Average-case complexity analysis of perfect sorting by reversals

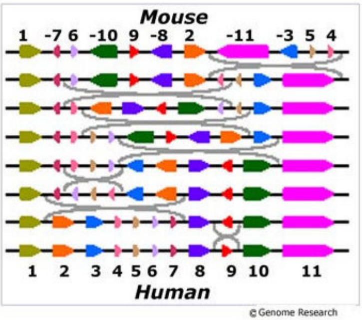
Mathilde Bouvel

with Cedric Chauve, Marni Mishna and Dominique Rossin

Algorithms and Permutations 2012

Outline of the talk

Biological motivations



Reconstruction of evolution scenarios
$\hookrightarrow$ Operation on genome = reversal

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

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

$\Downarrow$ Reversal $\Downarrow$

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

Sorting by reversals: the problem and solution

The problem:

- INPUT: Two signed permutations σ_1 and σ_2
- OUTPUT: A parsimonious scenario from σ_1 to σ_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 $\mathcal{O}(n^4)$ to $\mathcal{O}(n\sqrt{n \log n})$

Remark: the problem is *NP*-hard when permutations are unsigned.

Definition and motivation

Perfect sorting by reversals: do not break common intervals

Common interval between σ_1 and σ_2: windows of σ_1 and σ_2 containing the same elements (with no sign)
Example: $\sigma_1 = 5\ \bar{1}\ \bar{3}\ 7\ 6\ \bar{2}\ 4$ and $\sigma_2 = 6\ \bar{4}\ 7\ 1\ \bar{3}\ 2\ \bar{5}$

When $\sigma_2 = Id$, interval of σ_1 = window forming a range (in $\mathbb{N}$)
Example: $\sigma_1 = 4\ \bar{7}\ \bar{5}\ 6\ 3\ \bar{1}\ 2$

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

Algorithm and complexity

The problem:

- INPUT: Two signed permutations σ_1 and σ_2
- OUTPUT: A parsimonious perfect scenario (=shortest among perfect scenarios) from σ_1 to σ_2 or $\overline{\sigma_2}$

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 $\left(2^p \cdot n^{O(1)}\right)$

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:

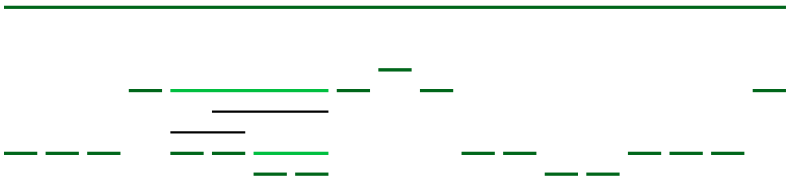
$$5 \quad \bar{6} \quad \bar{7} \quad 9 \quad 4 \quad \bar{3} \quad 1 \quad 2 \quad \bar{8} \quad \overline{10} \quad \overline{17} \quad 13 \quad \overline{15} \quad 12 \quad 11 \quad \overline{14} \quad 18 \quad \overline{19} \quad \overline{16}$$

Trivial intervals are always among strong intervals

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:



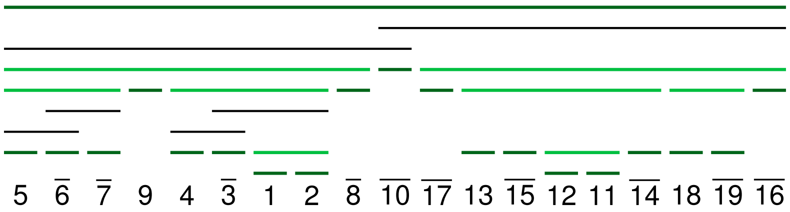
$5 \ \bar{6} \ \bar{7} \ 9 \ 4 \ \bar{3} \ 1 \ 2 \ \bar{8} \ \overline{10} \ \overline{17} \ 13 \ \overline{15} \ 12 \ 11 \ \overline{14} \ 18 \ \overline{19} \ \overline{16}$

Trivial intervals are always among strong intervals

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:

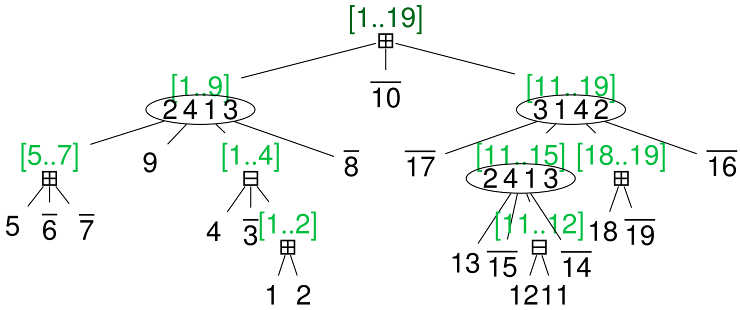


Trivial intervals are always among strong intervals

Decomposition trees of (signed) permutations

Also known as strong interval trees

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

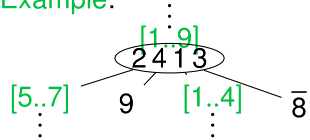


Computation: in linear time

Decomposition trees of (signed) permutations

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

Example:

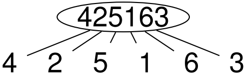


Two types of nodes:

- Linear nodes (□):
  - increasing, *i.e.* quotient permutation = 1 2 . . . k
  - $\Rightarrow$ label ⊞
  - decreasing, *i.e.* quotient permutation = k $(k-1)$ . . . 2 1
  - $\Rightarrow$ label ⊟
- Prime nodes (◯): the quotient permutation is simple

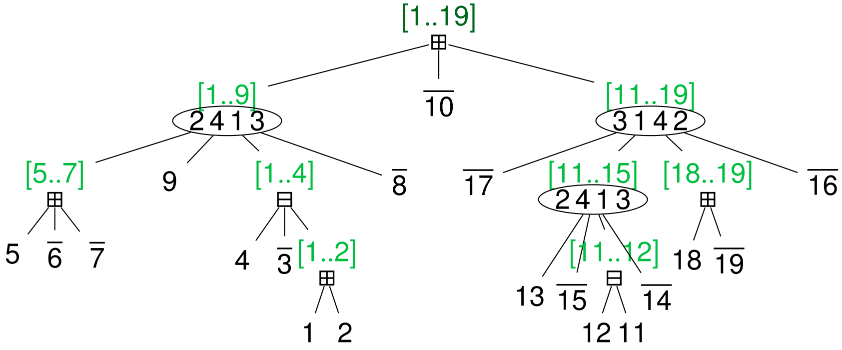
Simple permutations:
the only intervals are 1, 2,. . ., n and σ

Example: 425163, i.e.



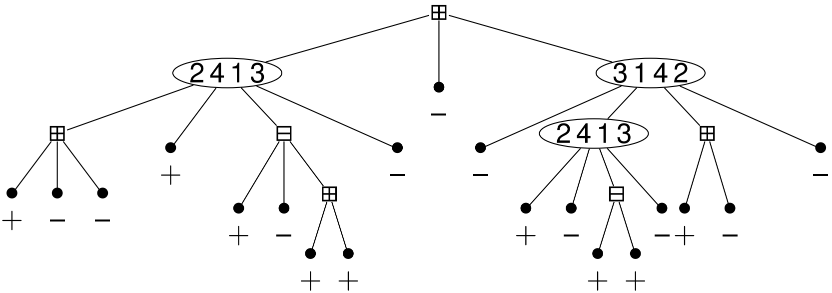
Simplified decomposition tree

Remark: redundant information ⇒ forget the leaves and intervals



Simplified decomposition tree

Remark: redundant information ⇒ forget the leaves and intervals



Tree uniquely defined by $\begin{cases}\text{labels of internal nodes} \\ +\text{signs of the leaves}\end{cases}$

