

# Average-case complexity analysis of perfect sorting by reversals

Mathilde Bouvel Cedric Chauve Marni Mishna  
Dominique Rossin

Séminaire des thésards



LIAFA

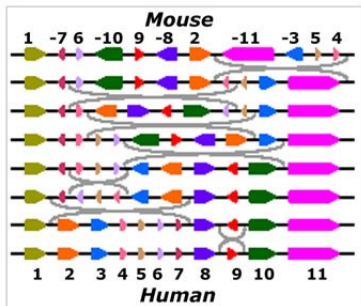


# Outline of the talk

- 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 commuting permutations
- 5 Open problems and perspectives

The context: Sorting by reversals

# Biological motivations



© Genome Research

Biologists have found that the related genes in man and mouse are not chaotically distributed over the genomes, but form 'conserved blocks' instead. These conserved blocks reveal the genetic organization of the common ancestor of human and mouse, allowing Pevzner and Tesler to reconstruct a rearrangement scenario of man-mouse evolution. Genomic sequences reveal at least 11 synteny blocks (where human and mouse genes are in the same order) of one million DNA letters or longer on the X chromosome. They provide evidence of at least 7 inversions (a type of rearrangement) which emanate from a common ancestor in the middle. Two of the 11 blocks show evidence of extensive micro-rearrangements. (Graphic by Glenn Tesler, UCSD)

Reconstruction of evolution scenarios

↪ Operation on genome = reversal

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

$$1 \bar{7} 6 \bar{10} 9 \bar{8} 2 \bar{11} \bar{3} 5 4$$

$$\Downarrow \text{Reversal} \Downarrow$$

$$1 \bar{7} 6 \bar{10} 9 \bar{8} 2 \bar{4} \bar{5} 3 11$$



# 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,  $\sigma_2 = Id = 1\ 2\ \dots\ n$

## 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.

## Definition and motivation

Perfect sorting by reversal: 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:**  $\sigma_1 = 5 \overline{1} \overline{3} \overline{7} 6 \overline{2} 4$  and  $\sigma_2 = 6 \overline{4} \overline{7} \overline{1} \overline{3} 2 \overline{5}$

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

**Example:**  $\sigma = 4 \overline{7} \overline{5} \overline{6} 3 \overline{1} 2$

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

- together in the common ancestor
- never separated during evolution

# Algorithm and complexity

## The problem:

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

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

**Beware:** Parsimonious perfect  $\not\Rightarrow$  parsimonious

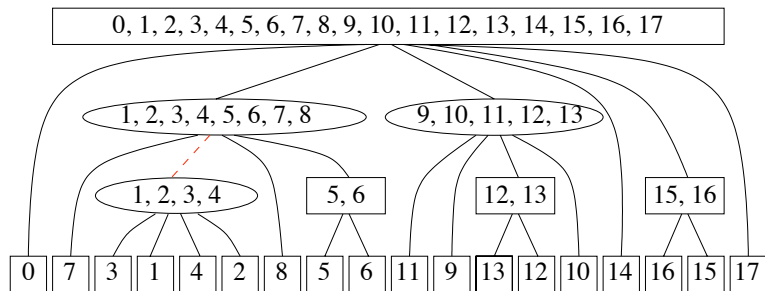
**Complexity:** NP-hard problem

**Algorithm** [Bérard, Bergeron, Chauve, Paul]: take advantage of decomposition trees to produce a “quasi-polynomial” algorithm

The problem we consider: Perfect sorting by reversals

## Decomposition trees of (signed) permutations

- **Strong interval** = does not overlap any other interval
- Inclusion order on strong intervals: a **tree-like** ordering



**Computation:** in linear time

# Decomposition trees of (signed) permutations

Two types of nodes:

- **Linear nodes** ( $\square$ ):
  - increasing, *i.e.* quotient permutation =  $1\ 2\ \dots\ k$   
 $\Rightarrow$  label  $\boxplus$
  - decreasing, *i.e.* quotient permutation =  $k\ (k-1)\ \dots\ 2\ 1$   
 $\Rightarrow$  label  $\boxminus$
- **Prime nodes** ( $\circ$ ): the quotient permutation is simple

