Average-case complexity analysis of perfect sorting by reversals

Mathilde Bouvel
with Cedric Chauve, Marni Mishna and Dominique Rossin

Algorithms and Permutations 2012
Outline of the talk

1. The context: Sorting by reversals
2. The problem we consider: Perfect sorting by reversals
3. Average-case complexity analysis
4. Restriction to the class of separable permutations
5. Conclusion and future work under non-uniform distributions
The context: Sorting by reversals

Biological motivations

Reconstruction of evolution scenarios

Operation on genome = reversal

- Model for genome = signed permutation
- Reversal = reverse a window of the permutation while changing the signs

\[
\begin{align*}
1 & \ 7 \ 6 \ 10 \ 9 \ 8 \ 2 \ 11 \ 3 \ 5 \ 4 \\
\downarrow & \quad \text{Reversal} \quad \downarrow \\
1 & \ 7 \ 6 \ 10 \ 9 \ 8 \ 2 \ 4 \ 5 \ 3 \ 11
\end{align*}
\]
The context: Sorting by reversals

Sorting by reversals: the problem and solution

The problem:

- **input**: Two signed permutations $\sigma_1$ and $\sigma_2$
- **output**: A parsimonious scenario from $\sigma_1$ to $\sigma_2$ or $\overline{\sigma_2}$

Parsimonious = shortest, i.e. minimal number of reversals. Without loss of generality, $\sigma_2 = \text{Id} = 1 \ 2 \ldots n$

The solution:

- Hannenhalli-Pevzner theory
- Polynomial algorithms: from $O(n^4)$ to $O(n \sqrt{n \log n})$

Remark: the problem is $NP$-hard when permutations are unsigned.
The problem we consider: Perfect sorting by reversals

Definition and motivation

Perfect sorting by reversals: do not break common intervals

Common interval between \( \sigma_1 \) and \( \sigma_2 \): windows of \( \sigma_1 \) and \( \sigma_2 \) containing the same elements (with no sign)

Example: \( \sigma_1 = 5 \, 7 \, 3 \, 7 \, 6 \, 2 \, 4 \) and \( \sigma_2 = 6 \, 4 \, 7 \, 1 \, 3 \, 2 \, 5 \)

When \( \sigma_2 = Id \), interval of \( \sigma_1 \) = window forming a range (in \( \mathbb{N} \))

Example: \( \sigma_1 = 4 \, 7 \, 5 \, 6 \, 3 \, 1 \, 2 \)

Biological argument: groups of identical (or homologous) genes appearing together in two species are likely to be

- together in the common ancestor
- never separated during evolution
The problem we consider: Perfect sorting by reversals

Algorithm and complexity

The problem:

- **INPUT**: Two signed permutations $\sigma_1$ and $\sigma_2$
- **OUTPUT**: A parsimonious perfect scenario (=shortest among perfect scenarios) from $\sigma_1$ to $\sigma_2$ or $\sigma_2$

Without loss of generality, $\sigma_2 = Id = 1\ 2\ \ldots\ n$

Watch out!: Parsimonious perfect $\not\Rightarrow$ parsimonious

Complexity: *NP*-hard problem

Algorithm [Bérard, Bergeron, Chauve, Paul]: take advantage of decomposition trees to produce a *FPT* algorithm $\left(2^p \cdot n^{O(1)}\right)$
The problem we consider: Perfect sorting by reversals

Strong intervals of (signed) permutations

- **Strong interval** = does not overlap any other interval
- Interval $I$ is strong iff $\forall J$, $I \subseteq J$ or $J \subseteq I$ or $I \cap J = \emptyset$

**Example** of intervals and strong intervals:

```
5  6  7  9  4  3  1  2  8  10  17  13  15  12  11  14  18  19  16
```

Trivial intervals are always among strong intervals

Mathilde Bouvel

Average-case complexity analysis of perfect sorting by reversals
The problem we consider: Perfect sorting by reversals

Strong intervals of (signed) permutations

- **Strong interval** = does not overlap any other interval
- Interval $I$ is strong iff $\forall J$, $I \subseteq J$ or $J \subseteq I$ or $I \cap J = \emptyset$

Example of intervals and strong intervals:

```
5 6 7 9 4 3 1 2 8 10 17 13 15 12 11 14 18 19 16
```

Trivial intervals are always among strong intervals
The problem we consider: Perfect sorting by reversals

