathbf{7}, -1, \mathbf{9}\rangle$) but not properly flipped: interchromosomal component A is oriented but B is not. (d) Step B19: Flip chromosome 1 to properly orient graph. The potential bond $(-6, 8)$ is illegal. (e) Step B19: Form a concatenation γ^* of the two fragments of Γ^*, and add the corresponding tail cycles (shown dotted); note there is a 4 vertex, oriented cycle, *not* just adjacencies. (f) Alternate concatenation γ^{**} and its corresponding tail cycles. This includes a 4 vertex, unoriented cycle, but it intersects an oriented one, so it's part of an interchromosomal oriented component.

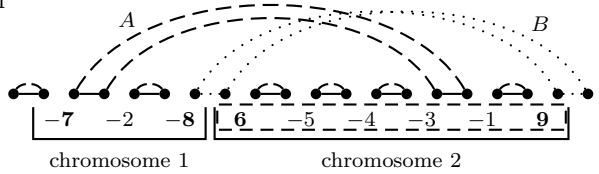
(a) Step 0: π^*



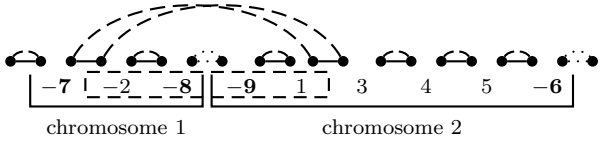
Step 1: Translocation



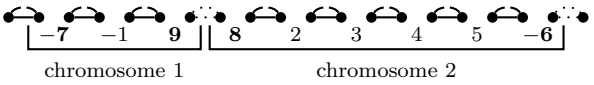
Step 2: Reversal



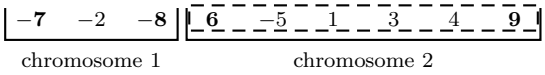
Step 3: Flip chromosome 2



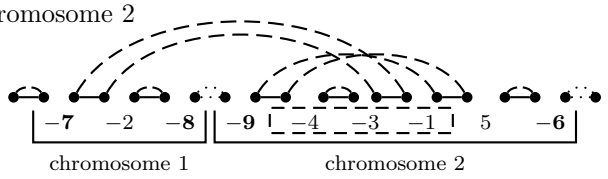
Step 4: Translocation; γ^{**}



(b) Step 1:



Step 2': Flip chromosome 2



Step 3': Reversal

FIGURE 10. Continuation of example in Fig. 9(f). Breakpoint graphs of two optimal scenarios transforming π^* into γ^{**}. At each step, the markers reversed in the transformation to the next step are boxed. (a) In step 2, interchromosomal cycle A is unoriented and interchromosomal tail cycle B is oriented. The component $\{A, B\}$ is only oriented because of the tail cycle B; without considering tail cycles, the graph would have unoriented component $\{A\}$ and would not be properly flipped. (b) Alternate scenario, performing chromosome flip earlier. One graph changes, as does annotation of its neighbors.

$d_{\mathrm{rev}}(\pi,\gamma)$	$\gamma: 12345$	$1\bar5\bar4\bar3\bar2$	$\bar112345$	$\bar1\bar5\bar4\bar3\bar2$	23451	$2345\bar1$	$\bar5\bar4\bar3\bar21$	$\bar5\bar4\bar3\bar2\bar1$
$\pi:\ \bar513124$	5	5	5	5	5	4	4	4
$\bar513\bar4\bar2$	5	4	5	4	5	5	4	4
$\bar3\bar1524$	5	5	4	4	5	4	5	4
$\bar3\bar154\bar2$	5	4	4	4	4	5	5	5
$245\bar113$	5	5	5	4	4	4	4	5
$24\bar3\bar15$	4	5	4	5	4	4	5	5
$\bar4\bar25\bar113$	4	4	5	5	4	5	4	5
$\bar4\bar23\bar15$	4	4	4	5	5	5	5	5

TABLE 2. All possible concatenates require a flip. $-x$ is abbreviated $\bar{x}$.

$N_c! \cdot 2^{N_c}$ concatenates of $\hat\Pi$ and $N_c! \cdot 2^{N_c}$ concatenates of $\hat\Gamma$. Let $\hat\pi, \hat\gamma$ be any such capped concatenates. We will show that $d_{\mathrm{rev}}(\hat\pi, \hat\gamma) \geq 4$.

Delete the caps from $\hat\pi, \hat\gamma$ to form uncapped concatenates π, γ. Clearly $d_{\mathrm{rev}}(\hat\pi, \hat\gamma) \geq d_{\mathrm{rev}}(\pi, \gamma)$ since any optimal reversal scenario transforming $\hat\pi$ to $\hat\gamma$ can be turned into a (possibly nonoptimal) reversal scenario turning π into γ, by deleting the caps. Table 2 shows $d_{\mathrm{rev}}(\pi, \gamma) \geq 4$ in all cases.