**Simple permutations:** the only intervals are  $1, 2, \dots, n$  and  $\sigma$



## Idea of the algorithm

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

- Leaf: sign of the element in  $\sigma$
- Linear node: + for  $\boxplus$  (increasing) and – for  $\boxminus$  (decreasing)
- Prime node whose parent is linear: sign of its parent
- Other prime node: ???
  - ↪ 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

# Complexity results

## Complexity:

- $O(2^p n \sqrt{n \log n})$ , with  $p = \#$  prime nodes
- polynomial on commuting permutations

## Our work:

- polynomial with probability 1 asymptotically
- polynomial on average
- in a parsimonious scenario for commuting permutations
  - average number of reversals  $\sim 1.2n$
  - average length of a reversal  $\sim 1.02 \sqrt{n}$

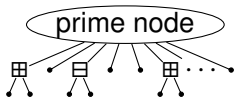
# Average shape of decomposition trees

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.

# Average complexity

**Average complexity** on permutations of size  $n$ :

$$\frac{\sum_p \#\{\sigma \text{ with } p \text{ prime nodes}\}}{n!} \leq C 2^p n \sqrt{n \log n}$$

**Thm:** When  $p \geq 2$ ,

number of permutations of size  $n$  with  $p$  prime nodes  $\leq \frac{48(n-1)!}{2^p}$

**Proof:** induction on  $p$

**Consequence:** Average complexity on permutations of size  $n$  is  $\leq 50Cn \sqrt{n \log n}$ . In particular, **polynomial on average**.

## Commuting (separable) permutations

**Def.:** No prime node in decomposition tree

In general, in the computed perfect sorting scenario, reversals =

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

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

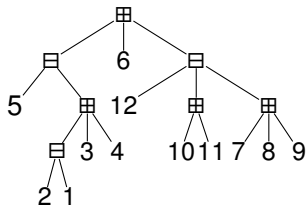
$$= \left\{ \begin{array}{l} \text{all internal nodes except the root} \\ \text{leaves with label different from its parent} \end{array} \right.$$

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

## Restriction to the class of commuting permutations

## Bijection between commuting perm. and Schröder trees

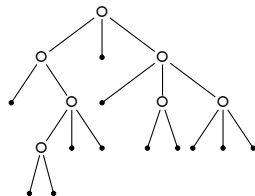
Decomp. tree of commuting perm.



$$\sigma = 5 \ 2 \ 1 \ 3 \ 4 \ 6 \ 12 \ 10 \ 11 \ 7 \ 8 \ 9$$

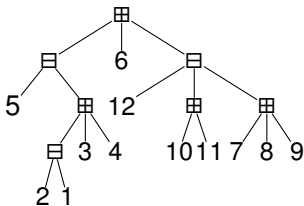
Only nodes  $\boxplus$  and  $\boxminus$ Labels  $\boxplus$  and  $\boxminus$  alternate

Schröder trees

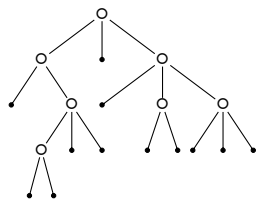
arity  $\geq 2$ + label  $\boxplus$  on the root

# Bijection preserved parameters

Decomp. tree of commuting perm.



Schröder trees



+ label  $\boxplus$  on the root

size of $\sigma$	$\longleftrightarrow$	number of leaves
reversal (except leaf)	$\longleftrightarrow$	internal node (except root)
length of a reversal	$\longleftrightarrow$	number of leaves in the subtree

# Average number of reversals

Average number of reversals for commuting permutations

$$\begin{aligned}
 &= \begin{cases} \text{average number of internal nodes (except root)} \\ + \text{average number of leaves with label different from its parent} \end{cases} \\
 &= \text{average number of internal nodes} - 1 + n/2
 \end{aligned}$$

Focus on average number of internal nodes in Schröder trees:  
using **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.



