On the Properties of Sequences of Reversals that Sort a Signed Permutation

Anne BERGERON[1,2] Cedric CHAUVE[1] Tzvika HARTMAN[3]
Karine ST-ONGE[1]

[1] LACIM, Université du Québec à Montréal, CP 8888 Succ. Centre-Ville, Montréal, Canada H3C 3P8.
[2] Institut Gaspard-Monge, Université de Marne-la-Vallée, F-77454, Marne-la-Vallée, France
[3] Department of Applied Mathematics and Computer Science, Weizmann Institute, Rehovot 76100, Israel.

E-mail {anne, chauve}@lacim.uqam.ca, tzvi@wisdom.weizmann.ac.il

Abstract

The sorting by reversals *problem is classical in the field of whole genome comparison. In this paper, we provide experimental and theoretical evidence showing that, typically, there is a huge number of optimal sequences of reversals that sort a given signed permutation. We study these sets of optimal sequences using secondary sorting constraints, and using theoretical tools developed in the context of trace monoids. We show that most sorting strategies work well with random permutations, and identify combinatorial parameters, such as stack height, that can be used to classify sequences of reversals, and permutations.*

Keywords: *reversal distance, genome rearrangements, sets of optimal sequences.*

1 Introduction

The genome of a species can be thought of as a set of ordered sequences of genes – the ordering devices being the chromosomes –, each gene having an orientation given by its location on the DNA double strand. Comparing two sets of similar genes appearing along a chromosome in two species yields two (signed) permutations. It is widely accepted that the *reversal distance* between these two permutations, defined as the minimal number of reversals that transform one permutation into the other, provides a good estimate of the evolutionary distance between the two species.

The related algorithmic research has focused on two problems. The first one is to find the reversal distance between two permutations [12, 1]. This distance is the basic step in phylogeny reconstruction from gene order data. A more complex task, called *sorting by reversals*, is the reconstruction of a a sequence of reversals which realizes the distance [12, 14, 2]: The goal is to provide the biologist with a sequence of events (reversals) that may give a clue on the evolutionary history between the two species.

In this paper, we give both experimental and theoretical evidence – which enhances the experimental evidence of [17] – that, typically, there is a huge number of optimal sequences for a given permutation. In the light of these results, we argue that in order to tell the real evolutionary history, one should take into consideration secondary biological constraints. These constraints will often translate as combinatorial parameters of the sequences, and we explore some of these in conjunction with the sorting problem.

In order to provide an adequate algorithmic background for the study of sorting strategies using secondary constraints, we also need to understand the structure and properties of sets of optimal sequences. For this purpose, we borrowed tools from the theory of trace monoids [8] to show that sets of optimal sequences have interesting properties. We also present experiments that show the dramatic impact of various optimal sorting strategies on the structure of the corresponding sequence.

The paper is organized as follows. The next section states some basic definitions and results from the Hannenhalli-Pevzner theory. In section 3 we provide both experimental and theoretical evidence that typically, there is a huge number of optimal sequences. Section 4 explores the structure of the set of optimal sequence, and gives a link to the theory of trace monoids.

2 Preliminaries

Let $\pi = (\pi_1, \ldots, \pi_n)$ be a signed permutation on n elements. As usual, we will assume that the permutation is *framed* by 0 and $n + 1$, and that those extra elements are always positive:

$$\pi = (0\ \pi_1 \ldots \pi_n\ n + 1).$$

A *reversal* $r_{i,j}$ (for $1 \leq i \leq j \leq n$) on the permutation π reverses the order and sign of the segment of π which starts at i and ends at j, yielding:

$$r_{i,j} \cdot \pi = (0\ \pi_1 \ldots - \pi_j \ldots - \pi_i \ldots \pi_n\ n + 1).$$