7.2. **Hannenhalli and Pevzner's example, revisited.** The example [5, p. 588] gives two genomes Π, Γ with $d(\Pi, \Gamma) = 7$. It produces cappings $\hat\Pi$, $\hat\Gamma$. It gives a concatenation $\hat\pi$ on input with no indication that it should be changed on output, and it outputs a capping $\hat\Gamma$ without indicating what concatenation to use. We focus on their steps (e) and (f). Forming the implied concatenation, we have

$$\begin{aligned} \hat\pi &= \langle \mathbf{13}, -3, -2, \mathbf{14}, \mathbf{15}, -1, 4, 5, 6, 7, 12, \mathbf{16}, \mathbf{17}, 10, 9, 11, 8, \mathbf{18} \rangle \\ \hat\gamma &= \langle \mathbf{13}, 1, 2, 3, 4, -\mathbf{15}, -\mathbf{14}, 5, 6, 7, 8, \mathbf{18}, \mathbf{17}, 9, 10, 11, 12, \mathbf{16} \rangle \end{aligned}$$

(where caps are shown in bold). We may compute $d_{\mathrm{rev}}(\hat\pi, \hat\gamma) = 9 \neq 7$. Note that there are improper bonds: $(16, 17) \neq (18, 17)$. If we flip the third chromosome of both concatenates, the reversal distance becomes 7, and the bonds become proper. (They do indicate that some chromosomes of Π may have to be flipped, but never say when to do that, and they do not discuss what operations must be done to Γ.)

The second error in the example is that the scenario (f) is not optimal for the capping (e). To see this, we place the caps of (e) into the steps of the scenario (f). There is only one way to do this. The chromosome order and orientation shown is arbitrary; this is not a concatenation.

0. $\hat{\Pi}$	$\{\langle \mathbf{13}, -3, -2, \mathbf{14}\rangle, \langle \mathbf{15}, -1, 4, 5, 6, 7, 12, \mathbf{16}\rangle,$ $\langle \mathbf{17}, 10, 9, 11, 8, \mathbf{18}\rangle\}$
1. Translocation	$\{\langle \mathbf{13}, -3, -2, \mathbf{14}\rangle, \langle \mathbf{15}, -1, 4, 5, 6, 7, 8, \mathbf{18}\rangle,$ $\langle \mathbf{17}, 10, 9, 11, 12, \mathbf{16}\rangle\}$
2. Fusion	$\{\langle \mathbf{13}, -3, -2, -1, 4, 5, 6, 7, 8, \mathbf{18}\rangle, \langle \mathbf{15}, \mathbf{14}\rangle,$ $\langle \mathbf{17}, 10, 9, 11, 12, \mathbf{16}\rangle\}$
3. Fission	$\{\langle \mathbf{13}, -3, -2, -1, 4, -\mathbf{15}\rangle, \langle -\mathbf{14}, 5, 6, 7, 8, \mathbf{18}\rangle,$ $\langle \mathbf{17}, 10, 9, 11, 12, \mathbf{16}\rangle\}$
4. Reversal	$\{\langle \mathbf{13}, 1, 2, 3, 4, -\mathbf{15}\rangle, \langle -\mathbf{14}, 5, 6, 7, 8, \mathbf{18}\rangle,$ $\langle \mathbf{17}, 10, 9, 11, 12, \mathbf{16}\rangle\}$
5. Reversal	$\{\langle \mathbf{13}, 1, 2, 3, 4, -\mathbf{15}\rangle, \langle -\mathbf{14}, 5, 6, 7, 8, \mathbf{18}\rangle,$ $\langle \mathbf{17}, -9, -10, 11, 12, \mathbf{16}\rangle\}$
6. Reversal	$\{\langle \mathbf{13}, 1, 2, 3, 4, -\mathbf{15}\rangle, \langle -\mathbf{14}, 5, 6, 7, 8, \mathbf{18}\rangle,$ $\langle \mathbf{17}, -9, 10, 11, 12, \mathbf{16}\rangle\}$
7. Reversal $\to \hat{\Gamma}$	$\{\langle \mathbf{13}, 1, 2, 3, 4, -\mathbf{15}\rangle, \langle -\mathbf{14}, 5, 6, 7, 8, \mathbf{18}\rangle,$ $\langle \mathbf{17}, 9, 10, 11, 12, \mathbf{16}\rangle\}$

Assume there are concatenates $\hat{\pi}$, $\hat{\gamma}$ that are consistent with this sequence of steps, with no additional chromosome flips or other operations. We will show that this cannot occur. Let $\hat{\pi}^{(0)}, \ldots, \hat{\pi}^{(7)}$ denote the concatenates at each step.