# Generating function for average 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.

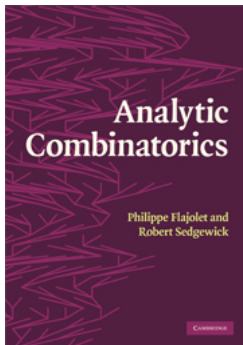
$$S = \bullet + S \begin{array}{c} \circ \\ \diagup \quad \diagdown \\ S \quad S \quad \cdots \quad S \end{array}$$

**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**  $= \frac{\sum_k k s_{n,k}}{\sum_k s_{n,k}} = \frac{[x^n] \frac{\partial S(x, y)}{\partial y} |_{y=1}}{[x^n] S(x, 1)}$

# From generating function to asymptotics



Development around singularity (here,  $3 - 2\sqrt{2}$ ):

- $S(x, 1) \sim \frac{2-\sqrt{2}}{2} - \frac{\sqrt{3\sqrt{2}-4}}{2} \left(1 - \frac{x}{3-2\sqrt{2}}\right)^{1/2}$
- $\frac{\partial S(x,y)}{\partial y} \Big|_{y=1} \sim \frac{3-2\sqrt{2}}{4\sqrt{3\sqrt{2}-4}} \left(1 - \frac{x}{3-2\sqrt{2}}\right)^{-1/2}$

Equivalent of coefficients:

- $[x^n]S(x, 1) \sim \frac{\sqrt{3\sqrt{2}-4}}{4} (3 + 2\sqrt{2})^n \frac{1}{\sqrt{\pi n^3}}$
- $[x^n] \frac{\partial S(x,y)}{\partial y} \Big|_{y=1} \sim \frac{3-2\sqrt{2}}{4\sqrt{3\sqrt{2}-4}} (3 + 2\sqrt{2})^n \frac{1}{\sqrt{\pi n}}$

Conclusion: 
$$\frac{[x^n] \frac{\partial S(x,y)}{\partial y} \Big|_{y=1}}{[x^n] S(x,1)} \sim \frac{3-2\sqrt{2}}{3\sqrt{2}-4} n \sim \frac{n}{\sqrt{2}}$$

## Results (1)

Average number of **internal nodes in Schröder trees**:

$$\frac{n}{\sqrt{2}} \text{ asymptotically}$$

This result is valid both for **unsigned** Schröder trees and for Schröder trees **with a sign** ( $\boxplus$  or  $\boxminus$ ) on the root.

Average number of **reversals for commuting permutations**:

$$\frac{n}{\sqrt{2}} - 1 + \frac{n}{2} \text{ i.e. } \frac{1 + \sqrt{2}}{2} n \text{ asymptotically}$$

**Remark:** Many reversals of length 1: confirm biological experiments.

# Average length of a reversal

Average length of a reversal =  $\frac{\text{average sum of the lengths of all reversals}}{\text{average number of reversals}}$

A reversal =  $\left\{ \begin{array}{l} \text{an internal node (except root)} \\ \text{a leaf with label different from its parent} \end{array} \right.$

Hence, **average sum of the lengths of all reversals** for commuting permutations

= average sum of the sizes of all subtrees in a Schröder tree  
 $-n$  (for the root)  $-n/2$  (for the leaves)

Focus on this **average sum**.

# Average sum of the lengths of all reversals

Use another **bivariate generating function**:

$$S(x, y) = \sum s_{n,k} x^n y^k,$$

where  $s_{n,k}$  = number of Schröder trees with  $n$  leaves and size of subtrees summing to  $k$ ,

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

**Solution:**  $\frac{\sum_k k s_{n,k}}{\sum_k s_{n,k}} = \frac{[x^n] \frac{\partial S(x, y)}{\partial y} |_{y=1}}{[x^n] S(x, 1)} \sim 2^{3/4} \sqrt{3 - 2\sqrt{2}} \sqrt{\pi n^3}.$

## Results (2)

Average **sum of the sizes of all subtrees in a Schröder tree**:

$$2^{3/4} \sqrt{3 - 2\sqrt{2}} \sqrt{\pi n^3} \text{ asymptotically}$$

Average **sum of the lengths of the reversals**: subtracting  $\frac{3}{2}n$  does not change the asymptotics.

Average **length of a reversal for commuting permutations**:

$$\frac{2^{7/4} \sqrt{3 - 2\sqrt{2}}}{1 + \sqrt{2}} \sqrt{\pi n} \simeq 1.02 \sqrt{n}$$



## Conclusion and perspectives

Perfect sorting by reversals for signed permutations:

- *NP*-hard problem
- algorithm running in polynomial time
  - ↪ on average
  - ↪ asymptotically with probability 1

Special case of commuting permutations:

- expected length of a parsimonious perfect scenario
- expected length of a reversal in such a scenario

**Perspective:**

extend this analysis to the Double-Cut-and-Join model.