Equivalently, a reversal can be identified by the subset of the (unsigned) reversed elements $S = \{\pi_i, \ldots, \pi_j\}$. The context will make it clear whether a reversal is identified by its boundaries or by its set of reversed elements. A set of integers S is *well defined* with respect to π if the elements of S appear consecutively in π. For example, if

$$\pi = (0\ \ -4\ \ 3\ \ -6\ \ -1\ \ 5\ \ 2\ 7),$$

then the set $\{1, 3, 6\}$ is well defined and corresponds to the reversal $r_{2,4}$.
The *reversal distance* between two permutations π and σ is the minimal number of reversals needed to transform π into σ. Without loss of generality we can assume that σ is the identity permutation, so the reversal distance of π, denoted by $d(\pi)$, is the minimum number of reversals needed to sort π.
A pair of elements $(\pi_i, \pi_{i+1}), 0 \leq i \leq n$, is called an *adjacency* if $\pi_{i+1} - \pi_i = 1$, otherwise, it is called a *breakpoint*. An *oriented pair* (π_i, π_j) is a pair of consecutive integers, that is $|\pi_i| - |\pi_j| = \pm 1$, with opposite signs. Oriented pairs can always be transformed into an adjacency with a suitable reversal, called an *oriented reversal*. For example, in

$$\pi = (0\ \ 3\ \ 1\ \ 5\ \ 6\ \ -2\ \ 4\ \ 7),$$

the pair $(5, 6)$ is an adjacency, all other pairs are breakpoints. The oriented pairs are $(1, -2)$ and $(-2, 3)$. The oriented pair $(1, -2)$ is transformed into an adjacency by the reversal $r_{3,5}$:

$$r_{3,5} \cdot \pi = (0\ \ 3\ \ 1\ \ 2\ \ -6\ \ -5\ \ 4\ \ 7),$$

and the oriented pair $(3, -2)$ is transformed into an adjacency by the reversal $r_{1,4}$:

$$r_{1,4} \cdot \pi = (0\ \ -6\ \ -5\ \ -1\ \ -3\ \ -2\ \ 4\ \ 7).$$

The relation between adjacencies and oriented reversals works in the other direction as well: A reversal that cuts an adjacency creates an oriented pair in the resulting permutation. A sequence of reversals $R = r_1, \ldots, r_d$ applied to a permutation π is called an *optimal sequence*, if its length $d = d(\pi)$, and the sequence sorts π.

3 There is a Huge Number of Optimal Sequences

In order to understand the structure and properties of the (huge) set of optimal sequences, we choose to generate, with different strategies, many examples of optimal sequences. The main conclusion of these experiments is that almost any sorting strategy will produce an optimal sequence most of the time. This opens the way to sorting strategies based on biological constraints more stringent than mere minimality of the number of operations.

3.1 Almost Anything Works

In this section, we will explore various strategies for constructing optimal sequences, with the assumption that there exists at least one sequence of oriented reversals that sorts the permutation, called permutation *without hurdles*. This assumption excludes a subset of permutations that appear with very low frequency, both in random permutations, and in permutations that are associated to biological data. Permutations that have no negative elements are examples of these hard to sort permutations. Note that a permutation without hurdles always have at least one oriented reversal.

Even if a permutation is without hurdles, some bad choices of oriented reversals can create hard to sort permutations. For example, choosing the obvious reversal of the first two elements in the permutation:

$$(\quad 0 \quad \underline{-2 \quad -1} \quad 4 \quad 3 \quad 5 \quad)$$

for which $d = 3$, yields the permutation

$$(\quad 0 \quad 1 \quad 2 \quad 4 \quad 3 \quad 5 \quad)$$

that has no oriented reversals. On the other hand, the original permutation can be sorted by the following sequence of three reversals.

$$(\quad 0 \quad \underline{-2 \quad -1 \quad 4} \quad 3 \quad 5 \quad)$$

$$(\quad 0 \quad -4 \quad \underline{1 \quad 2 \quad 3} \quad 5 \quad)$$

$$(\quad 0 \quad \underline{-4 \quad -3 \quad -2 \quad -1} \quad 5 \quad)$$

$$(\quad 0 \quad 1 \quad 2 \quad 3 \quad 4 \quad 5 \quad)$$

Definition 1 *Let π be a permutation without hurdles, a reversal r is* safe *if r is an oriented reversal of π such that $r \cdot \pi$ is also without hurdles, otherwise it is* unsafe

