Reversal and Transposition Distance of Linear Chromosomes

Maria Emilia M. T. Walter
University of Brasília
University of Campinas-Brazil
emilia@dcc.unicamp.br

Zanoni Dias
University of Campinas-Brazil
zanoni@dcc.unicamp.br

João Meidanis
University of Campinas-Brazil
meidanis@dcc.unicamp.br

Abstract

In recent years we are seeing increasing interest in research on mutational events acting on large portions of the chromosomes. Among these events, a reversal acts on a fragment of a chromosome reversing the order and orientation of the genes, and a transposition moves fragments from one region to another within a chromosome. In this article we analyze genomes evolving by reversals and transpositions. We present approximation algorithms to compute the reversal and transposition distance for linear permutations, and a lower bound on the reversal and transposition diameter of signed linear permutations.

1. Introduction

The huge amount of data resulting from genome sequencing in molecular biology is giving rise to an increasing interest in the development of algorithms for comparing genomes of related species. Particularly these data prompted research on mutational events acting on large portions of the chromosomes. Such events can be used to compare genomes for which the traditional alignment methods of comparing DNA sequences are not conclusive. The field originated by the study of non-local mutations on chromosomes is known as *genome rearrangements*.

There are several mutational events affecting large fragments of genomes of organisms, including duplication, insertion, deletion, reversal, transposition (acting on a single chromosome), translocation, fusion and fission (involving more than one chromosome). Each such event or combination of events gives rise to a theoretical problem of finding, given two genomes, the shortest series of events that transforms one genome into the other. We seek the shortest series because it has the largest likelihood of occurrence under a general principle of parsimony. Notice that in general more than one shortest series exists. The length of the shortest series is called the *distance* between the two genomes.

In this article we are working with genomes composed by a single chromosome. Chromosomes are usually represented as *permutations* of integers in the range $1..n$, for a given n, each integer representing a gene or a genetic marker. Sometimes the integers are signed to indicate the orientation of the gene. However, when gene orientations are unknown, the integers are unsigned.

In the last few years we have witnessed formidable advances in our understanding of genome rearrangements. A partial list of known results follows. With respect to the reversal event, Kececioglu and Sankoff [14] presented the first algorithms for computing the reversal distance between two unsigned linear chromosomes. Bafna and Pevzner [1] improved the Kececioglu and Sankoff algorithm, for signed and unsigned linear permutations. Hannenhalli and Pevzner [11] presented the first polynomial time algorithm to find the reversal distance of signed linear chromosomes, later improved on its running time by Berman and Hannenhalli [3] and Kaplan, Shamir and Tarjan [12]. Caprara, Lancia, and Ng [5] implemented a branch-and-bound algorithm for computing the exact reversal distance between two unsigned permutations which performs very well in practice. Caprara [4] later showed that this problem is NP-hard.

Regarding the problem of reversal distance between two signed circular permutations, Kececioglu and Sankoff [13] gave an approximation algorithm, and Meidanis, Walter and Dias [15] gave a polynomial time algorithm for it. With respect to the transposition event, Bafna and Pevzner [2] analyzed the transposition distance problem between two unsigned linear chromosomes, presenting several approximation algorithms. Christie [7] gave a polynomial time algorithm for computing distance under a novel operation, block interchange.

Analyzing genomes evolving due to different mutational events represents today a great challenge. Hannenhalli and co-authors [9] analyzed genomes evolving by different events, particularly reversals and transpositions. Hannenhalli and Pevzner [10] presented a polynomial time algorithm for comparing genomes evolving by reversals, translocations, fusions and fissions. Gu, Peng and Sudborough [8] gave approximation algorithms to compute the distance between two signed permutations, allowing three operations, reversal, transposition and reversal+transposition simultaneously.

In this paper we want to contribute in the analysis of reversals and transpositions acting on a single chromosome. The results of this work are as follows. We extend the analysis of transpositions to signed permutations, and obtain approximation algorithms for computing the reversal and transposition distance for both signed and unsigned permutations. Finally, we present lower bound for the reversal and transposition diameter of signed permutations, and conclude.

2. Definitions

In this section we formalize the problem of computing the reversal and transposition distance of linear chromosomes.