Mathilde Bouvel

The problem we consider: Perfect sorting by reversals

Strong intervals of (signed) permutations

- **Strong interval** = does not overlap any other interval
- Interval \( I \) is strong iff \( \forall J, I \subseteq J \) or \( J \subseteq I \) or \( I \cap J = \emptyset \)

Example of intervals and strong intervals:

```
5 6 7 9 4 3 1 2 8 10 17 13 15 12 11 14 18 19 16
```

Trivial intervals are always among strong intervals
The problem we consider: Perfect sorting by reversals

Decomposition trees of (signed) permutations

Also known as strong interval trees

- Inclusion order on strong intervals: a tree-like ordering

Computation: in linear time
The problem we consider: Perfect sorting by reversals

Decomposition trees of (signed) permutations

Quotient permutation =
order of the children (that are intervals)

Example:

\[
\begin{array}{c}
\sigma = \begin{bmatrix} 1 & \ldots & 9 \\ 2 & 4 & 1 & 3 \\ 5 & \ldots & 7 \\ 9 & \ldots & 9 \\
1 & \ldots & 4 \\ \ldots & 8
\end{bmatrix}
\end{array}
\]

Two types of nodes:
- Linear nodes (□):
  - increasing, i.e. quotient permutation = 1 2 \ldots k
  ⇒ label □
  - decreasing, i.e. quotient permutation = k (k − 1) \ldots 2 1
  ⇒ label □
- Prime nodes (○): the quotient permutation is simple

Simple permutations:
the only intervals are 1, 2, \ldots, n and \sigma

Example: 425163, i.e.

\[
\begin{array}{c}
\sigma = 425163
\end{array}
\]
The problem we consider: Perfect sorting by reversals

Simplified decomposition tree

Remark: redundant information ⇒ forget the leaves and intervals
The problem we consider: Perfect sorting by reversals

Simplified decomposition tree

Remark: redundant information ⇒ forget the leaves and intervals

Tree uniquely defined by \( \{ \text{labels of internal nodes, signs of the leaves} \} \)
The problem we consider: Perfect sorting by reversals

Idea of the algorithm to solve perfect sorting

Put labels + or − on the nodes of the decomposition tree of $\sigma$

- Leaf: sign of the element in $\sigma$
- Linear node: + for $\uparrow$ (increasing) and − for $\downarrow$ (decreasing)
- Prime node whose parent is linear: sign of its parent
- Other prime node: ???
  $\leftrightarrow$ Test labels + and − and choose the shortest scenario

Algorithm:

- Perform Hannenhalli-Pevzner (or improved version) on prime nodes
- Signed node belongs to scenario iff its sign is different from its linear parent
The problem we consider: Perfect sorting by reversals

Example of labeled decomposition tree

Mathilde Bouvel

Average-case complexity analysis of perfect sorting by reversals
The problem we consider: Perfect sorting by reversals

Complexity results

**Complexity:**
- $O(2^p n \sqrt{n \log n})$, with $p = \#$ prime nodes
- polynomial on separable permutations ($p = 0$)

**Our work:**
- polynomial with probability 1 asymptotically
- polynomial on average
- in a parsimonious perfect scenario for separable permutations
  - average number of reversals $\sim 1.27n$
  - average length of a reversal $\sim 1.054 \sqrt{n}$

**Probability distribution:** always uniform
"Average shape" of decomposition trees

Enumeration of simple permutations: asymptotically $\frac{n!}{e^2}$

⇒ Asymptotically, a proportion $\frac{1}{e^2}$ of decomposition trees are reduced to one prime node.

Thm: Asymptotically, the proportion of decomposition trees made of a prime root with children that are leaves or *twins* is 1.

twin = linear node with only two children, that are leaves

Consequence: Asymptotically, with probability 1, the algorithm runs in polynomial time.

Rem.: The number of *twins* follows a Poisson distribution of parameter 2.
Average complexity on permutations of size $n$:

$$
\sum_{p=0}^{n} \frac{\#\{\sigma \text{ with } p \text{ prime nodes}\} \cdot C \cdot 2^p n \sqrt{n \log n}}{n!} \cdot 2^p \cdot 48(n-1)! 
$$

**Thm:** When $p \geq 2$, the number of (unsigned) permutations of size $n$ with $p$ prime nodes is at most $\frac{48(n-1)!}{2^p}$.