Experimental evidence [17] shows that, typically, almost all oriented reversals are safe, and many secondary constraints can be used to generate optimal sequences that have particular characteristics. In a first series of experiments, performed on random permutations, we measured the rate of success of various strategies in selecting safe reversals. The strategies are the following:

1. *Maximal Score:* This strategy chooses an oriented reversal that maximizes the number of oriented reversals in the resulting permutation. For permutations without hurdles, this strategy always finds an optimal sequence [2], and provides a way to identify permutations without hurdles.

2. *Maximal Length:* This strategy chooses an oriented reversal that maximizes the number of reversed elements.

3. *Minimal Length:* This strategy chooses an oriented reversal that minimizes the number of reversed elements.

4. *Random:* This strategy chooses a random oriented reversal.

5. *Minimal Score:* This strategy chooses an oriented reversal that minimizes the number of oriented reversals in the resulting permutation. From a theoretical point of view, this strategy should be one of the worst.

Surprisingly, most strategies work pretty well. The following table presents some of our results, obtained with a set of 10 000 random signed permutations on 128 elements. We applied the five strategies on each of the 10 000 permutations, stopping when the resulting permutation contained only positive elements. A sequence of reversals is considered successful if the resulting permutation is the identity.

Strategy	Success Rate (in %)
Maximal Score	100
Maximal Length	79
Minimal Length	64
Random	62
Minimal Score	12

The 100% success rate of the maximal score strategy is a consequence of its proven success in the absence of hurdles, and from the fact that permutation with hurdles are almost non existent in permutations on 128 elements [6]. The other rates are generally much more that we expected, since each experiment involved a choice of about 124 different reversals – the mean value of $d(\pi)$ for the set of 10 000 permutations is 124.33 – and performing a single unsafe reversal in the sequence ruins the experiment. Even the 12% success rate of the minimal score strategy is amazing, considering that this strategy is almost guaranteed not to succeed.

3.2 On the Proportion of Unsafe Reversals

In this section we provide theoretical explanations for the fact that most sorting strategies are quite successful. The basis of the argument is that permutations without hurdles are scarce, and most reversals are safe.

Denote the set of signed permutations over n elements by Σ_n, and the subset of signed permutations with no adjacencies by Σ_n^0. Adjacencies in a permutation simplify the sorting problem, in the sense that there is no point in breaking an adjacency [13]. A permutation π in Σ_n with k adjacencies can always be reduced to a permutation in Σ_{n-k}^0 with no adjacencies by *gluing* the adjacencies of π, and renaming the other elements. If $\Sigma_{\leq n}^0$ denotes the set of permutations on n or less elements with no adjacencies, the gluing operation is a well defined function:

$$Glue : \Sigma_n \longrightarrow \Sigma_{\leq n}^0$$

For example, the permutation $\pi = (0\ \ -1\ \ 4\ \ 5\ \ 2\ \ 3\ \ 6)$ has two adjacencies, and

$$Glue(\pi) = (0\ \ -1\ \ 3\ \ 2\ \ 4).$$

The inverse operation, which expands the j^{th} element of a permutation into $(\pi_j, \pi_j + 1)$, or $(-\pi_j - 1, -\pi_j)$, and renames the other elements, is given by the function:

$$Ext_{n-1} : \Sigma_{n-1} \times \{1, \ldots, n-1\} \longrightarrow \Sigma_n \setminus \Sigma_n^0.$$

For example, if $\pi = (0\ \ -1\ \ 3\ \ 2\ \ 4)$, then

$$\begin{aligned} Ext(\pi, 1) &= (0\ \ -2\ \ -1\ \ 4\ \ 3\ \ 5), \\ Ext(\pi, 2) &= (0\ \ -1\ \ 3\ \ 4\ \ 2\ \ 5). \end{aligned}$$

The following lemma states that the size of the set Σ_n^0 is comparable to the size of Σ_n.