We assume that the order of genes in a chromosome is represented by a permutation $\pi = (\pi_1, \pi_2 \dots \pi_n)$, where each π_i is an integer in $1..n$. If the gene orientations are known, each π_i is a signed integer.

A *reversal* is an operation that transforms a permutation into another, reversing the order of the genes on a certain portion of the permutation. A reversal $r(i,j)$ is defined by two integers i, j, such that $1 \leq i \leq j \leq n$, reversing the order of the genes between i and j, including the extremes. Thus, we have

$$r(i,j) \cdot (\pi_1 \dots \pi_{i-1} \pi_i \pi_{i+1} \dots \pi_j \pi_{j+1} \dots \pi_n) =$$

$$(\pi_1 \dots \pi_{i-1} \overline{\pi}_j \dots \overline{\pi}_{i+1} \overline{\pi}_i \pi_{j+1} \dots \pi_n)$$

where $\overline{\pi}_k$ symbol means $-\pi_k$ if the integer is signed, or π_k if the integer is not signed.

A *transposition* is an operation transforming a permutation into another, "cutting" a certain portion of the permutation and "pasting" it elsewhere in the same permutation. A transposition $t(i,j,k)$ is defined by three integers i, j, and k such that $1 \leq i < j \leq n+1, 1 \leq k \leq n+1$, and $k \notin [i,j]$, in the following way. It "cuts" the portion between positions i and $j-1$, including the extremes, and "pastes" it just before position k. Thus, we can write

$$t(i,j,k) \cdot (\pi_1 \dots \pi_{i-1} \pi_i \dots \pi_{j-1} \pi_j \dots \pi_{k-1} \pi_k \dots \pi_n)$$

$$(\pi_1 \dots \pi_{i-1} \pi_j \dots \pi_{k-1} \pi_i \dots \pi_{j-1} \pi_k \dots \pi_n)$$

if $i < j < k$, and

$$t(i,j,k) \cdot (\pi_1 \dots \pi_{k-1} \pi_k \dots \pi_{i-1} \pi_i \dots \pi_{j-1} \pi_j \dots \pi_n)$$

$$(\pi_1 \dots \pi_{k-1} \pi_i \dots \pi_{j-1} \pi_k \dots \pi_{i-1} \pi_j \dots \pi_n)$$

if $k < i < j$. Notice that $t(i,j,k) = t(j,k,i)$ when $i < j < k$.

Given two permutations π and σ, we want to compute a shortest series of reversals and transpositions that transforms π into σ, that is, we want to find ρ_1, ρ_2, $\dots$, ρ_u, where ρ_i is either a reversal or a transposition, such that $\rho_u \cdot \rho_{u-1} \cdot \dots \cdot \rho_2 \cdot \rho_1 \cdot \pi = \sigma$ and u is minimum. We call u the **reversal and transposition distance** and denote it by $d(\pi, \sigma)$. Without loss of generality we can fix σ. Unless otherwise noted, all our developments will be done with σ being the identity permutation, which is $\sigma = (1 \dots n)$ in the unsigned case and $\sigma = (+1 \dots + n)$ in the signed case.

In the following an **operation** can be a reversal or a transposition.

We usually extend permutation π by adding $\pi_0 = 0$ and $\pi_{n+1} = n+1$ in the unsigned case, or $\pi_0 = +0$ and $\pi_{n+1} = +(n+1)$ in the signed case. The extended permutation will still be denoted by π.

A **breakpoint** of a permutation π is a pair $x = (\pi_i, \pi_{i+1})$ such that neither x nor $\overline{x} = (\overline{\pi}_{i+1}, \overline{\pi}_i)$ are of the form (σ_j, σ_{j+1}) for some j such that $0 \leq j \leq n$. Therefore, to reach σ from π, we must have at least one operation "separating" π_i and π_{i+1}. Breakpoints are indicated by a bullet (•) between π_i and π_{i+1} (see Figure 1). We denote by $b(\pi, \sigma)$ the number of breakpoints of π with respect to σ.