Idea of the algorithm to solve perfect sorting

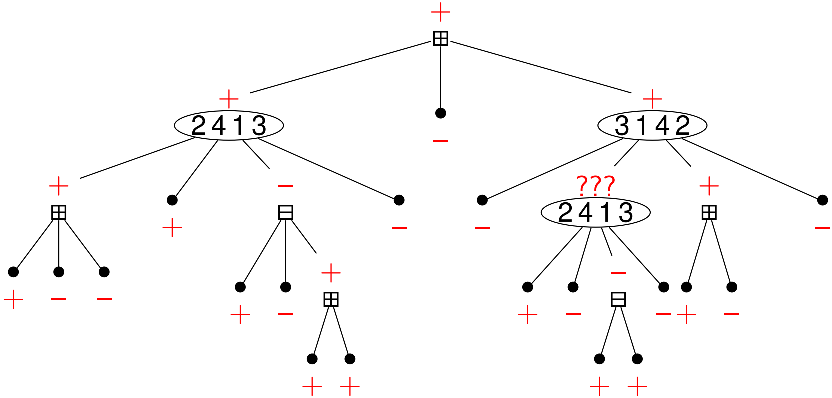
Put labels $+$ or $-$ on the nodes of the decomposition tree of σ

- Leaf: sign of the element in σ
- Linear node: $+$ for ⊞ (increasing) and $-$ for ⊟ (decreasing)
- Prime node whose parent is linear: sign of its parent
- Other prime node: ???
  - $\hookrightarrow$ 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

Example of labeled decomposition tree



Complexity results

Complexity:

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

Our work:

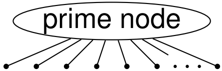
- polynomial with probability 1 asymptotically
- polynomial on average
- in a parsimonious perfect scenario for separable permutations
  - average number of reversals $\sim 1.27n$
  - average length of a reversal $\sim 1.054\sqrt{n}$

Probability distribution: always uniform

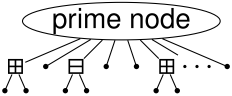
"Average shape" of decomposition trees

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

$\Rightarrow$ Asymptotically, a proportion $\frac{1}{e^2}$ of decom- -position 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

Average complexity on permutations of size n:

$$\frac{\sum_{p=0}^{n} \sharp\{\sigma \text{ with } p \text{ prime nodes}\}\; C\, 2^p n \sqrt{n \log n}}{n!}$$

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 $\leq 51 C n \sqrt{n \log n}$. In particular, **polynomial on average.**

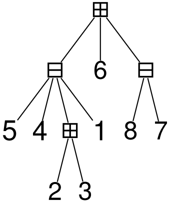
Separable (= commuting) permutations

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.

Example:
54231687 i.e.



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 ⊞ – ⊞ nor ⊟ – ⊟ edge in decomposition trees

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

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

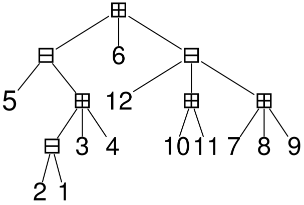
$$\boxed{\text{Reversals} \approx \text{internals nodes} - \text{the root} + \text{half of the leaves}}$$

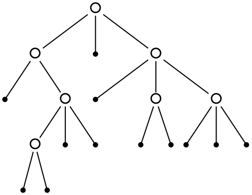
$\Rightarrow$ The shape of the tree is sufficient to study reversals

Bijection between separable perm. and Schröder trees

Decomposition trees of (unsigned) separable permutation

Schröder trees
$+$ label ⊞ or ⊟ on the root





size of σ	$\longleftrightarrow$	number of leaves
reversal of length ≥ 2	$\longleftrightarrow$	internal node except the root
reversal of length 1	$\longleftrightarrow$	some leaves (half of them)
length of a reversal	$\longleftrightarrow$	size ($= \sharp$ 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:

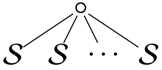
- Average number of reversals
- Average length of a reversal

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: $\mathcal{S} = \bullet \ + $ 

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: $\sim 1.27 n^{\frac{3}{2}}$

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:

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

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

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

$\hookrightarrow$ to describe models that are closer to the biological reality

$\hookrightarrow$ to identify non-random evolution (w.r.t. a *good* distribution)