Lemma 2 *For all $n > 1$, $\frac{1}{2}|\Sigma_n| < |\Sigma_n^0| < |\Sigma_n|$.*

Proof: Clearly, $|\Sigma_n| = 2^n n!$. Since the function Ext_{n-1} is surjective, we have

$$|\Sigma_n \setminus \Sigma_n^0| \leq |\Sigma_{n-1} \times \{1, \ldots, n-1\}| = 2^{n-1}(n-1)!(n-1),$$

which implies $|\Sigma_n^0| \geq 2^n n! - 2^{n-1}(n-1)!(n-1) > \frac{1}{2}|\Sigma_n|$. ■

Reasoning on random oriented reversals of random permutations requires to consider the set of pairs (π, r) in which r is an oriented reversal of π. This set splits into two sets, depending whether r creates one or two adjacencies in π. Denote these two sets respectively by A_1 and A_2. Consider the functions:

$$\begin{aligned} f_1 : A_1 &\longrightarrow \Sigma_{n-1}^0 \\ f_2 : A_2 &\longrightarrow \Sigma_{n-2}^0 \end{aligned}$$

where both functions associate to a pair (π, r), the permutation $Glue(r\pi)$. We have:

Lemma 3 *All the subsets $f_1^{-1}(\pi)$ have the same cardinality, $n(n-1)$, and all the subsets $f_2^{-1}(\pi)$ have the same cardinality, $\binom{n-2}{2}$.*

Proof: Given $\pi \in \Sigma_{n-1}^0$, an element of $f_1^{-1}(\pi)$ is obtained by picking an element and expanding it into an adjacency ($n-1$ possibilities), and then perform a reversal which cuts the new adjacency (n possibilities). Note that for every π, all resulting permutations in Σ_n^0 are different, and thus, there are $n(n-1)$ such permutations.

Given $\pi \in \Sigma_{n-2}^0$, an element of $f_2^{-1}(\pi)$ is obtained by picking two elements and expanding them into two adjacencies ($\binom{n-2}{2}$ possibilities), and then by performing the unique reversal which cuts both adjacencies (given the two adjacencies, this reversal is unique, since its two endpoints are determined). Thus, every permutation in Σ_{n-2}^0 is the image of $\binom{n}{2}$ elements of A_2. ■

Lemma 3 implies that, for any subset S of Σ^0_{n-1}, and T of Σ^0_{n-2}, the probability that a random pair (π, r) will be mapped on $S \cup T$ is smaller than $|S|/|\Sigma_{n-1}| + |T|/|\Sigma_{n-2}|$. Using a theorem due to Caprara [6], that establishes that the proportion of permutations in Σ_n that contains a hurdle is $\Theta(1/n^2)$, and Lemma 2, we easily obtain that the proportion of permutations in Σ^0_n that contains a hurdle is $\mathcal{O}(1/n^2)$. We thus have:

Theorem 4 *If π is a random permutation on n elements, and if r a random oriented reversal of π, then probability that r is unsafe is $\mathcal{O}(1/n^2)$.*

4 Commutation and Optimal Sequences of Reversals

In this section, we introduce some classical objects from discrete mathematics and theoretical computer science, trace monoids, and use them to study the set of optimal sequences sorting a permutation.

4.1 Trace monoids and Optimal Sequences of Reversals

Given a word w on an alphabet A, we are interested in the set of all words that can be generated by exchanging consecutive letters of w according to fixed rules. Such sets are studied under the name of traces and are defined formally in the next paragraph.