Breakpoints divide a permutation into **strips**. If the target permutation σ is the identity, strips are always sequences of consecutive integers. In the unsigned case, a strip can be either increasing or decreasing as a sequence of integers, and we will call them accordingly as **increasing strips** or **decreasing strips**. In the signed case, all strips are increasing, but we separate them into **positive** or **negative** strips, according to the sign of their elements (all elements in a strip must have the same sign).

$$0 \bullet 5 \bullet 1 \quad 2 \bullet 4 \bullet 7 \quad 6 \bullet 3 \bullet 9 \quad 8 \bullet 10$$

Figure 1. Strips and breakpoints of a permutation $\pi = (\ 0\ 5\ 1\ 2\ 4\ 7\ 6\ 3\ 9\ 8\ 10\ )$ with respect to $\sigma = (\ 0\ 1\ 2\ 3\ 4\ 5\ 6\ 7\ 8\ 9\ 10\ )$. Strips are the sequences between two consecutive breakpoints.

A powerful tool for studying the reversal and transposition distance is the **reality and desire diagram** of two permutations. In the literature [1, 11, 3] this is called the *breakpoint graph* of two permutations, but we prefer to call it a diagram because its graph structure alone does not capture all the important information: the order of nodes is relevant too.

The rest of this section refers to signed permutations only. We construct this diagram writing the original permutation π in the following way. Replace each integer i by a pair of points $-i$ and $+i$, in this order. For instance, $+4$ is replaced by -4 and $+4$; -8 is replaced by $+8$ and -8. Add two extra points, one called $+0$ at the beginning of the sequence, and one called $-(n+1)$ at the end of the sequence. Now draw **reality** edges between $+0$ and $-\pi_1$, between $+\pi_i$ and $-\pi_{i+1}$, and between $+\pi_n$ and $-(n+1)$. Finally, draw **desire** edges between $+0$ and $-\sigma_1$, between $+\sigma_i$ and $-\sigma_{i+1}$, and between $+\sigma_n$ and $-(n+1)$. Again, in the literature, reality edges are called *black edges* and desire edges are called *gray edges*. We prefer the denominations reality and desire because they are more informative: reality edges refer to the current permutation and desire edges refer to the target permutation.

The diagram has exactly $n+1$ reality edges and the same number of desire edges. The idea is that reality edges indicate the situation as it is now, while desire edges indicate the situation sought. When reality equals desire in all edges, we have $\pi = \sigma$ and $d = 0$. Therefore, our goal is to apply reversals and transpositions so that reality becomes desire. Figure 2 shows the diagram corresponding to a pair of permutations. We denote by $G(\pi, \sigma)$ the diagram of the permutations π and σ.

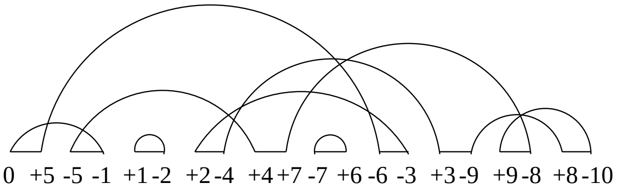

Figure 2. Reality and desire diagram for two permutations, $\pi = (-5\ +1\ +2\ +4\ -7\ -6\ +3\ +9\ +8)$ and $\sigma = (+1\ +2\ +3\ +4\ +5\ +6\ +7\ +8\ +9\ +10)$. The value of $c(\pi, \sigma)$ is 3 in this case.

Observe that the diagram is composed of a number of cycles, with each cycle alternating between reality and desire edges. The **length** of a cycle is the number of reality edges in it (which is the same as the number of desire edges in it). We will denote by **k-cycle** a cycle with length k. The decomposition of $G(\pi, \sigma)$ into cycles is unique and we denote by $c(\pi, \sigma)$ the number of the cycles in $G(\pi, \sigma)$.

3. Approximation algorithms

We present now approximation algorithms for computing the reversal and transposition distance of two permutations. We will give a 3-approximation algorithm for the unsigned case and a 2-approximation algorithm for the signed case.

Let us begin with the unsigned case. Note that the only permutation having 0 breakpoints with respect to σ is exactly σ, and then the sequence of reversals and transpositions transforming π into σ must take the number of breakpoints from $b(\pi, \sigma)$ to 0. We also observe that reversals can remove at most two breakpoints, and transpositions can remove at most three breakpoints. This observation implies immediately a lower bound, given in the next theorem.