**Proof:** induction on $p$

**Consequence:** Average complexity on permutations of size $n$ is $\leq 51Cn \sqrt{n \log n}$. In particular, **polynomial on average**.
Separable (= commuting) permutations

Def.: Commuting permutation = permutation sorted by a scenario where any pair of reversals commutes (= does not overlap)

Rem.: Here, scenario = set of intervals, in any order

Equivalently: Commuting permutation = permutation with no prime node in its decomposition tree
Also called separable permutations.
Scenarios for separable permutations

In general, in the computed scenario, reversals are:

- linear nodes with label different from its linear parent
- inside prime nodes

Prop.: No $\square \rightarrow \square$ nor $\square \rightarrow \square$ edge in decomposition trees

Consequence: For separable permutations, reversals = linear nodes with label different from its linear parent

$\approx \begin{cases} 
\text{all internal nodes except the root} \\
+ \text{leaves with label different from its parent}
\end{cases}$

Reversals $\approx$ internals nodes $- \text{the root} + \text{half of the leaves}$

$\Rightarrow$ The shape of the tree is sufficient to study reversals
Bijection between separable perm. and Schröder trees

Decomposition trees of (unsigned) separable permutation

Schröder trees

+ label □ or □ on the root

size of $\sigma$ $\longleftrightarrow$ number of leaves
reversal of length $\geq 2$ $\longleftrightarrow$ internal node except the root
reversal of length 1 $\longleftrightarrow$ some leaves (half of them)
length of a reversal $\longleftrightarrow$ size ($= \# $ leaves) of the subtree
Parameters on Schröder trees

Two parameters on Schröder trees:
- Number of internal nodes
- Pathlength = sum of the sizes of the subtrees

Study their average gives access to:
- Average number of reversals
- Average length of a reversal

in a scenario for a separable permutation

Analytic combinatorics:
average from bivariate generating functions \( S(x, y) = \sum s_{n,k} x^n y^k \)
where \( s_{n,k} \) = number of Schröder trees with \( n \) leaves and \( k \)
internal nodes (resp. pathlength \( k \))
Average value of a parameter (number of internal nodes)

Definition: \( S(x, y) = \sum s_{n,k} x^n y^k \),
where \( s_{n,k} \) = number of Schröder trees with \( n \) leaves and \( k \)
internal nodes

Combinatorial specification: \( S = \bullet + \sum S \cdot S \cdot \ldots \cdot S \)

Functional equation: \( S(x, y) = x + y \frac{S(x,y)^2}{1-S(x,y)} \)

Solution: \( S(x, y) = \frac{(x+1)-\sqrt{(x+1)^2-4x(y+1)}}{2(y+1)} \)

Average number of internal nodes = \( \frac{\sum_k k s_{n,k}}{\sum_k s_{n,k}} = \frac{[x^n] \frac{\partial S(x,y)}{\partial y} |_{y=1}}{[x^n] S(x,1)} \)

Asymptotic estimate of \( [x^n] S(x, 1) \) when \( n \to +\infty \): from
asymptotic estimate of \( S(x, 1) \) when \( x \to \) dominant singularity
Results

Application of the methodology of [Flajolet, Sedgewick]

In Schröder trees with $n$ leaves:
- Average number of internal nodes: $\sim \frac{n}{\sqrt{2}}$
- Average pathlength: $\sim 1.27n^{\frac{3}{2}}$

In scenarios for separable permutations of size $n$:
- Average number of reversals: $\sim \frac{1+\sqrt{2}}{2}n$
- Average length of a reversal: $\sim 1.054 \sqrt{n}$
Results so far and future work

Perfect sorting by reversals for signed permutations:

- \textit{NP}-hard problem
- algorithm running in polynomial time
  - on average
  - asymptotically with probability 1
  - for the \textit{uniform} distribution on permutations of size \(n\)

Special case of separable permutations (no prime nodes):

- expected length of a parsimonious perfect scenario \(\sim 1.27n\)
- expected length of a reversal in such a scenario \(\sim 1.054 \sqrt{n}\)

Using analytic combinatorics techniques

\textbf{Work in progress}: influence on the probability distribution to obtain a model closer to the biological observations

Mathilde Bouvel

Average-case complexity analysis of perfect sorting by reversals
Non-uniform distributions

Results under the uniform distribution: mostly theoretical results
Biological data: not uniformly distributed (few prime nodes, ...)

Combinatorial specification as decomposition trees: allows to introduce some constraints on the prime nodes (maximal arity, number, ...) for:
- the study of parameters (on average)
- (Boltzmann) random generation
under non uniform distributions

Comparison between these results (theoretical or simulation) and biological data
- to describe models that are closer to the biological reality
- to identify non-random evolution (w.r.t. a good distribution)