Trace monoids. Let A be an alphabet and $\theta \subseteq A \times A$ an irreflexive and symmetric relation on A. When the pair (x, y) is in θ, we say that the letters x and y *commute*. Two words w and w' over the alphabet A are *equivalent* with respect to the relation θ if we can transform w into w' by applying a finite sequence of commutations.
For example, if $A = \{a, b, c\}$, $\theta = \{(a, b), (a, c)\}$, the words $abacbbca$ and $bcababac$ are equivalent since we can transform one into the other with the following sequence of commutations:

$$\underline{ab}acbbca \longrightarrow ba\underline{ac}bbca \longrightarrow b\underline{ac}abbca \longrightarrow bcaabb\underline{ca} \longrightarrow bca\underline{ab}bac \longrightarrow bcababac.$$

The *trace monoid* T associated to A and θ, is the set of equivalence classes of words over A with respect to the relation θ, such classes are called *traces*. For example, with $A = \{a, b, c\}$ and $\theta = \{(a, b), (a, c)\}$, the sets of words $\{aabc, abac, baac, abca, baca, bcaa\}$ and $\{bc\}$ are two traces.

The concept of trace monoids was introduced by Cartier and Foata [8], under the name of partially commutative monoids, in relationship to combinatorial problems. It was later shown to have connections with the theory of formal languages and with concurrent systems. There is a rich corpus of results related to these objects, both from a mathematical and algorithmic point of views [9, 10].

Sets of Optimal Sequences as Traces In this section, we show that the set of optimal sequences sorting a permutation can be described in terms of traces. Recall that a reversal can be identified with the set of displaced elements in the permutation. A sequence of reversals is thus a word over the alphabet $\mathcal{P}_n$ of all non-empty subsets of $\{1, \ldots, n\}$. Two letters S and T of this alphabet are said to *commute* if one of the following holds:

1. $S \cap T = \emptyset$
2. $S \subset T$ or $T \subset S$

We will denote by T_n the trace monoid associated to $\mathcal{P}_n$ and the above commutation relation. We have the following:

Proposition 5 *Let π be a signed permutation of Σ_n. The set of optimal sequences of reversals sorting π is a union of traces belonging to the trace monoid T_n.*

Proof: We have to show that if a sequence $R = r_1 r_2 \ldots r_d$ of reversals sorts a permutation π, then all sequences in the trace of R also sorts the permutation. Let S and T be the sets corresponding to two consecutive reversals in R, and let π' be the permutation obtained just before applying T.
If $S \cap T = \emptyset$, then the two reversals affect disjoint intervals in the permutation π', and changing their order yields a well defined sequence of reversals that sorts π'.

If $S \subset T$, since T is well defined in π' and S is well defined in $T\pi'$, the elements of S must be consecutive also in π', thus the sequence $TS\pi'$ is well defined. The set T can thus be described as three consecutive intervals ASB of π', and both $ST\pi'$ and $TS\pi'$ transform these intervals into $-BS-A$. ■

Consider, for example, the permutation $\pi = (0\ \ 4\ \ -3\ \ 1\ \ 2\ \ 5)$. The sequence $R = \{1,2\}\{4\}\{1,2,3,4\}$ is an optimal sorting sequence, so is any permutation of the three reversals, since they all commute.
The trace monoid T_n is the best one, in the sense that no other commutation law holds between consecutive reversals. Indeed, two reversals defined by the sets S and T, such that $S \cap T \neq \emptyset$, and such that neither of them is contained in the other, cannot commute: if S is well defined in $T\pi'$, then S is not well defined in π.

Cartier-Foata Normal Form. Among the fundamental concepts related to trace monoids is the *normal form of a trace*. Indeed a trace is a (finite but possibly huge) set of words, and it would be interesting to represent such a set by a unique (canonical) word, that would allow to characterize it completely. In order to achieve this goal, the concept of normal form was introduced, which consists in defining a property of words that is verified by exactly one word in every trace.
Let A be an alphabet, θ a set of commutation rules over A and w a word of length n over A. A word w is said to be in *Cartier-Foata Normal Form* if w can be decomposed in consecutive subwords, say $u_1 \mid \ldots \mid u_m$, such that:

- every pair (x, y) of letters in u_i commute, for every $i \in [1..m]$,
- for $i > 1$, and for every letter x in u_i, there is at least one letter y in u_{i-1} such that x and y do not commute,
- u_i is a non-empty increasing word, in the lexicographic order induced by A,

Theorem 6 Cartier-Foata (69) *Let T be a trace monoid. For every trace $t \in T$, there is a unique word in t that is in Cartier-Foata Normal Form.*

The number of subwords in the Cartier-Foata Normal Form of a word is called its *stack height*.
For example, for

$$\pi = (0\ \ -4\ \ 1\ \ -3\ \ 6\ \ -7\ \ -5\ \ 2\ \ 8),$$

we have $d(\pi) = 6$, and the set of optimal sorting sequences is the union of 4 traces belonging to T_7:

$$\begin{array}{l}
\{2,5,6,7\}\{6,7\}\{7\} \mid \{1,2,3\}\{1,2,3,4\}\{2,3\} \\
\{2,5,7\} \mid \{1,2,3,6\}\{1,2,3,4,6\}\{2,3,6\} \mid \{3,4,5\}\{3,4,5,6\} \\
\{2,5,7\} \mid \{1,2,3,4,6\} \mid \{1,3,4,5\}\{1,3,4,5,6\} \mid \{1,2,3\}\{2,3\} \\
\{2,5,7\} \mid \{2,3,6\} \mid \{3,5\}\{3,5,6\} \mid \{1,2,3\}\{1,2,3,4\}
\end{array}$$

Their stack heights are, respectively, 2, 3, 4, and 4. The largest trace is the first one, with 180 elements in it. The second one has size 12, and the two others have size 4.

4.2 Applications

Analyzing a trace. As a consequence of the results of Section 4.1, we can use the algorithmic tools related to the theory of traces. We give three examples of such applications, in which R is an optimal sequence that sorts a permutation π, belonging to a trace t.

1. A first problem is the exhaustive generation of all optimal sequences in the trace t. This problem reduces to generating all the linear extensions of the partially ordered set associated to a trace [18]. The problem of generating linear extensions is classical in computer science and recent papers give efficient algorithms in amortized constant time [1] [15, 7].

1. By amortized constant time, we mean that the complete set of linear extensions will be generated in time proportional to the size of this set.

2. If we are interested in the size of t, we can apply the randomized algorithm described in [4] for approximating the number of linear extensions of a partially ordered set.

Question 7 *What is the complexity of computing the size of a trace of T_n?*

3. When one is interested in the combinatorial study of big sets of objects, a classical approach is to generate enough random elements of these set to get an idea on the properties of the "average object", or on the distribution of some combinatorial parameters in these sets. If we want to obtain a *random* optimal sequence belonging to t, we can use, for example, the polynomial time algorithm proposed in [11, 5] for the random generation of linear extensions of posets.

Analyzing the set of optimal sequences. As we saw above, the theory of trace monoids provides useful tools for dealing with one trace. But, in most of the cases, the set of optimal sequences is a set of traces. In this section, we discuss experiments on a large number of permutations that show the diversity, and structure, of the sets of optimal sequences.

1. The exhaustive generation of the set of optimal sequences is costly [17], and the problem of generating only one element in every different trace is currently unsolved:

Question 8 *What is the number of traces? And what is the complexity of generating an element of every trace?*

One might want to extend the notion of commutation to more than two elements, and consider general rewriting rules that would allow to generate an element of a different trace working on an element of a given trace. Such rules exist. For example, for a sequence of three reversals $R = S_1S_2S_3$, if S_3 is the symmetric difference of S_1 and S_2, then R is equivalent to $S_3S_2S_1$. However, in the general case, we know that a bounded set of rewriting rules will not allow the generation of all traces. Indeed, for each d, there exists a permutation π such that $d(\pi) = d$, and π has exactly two sorting sequences. The following permutation is an example, for d = 6:

$$(0\ \ 6\ \ 5\ \ 1\ \ -10\ \ -2\ \ 11\ \ 8\ \ -3\ \ -9\ \ 4\ \ 7\ \ 12)$$

There are two possible oriented reversals that can be applied on the above permutation. After a first reversal, the choice becomes deterministic, with only one possible reversal at each step.