Theorem 1 *Given two unsigned permutations π and σ we have*

$$\frac{b(\pi, \sigma)}{3} \leq d(\pi, \sigma).$$

Theorem 2 *Given two permutations π and σ, with $\pi \neq \sigma$, there is an operation ρ removing at least one breakpoint.*

Proof: The intuitive idea is to increase the first strip on each operation, removing its rightmost breakpoint without introducing new breakpoints.

The first strip on the left is always an increasing strip. Taking the maximum element on this first strip, find its successor, which will be necessarily to the right. If the successor is in the beginning of a strip, or is the only element on the strip, we apply a transposition. If it is in the end, we apply a reversal. $\square$

Repeated application of Theorem 2 gives a 3-approximation algorithm for computing the reversal and transposition distance of unsigned permutations. Its time complexity is $O(n^2)$, where n is the size of the permutations. It takes time $O(n)$ to find the operation and apply it.

3.1. Signed Permutations

Now we turn to the signed case. Note that the diagram $G(\sigma, \sigma)$ is the only one having $n+1$ cycles. So, the sequence of reversals and transpositions transforming π into σ must take the number of cycles from $c(\pi, \sigma)$ to $n+1$. For two permutations π and σ, and an operation ρ, denote $\Delta c(\rho) = c(\rho \cdot \pi, \sigma) - c(\pi, \sigma)$ as the gain in the number of cycles due to an operation ρ.

Lemma 1 $\Delta c(\rho) \in \{-2, -1, 0, 1, 2\}$

Proof: We note first that ρ can be a reversal or a transposition.

Each reversal acts on two reality edges belonging to at most two cycles, creating or destroying at most one cycle. Hannenhalli and Pevzner [11] have shown that, for a reversal, $\Delta c(\rho) \in \{-1, 0, 1\}$.

Each transposition acts on three reality edges belonging to at most three cycles. Bafna and Pevzner [2] have shown that for the unsigned case $\Delta c(\rho) \in \{-2, 0, 2\}$. It corresponds, in the signed case, to a diagram generated by a permutation composed only by positive strips. However, in the signed case, we have also $\Delta c(\rho) = -1$ or $+1$. This can be seen from Figure 3, which shows all possible actions of a transposition on signed permutations. □

The following theorem comes directly from Lemma 1.

Theorem 3 *Given two signed permutations π and σ then we have*

$$\frac{(n+1) - c(\pi, \sigma)}{2} \leq d(\pi, \sigma)$$

For $x \in \{2, 1, 0, -1, -2\}$, define a x-**move** on π with respect to σ as an operation ρ such that $\Delta c(\rho) = x$. As we mentioned, Figure 3 shows all possible actions on signed permutations. In each of the cases shown, a transposition transforms reality edges $(b, a), (d, c)$ and (f, e) into $(d, a), (b, e)$ and (f, c). Dashed lines denote a path that can be formed by one or more desire/reality edges. Since the inverse of a transposition is a transposition, the transformations are reversible. Notice that there is only one pattern corresponding to a 2-move, and only three patterns corresponding to an 1-move.Notice that only one case corresponds to a 2-move. This fact leads to the following theorem.

Theorem 4 *A diagram admits a 2-move if and only if there are three reality edges (a, b), (c, d), and (e, f) such that*

1. *they appear in this order in the diagram*
2. *they belong to the same cycle*
3. *a is connected to d, b to e, and c to f, by paths not containing any of the edges (a, b), (c, d), and (e, f).*

We show now a way to apply a reversal or a transposition on a signed permutation in order to obtain an increase of $c(\pi, \sigma)$ by at least 2 in two consecutive moves.

Theorem 5 *Given two signed permutations π and σ, there is either a 1-move, a 2-move or a 0-move followed by a 2-move.*

Proof: If there are negative strips, Hannenhalli and Pevzner [11] have shown that there is always a reversal increasing the number of cycles, and is therefore an 1-move.