The chromosomes of step 1 are $A = \langle \mathbf{13}, -3, -2, \mathbf{14}\rangle$, $B = \langle \mathbf{15}, -1, 4, 5, 6, 7, 8, \mathbf{18}\rangle$, $C = \langle \mathbf{17}, 10, 9, 11, 12, \mathbf{16}\rangle$. In order to mimic the fusion in step 2, $\hat{\pi}^{(1)}$ must have one of the forms $-A + B \pm C$, $-A \pm C + B$, $\pm C - A + B$, or the reverse of one of those. This is 12 possible concatenations in all. With $\hat{\pi}^{(1)} = -A + B + C$, we have

$$\begin{aligned}
\hat{\pi}^{(1)} &= \langle -\mathbf{14}, 2, 3, -\mathbf{13}, \mathbf{15}, -1, 4, 5, 6, 7, 8, \mathbf{18}, \mathbf{17}, 10, 9, 11, 12, \mathbf{16}\rangle \\
\hat{\pi}^{(2)} &= \langle -\mathbf{14}, -\mathbf{15}, \mathbf{13}, -3, -2, -1, 4, 5, 6, 7, 8, \mathbf{18}, \mathbf{17}, 10, 9, 11, 12, \mathbf{16}\rangle \\
\hat{\pi}^{(3)} &= \langle -\mathbf{14}, -4, 1, 2, 3, -\mathbf{13}, \mathbf{15}, 5, 6, 7, 8, \mathbf{18}, \mathbf{17}, 10, 9, 11, 12, \mathbf{16}\rangle
\end{aligned}$$

but the caps on the first two chromosomes of $\hat{\pi}^{(3)}$ are incorrect (interchange $\pm 14 \leftrightarrow \mp 15$). Propagating this through to step 7 gives a capping

$$\hat{\Gamma}' = \{\langle \mathbf{13}, 1, 2, 3, 4, \mathbf{14}\rangle, \langle \mathbf{15}, 5, 6, 7, 8, \mathbf{18}\rangle, \langle \mathbf{17}, 9, 10, 11, 12, \mathbf{16}\rangle\} \ .$$

Note that in step (e) there were two ways to join the ΠΠ- and ΓΓ-paths; the other choice would have given exactly the capping $\hat{\Gamma}'$.

The analysis of the other 11 possible concatenations $\hat{\pi}^{(1)}$ is similar, and they are all consistent only with the capping $\hat{\Gamma}'$.

8. Acknowledgements

I would like to thank Guillaume Bourque, Pavel Pevzner, and Yang Yu for valuable discussions and assistance.

References

1. D. Bader, B.M.E. Moret, T. Warnow, S. Wyman, and M. Yan, *GRAPPA*, 2000, http://www.cs.unm.edu/∼moret/GRAPPA.
2. D.A. Bader, B.M.E. Moret, and M. Yan, *A linear-time algorithm for computing inversion distances between signed permutations with an experimental study*, J. Comput. Biol **8** (2001), no. 5, 483–491.
3. G. Bourque and P.A. Pevzner, *Genome-Scale Evolution: Reconstructing Gene Orders in the Ancestral Species*, Genome Research **12** (2002), no. 1, 26–36.

4. A. Caprara, *Formulations and complexity of multiple sorting by reversals*, Proceedings of the Third Annual International Conference on Computational Molecular Biology (RECOMB '99) (Lyon, France), ACM Press, 1999, pp. 84–93.
5. S. Hannenhalli and P.A. Pevzner, *Transforming men into mice (polynomial algorithm for genomic distance problem)*, 36th Annual Symposium on Foundations of Computer Science (Milwaukee, WI, 1995) (Los Alamitos, CA), IEEE Comput. Soc. Press, Los Alamitos, CA, 1995, pp. 581–592.
6. ———, *Transforming cabbage into turnip: polynomial algorithm for sorting signed permutations by reversals*, J. ACM **46** (1999), no. 1, 1–27.
7. H. Kaplan, R. Shamir, and R.E. Tarjan, *A faster and simpler algorithm for sorting signed permutations by reversals*, SIAM J. Comput. **29** (2000), no. 3, 880–892 (electronic).
8. J. Kececioglu and D. Sankoff, *Efficient bounds for oriented chromosome inversion distance*, Combinatorial pattern matching (Asilomar, CA, 1994) (Berlin), Springer, Berlin, 1994, pp. 307–325.
9. ———, *Exact and approximation algorithms for sorting by reversals, with application to genome rearrangement*, Algorithmica **13** (1995), no. 1-2, 180–210.
10. P.A. Pevzner, *Computational molecular biology: An algorithmic approach*, The MIT Press, Cambridge, MA, 2000, Chapter 10.
11. G. Tesler, *GRIMM*, 2001, http://www-cse.ucsd.edu/groups/bioinformatics/GRIMM.
12. ———, *GRIMM: genome rearrangements web server*, Bioinformatics **18** (2002), no. 3, 492–493.