2. We also performed extensive computations, to generate exhaustively the set of optimal sequences for many different signed permutations and observed that the sizes of these traces vary dramatically. In most of the cases, one of the traces is much bigger than the others. This leads us to the following, quite informal, questions.

Question 9 *For a signed permutation π, does the set of traces contain a class whose size is proportional to the number of optimal sequences of reversals sorting π ? What is the complexity of generating an optimal sequence of reversals belonging to the biggest trace?*

As an illustration of the variations of the size of a trace, consider the following permutation on 32 elements:

$$(0\ \ 13\ \ -19\ \ -29\ \ -23\ \ 16\ \ -31\ \ -10\ \ 7\ \ -27\ \ 12\ \ -30\ \ -14\ \ 15\ \ -20\ \ -6\ \ 3$$
$$6\ \ -2\ \ -22\ \ -28\ \ -21\ \ -5\ \ -1\ \ 11\ \ 4\ \ 8\ \ 25\ \ 9\ \ 18\ \ -17\ \ -24\ \ 32)$$

One of the trace of the set of optimal sequences of π has size larger than 15 049 359 360 000, but others have barely a few thousand elements. The key parameter to explain those differences is the stack height. It had been shown, in the field of distributed computing, that this parameter is strongly related to the ability of a program to be parallelized [16]. Briefly, if we consider the letters of the alphabet A as tasks that can be performed by a parallel system, and the set of commutation rules as the set of pairs of tasks that can be processed in parallel, a trace is a description of a program for this system. In this context, a subword u_i of the Cartier-Foata Normal Form is the

set of tasks that can be processed at the same time by the system. The subword u_{i+1} is the set of tasks that can be processed once all the tasks of u_i are completed. The ratio between the length of a word and its stack height, called the *concurrency degree*, has been considered by several authors as a good measure of the ability of the corresponding program to be parallelized.
If we consider that the evolution process is a kind of program, it leads to some question related to the parallelization of an evolution process and the stack height of a trace. The goal could be to obtain a qualitative classification of traces sorting a given permutation in terms of their size. A natural question is whether a highly parallelizable evolution scenario is plausible or not. Is there a biological interpretation of the stack height of a sequence of reversals?

3. Following the question of the last paragraph, we studied a more global aspect of stack heights, asking for the minimal stack height that could be achieved among the traces associated to a given permutation. This lead us to the following sorting strategy that "promotes" commutation:

6. *Commutation:* This strategy chooses, among all oriented reversals defined in π, a maximal subset in which all reversals commute. All those reversals are applied to π, and the process is repeated until the resulting permutation is positive.

The following table compares the mean stack heights obtained by sorting 10 000 random permutations, of lengths 32 to 224, according to the maximal length strategy, the maximal score strategy, and the commutation strategy.

	Maximal Length	Maximal Score	Commutation
$n = 32$	19.3	10.2	7.4
64	42.9	19.6	11.6
96	66.7	28.3	14.8
128	91.2	36.7	17.6
160	115.5	45.3	20.0
192	139.9	52.8	22.3
224	164.6	60.1	24.2

Note that the maximal length strategy have a very negative impact on commutation, meaning that this strategy yields almost deterministic evolutionary histories. On the other hand, the strategy that promotes commutation drastically reduces stack heights, yielding a number of equivalent sequences that grows factorially with an increasing function of the length.