If all strips are positive, we can view this permutation as an unsigned one, and apply a result from Bafna and Pevzner

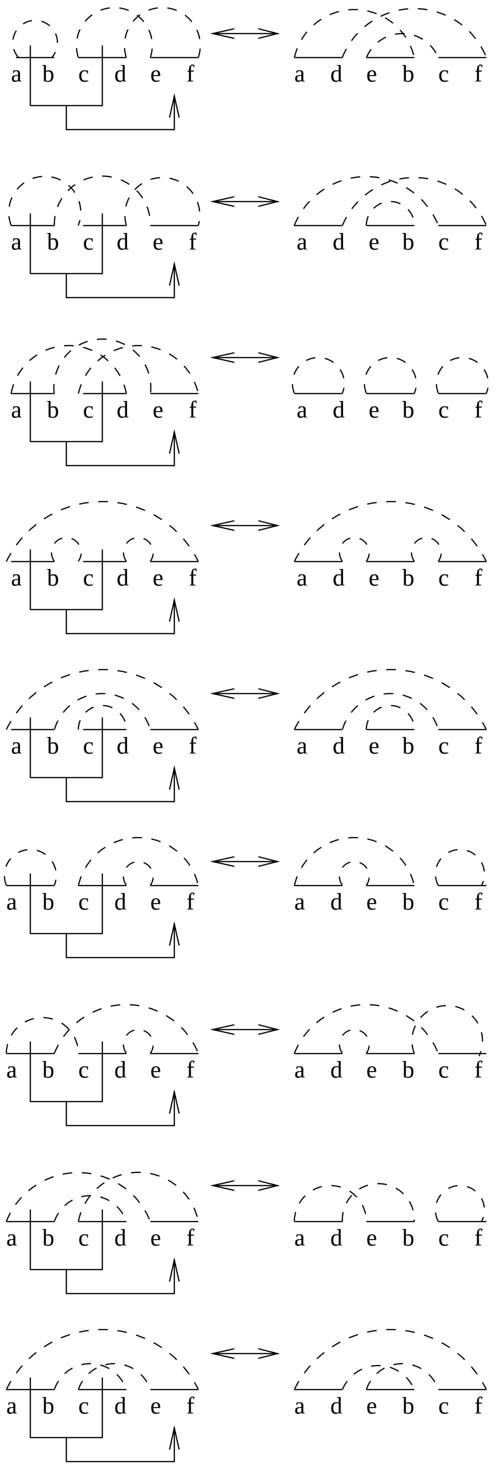

Figure 3. This figure shows all possible cases of transposition acting on a signed permutation, where only the affected cycles are shown.

[2], guaranteeing the existence of either a 2-move, or a 0-move followed by a 2-move. ☐

From Theorem 5 we can derive an upper bound for the reversal and transposition distance.

Theorem 6 *Given two signed permutations π and σ we have*

$$d(\pi,\sigma) \leq (n+1) - c(\pi,\sigma)$$

Given a permutation π to be transformed into σ, the intuitive idea of the algorithm is, while we have negative strips on π with respect to σ we apply reversals as described on Theorem 5. If we cannot apply reversals of this kind, and this sequence of reversals did not transform π into σ then the diagram is generated by a permutation having only positive strips with respect to σ. Then we use the results of Bafna and Pevzner [2] to discover the sequence of transpositions to be applied. We note that, when these transpositions are being applied, all diagrams are generated from permutations having only positive strips with respect to σ.

This gives a 2-approximation algorithm for computing the reversal and transposition distance of signed permutations. Its time complexity is $O(n^2)$, where n is the size of the permutations. Both a suitable reversal and a suitable transposition, as specified in the proof of Theorem 5, can be found in time $O(n)$ [2, 11].

4. Reversal and transposition diameter

In this section we give initial steps for computing the maximum number of operations for the reversal and transposition distance of signed permutations.

Taking S_n as the set of all permutations with size n, define

$$D(n) = \max_{\pi,\sigma \in S_n} d(\pi,\sigma)$$

to be the **reversal and transposition diameter** of this set. Let $\pi = (-1\ -2\ \ldots\ -(n-1)\ -n)$ and $\sigma = (+1\ +2\ \ldots\ +(n-1)\ +n)$. Then $c(\pi,\sigma) = 1$ for all n, and we have a lower bound (by Theorem 3),

$$\left\lfloor \frac{n}{2} \right\rfloor \leq d(\pi,\sigma).$$

More precisely, we can prove the following result.

Theorem 7 *Taking $\pi = (-1\ -2\ \ldots\ -(n-1)\ -n)$ and $\sigma = (+1\ +2\ \ldots\ n-1\ n)$, then we have*

$$d(\pi,\sigma) = \begin{cases} \lfloor \frac{n}{2} \rfloor + 1 & \textit{if}\, n = 1,2 \\ \lfloor \frac{n}{2} \rfloor + 2 & \textit{if}\, n \geq 3 \end{cases}$$

Proof: We have two cases according to the parity of n.

- When n is odd, a lower bound is $\lceil \frac{n}{2} \rceil \leq d(\pi,\sigma)$. But $\lceil \frac{n}{2} \rceil = \frac{n+1}{2} = \lfloor \frac{n}{2} \rfloor + 1 \leq d(\pi,\sigma)$. To achieve the lower bound, all operations applied must be 2-moves, except one which must be a 1-move. No 2-moves exist in the original diagram, and for every 1-move in the first step, the resulting diagram does not admit 2-moves. Hence, the lower bound cannot be achieved and we have $d(\pi,\sigma) \geq \lfloor \frac{n}{2} \rfloor + 2$.
- When n is even, Tables 1 summarize the argument. The table labeled "FIRST MOVE" analyzes all possibilities for the first move. Two of these possibilities require an analysis of the second move as well, which is done in the table labeled "SECOND MOVE." Two arguments are used heavily in this table. One of them is that when negative elements remain, we cannot achieve the $\lfloor \frac{n}{2} \rfloor + 1$ lower bound because we are forced to use at least one reversal, which is never a 2-move. The other is that one can verify in some cases that a 2-move does not exist looking for the characterization given by Theorem 4.

When $n \geq 3$ we can obtain an upper bound for $d(\pi,\sigma)$ in the following way. First, we apply a reversal on π, obtaining $r(1,n)\cdot\pi = (+n\ +(n-1)\ \ldots\ +2\ +1)$. Then we use the result of Christie [6], also obtained independently by Meidanis, Walter and Dias [16], that determines the transposition distance $d_t(r\cdot\pi,\sigma) = \lfloor \frac{n}{2} \rfloor + 1$, for $n > 2$. Therefore, we have the upper bound, $d(\pi,\sigma) \leq \lfloor \frac{n}{2} \rfloor + 2$, for $n \geq 3$. This completes the proof. ☐

An observation here is that an algorithm to obtain a shortest sequence of operations to transform $\pi = (-1\ -2\ \ldots\ -(n-1)\ -n)$ into the identity is: first apply a reversal on π to obtain $r(1,n)\cdot\pi = (+n\ +(n-1)\ \ldots\ +2\ +1)$, and then use the algorithm of Meidanis, Walter and Dias [16] to get the transpositions for computing $d(r\cdot\pi,\sigma)$.

So, we have a lower bound on the diameter

$$\left\lfloor \frac{n}{2} \right\rfloor + 2 \leq D(n).$$

We can verify that $D(n) = \lfloor \frac{n}{2} \rfloor + 2$ for $n = 3, 4$.

5. Conclusions

In this article we have presented approximation algorithms for computing the reversal and transposition distance. For the signed and unsigned cases we have shown algorithms based on the notion of breakpoints and cycles, respectively. For the signed case our algorithm uses a specific type of reversal while possible, and after that it uses part of the Bafna and Pevzner theory [2] to get the transpositions to be applied.

The lower bounds used to estimate the approximation factor were simple, yet they lead to a deeper result, namely,

FIRST MOVE		
reversal	2-move	impossible, no reversal is a 2-move
	1-move	must be $r(i,j)$ with $j-i$ even; analyze second move
	0-move	if $r(1,n)$, use known result on transposition distance (a); otherwise negative elements remain
transposition	2-move	the unique 2-move pattern in Figure 3 does not exist in the diagram
	1-move	must be $t(i,j,k)$ with $j-i$ and $k-i$ both odd; analyze second move
	0-move	negative elements remain

SECOND MOVE		
$r(i,j)$ with $j-i$ even	2-move	the unique 2-move pattern in Figure 3 does not exist in the diagram
	1-move	if transposition, negative elements remain; if reversal, negative elements remain except when first move was $r(1,n)$ or $r(i+1,n)$ for odd i, but then the unique 2-move pattern in Figure 3 does not exist in the diagram
$t(i,j,k)$ with both $j-i$ and $k-j$ odd	2-move	the unique 2-move pattern in Figure 3 does not exist in the diagram
	1-move	if transposition, negative elements remain; if reversal, negative elements remain except when it is $r(1,n)$. But then $r(1,n) \cdot t(i,j,k) = t(n+2-k, n+2-j, n+2-i) \cdot r(1,n)$, which was already analyzed