5 Conclusion

Even if the reversal distance provides a good estimate of the evolutionary distance between the two species, the number of optimal sequences realizing this distance is typically to big to be of any use in telling the real evolutionary history between two genomes. We showed, in this paper, that other parameters of optimal sequences, such as the length of reversals, or the "degree" of commutation, can be used as secondary sorting strategies, yielding sequences that have very different structures. While some of these parameters have an immediate biological meaning – short reversals vs long reversals, for example –, others, such as stack height do not have an easy evolutionary interpretation.
In future work, we want to identify other combinatorial parameters that are related to set of optimal sequences, especially those already identified by biologists as evolutionary relevant. One that seems rich is the concept of *conserved segments*, that has rich interactions with the sorting by reversal problem.

6 Acknowledgements

Tzvika Hartman was supported in part by a grant (awarded to his advisor, Prof. Ron Shamir) by the Israel Science Foundation formed by the Israel Academy of Sciences and Humanities.

References

[1] D. Bader, B. Moret and M. Yan. A Linear-Time Algorithm for Computing Inversion Distance Between Signed Permutations with an Experimental Study. *J. Comput. Biol.* 8:483–491, 2001.

[2] A. Bergeron. A Very Elementary Presentation of the Hannenhalli-Pevzner Theory. In *CPM'01*, volume 2089 of *Lecture Notes in Computer Science*, pages 106–117, Springer, Berlin, 2001.

[3] A. Bergeron and F. Strasbourg. Expriments in computing sequences of reversals. In *WABI'01*, volume 2149 of *Lecture Notes in Computer Science*, pages 164–174, Springer, Berlin, 2001.

[4] G. Brightwell and P. Winkler. Counting linear extensions. *Order*, 8:225–242, 1991.

[5] R. Bubley and M. Dyer. Faster random generation of linear extensions. *Discrete Math.*, 201(1-3):81–88, 1999.

[6] A. Caprara. On the tightness of the alternating-cycle lower bound for sorting by reversals. *Journal of Combinatorial Optimization*, 3:149–18, 1999.

[7] E.R. Canfield and S.G. Williamson. A loop-free algorithm for generating the linear extensions of a poset. *Order*, 12:57–75, 1995.

[8] P. Cartier and D. Foata. *Problèmes combinatoires de commutations et réarrangements*, volume 85 of *Lecture Notes in Math.*, Springer, Berlin, 1969.

[9] V. Diekert and Y. Métivier. Partial commutation and traces. In Rozenberg G. and Salomaa A., editors, *Handbook of formal languages, Volume 3*, pages 457–533. Springer, Berlin, 1997.

[10] V. Diekert and G. Rozenberg, editors. *The book of traces*. World Scientific Publishing, River Edge (NJ), 1995.

[11] P. Felsner and L. Wernisch. Markov chains for linear extensions, the two-dimensional case. In *SODA'97*, pages 239–247, ACM Press, 1997.

[12] S. Hannenhalli and P. Pevzner. Transforming Cabbage into Turnip: Polynomial Algorithm for Sorting Signed Permutations by Reversals. *J. of the ACM* 6(1):1–27, 1999.

[13] J. Kececioglu and D. Sankoff. Efficient bounds for oriented chromosome inversion distance. In *CPM'94*, volume 807 of *Lecture Notes in Computer Science*, pages 307–325, Springer, Berklin, 994.

[14] H. Kaplan, R. Shamir and R. Tarjan A Faster and Simpler Algorithm for Sorting Signed Permutations by Reversals. *SIAM J. Comput.* 29(3):880–892, 1999.

[15] G. Pruesse and F. Ruskey. Generating linear extensions fast. *SIAM J. Comput.*, 23(2):373–386, 1994.

[16] N. Saheb. Concurrency measure in commutation monoids. *Dicrete Appl. Math.*, 24(1-3):223–236, 1989.

[17] A. Siepel. An algorithm to find all sorting reversals. In Proceedings of *RECOMB'02*, ACM Press, pp 281-290, 2002.

[18] X. G. Viennot. Heaps of pieces, I: basic definitions and combinatorial lemmas. In *Combinatoire énumérative*, volume 1234 of *Lecture Notes in Math.*, pages 321–350. Springer, Berlin, 1985.