Table 1. Analysis of the first two steps in computing the distance. (a) See result by Christie and Meidanis, Walter, Dias in the text. In the SECOND MOVE table we do not consider 0-moves since if the second move is a 0-move, the lower bound cannot be achieved, because the first move was an 1-move.

the calculation of the exact distance between permutation $(-1\ -2\ \ldots\ -(n-1)\ -n)$ and the identity. This proof is more involved and uses the characterization of 2- and 1-moves given in Figure 3. Of course the result provides a lower bound for the diameter, which we conjecture to be an upper bound as well.

Plans for future work include dealing with other operations, notably the combined reversal+transposition, which is a natural operation to consider form the biological standpoint, and studying weighted problems, where each type of operation has a different weight, and the goal is to minimize the total weight.

References

[1] V. Bafna and P. Pevzner. Genome rearrangements and sorting by reversals. In *Proc. 34th Annual IEEE Symposium on Foundations of Computer Science*, pages 148–157, 1993. Extended version published with same title in SIAM Journal of Computing, 25(2):272–289,1996.

[2] V. Bafna and P. Pevzner. Sorting by transpositions. In *Proc. 6th Annual ACM-SIAM Symposium on Discrete Algorithms - SODA'95*, pages 614–623, 1995.

[3] P. Berman and S. Hannenhalli. Fast sorting by reversals. In *Proc. 7th Annual Symposium on Combinatorial Pattern Matching - CPM'96*, pages 168–185, 1996.

[4] A. Caprara. Sorting by reversals is difficult. In *Proc. 1st Annual International Conference on Computational Molecular Biology - RECOMB'97*, pages 75–83, 1997.

[5] A. Caprara, G. Lancia, and S. K. Ng. A column-generation based branch-and-bound algorithm for sorting by reversals. Technical Report OR/95/7, DEIS, Universitá di Bologna, 1995.

[6] D. A. Christie. Second year report. University of Glasgow, http://www.dcs.gla.ac.uk/∼christie, 1996.

[7] D. A. Christie. Sorting permutations by block interchanges. *Information Processing Letters*, 60:165–169, 1996.

[8] Q.-P. Gu, S. Peng, and H. Sudborough. Approximating algorithms for genome rearrangements. In *Proc. 7th Worshop on Genome Informatics - GIW'96*, 1996.

[9] S. Hannenhalli, C. Chappey, E. V. Koonin, and P. Pevzner. Genome sequence comparison and scenarios for gene rearrangements: A test case. *Genomics*, 30:299–311, 1995.

[10] S. Hannenhalli and P. Pevzner. Tranforming men into mice (polynomial algorithmic for genomic distance problem). In *Proc. 36th Annual IEEE Symposium on Foundations of Computer Science - FOCS'95*, pages 581–592, 1995.

[11] S. Hannenhalli and P. Pevzner. Transforming cabbage into turnip (polynomial algorithm for sorting signed permutations by reversals). In *Proc. 20th Annual ACM Symposium on Theory of Computing - STOC'95*, pages 178–189, 1995.

[12] H. Kaplan, R. Shamir, and R. E. Tarjan. Faster and simpler algorithm for sorting signed permutations by reversals. In *Proc. 8th Annual ACM-SIAM Symposium on Discrete Algorithms - SODA'97*, pages 344–351, 1997.

[13] J. Kececioglu and D. Sankoff. Efficient bounds for oriented chromosome inversion distance. In *Proc. 5th Annual Symposium on Combinatorial Pattern - CPM'94*, pages 307–325, 1994.

[14] J. Kececioglu and D. Sankoff. Exact and approximation algorithms for the inversion distance between two permutations. *Algorithmica*, 13:180–210, 1995.

[15] J. Meidanis, M. Walter, and Z. Dias. Distância de reversão de cromossomos circulares com sinais. In *Proc. XXIV Seminário Integrado de Software e Hardware - SEMISH'97*, Aug. 1997. In Portuguese.

[16] J. Meidanis, M. Walter, and Z. Dias. Transposition distance of strictly decreasing sequences. In *Proc. 4th South American Workshop on String Processing*, pages 70–79, 1997.