Longest common separable pattern among permutations

Mathilde Bouvel, Dominique Rossin, and Stéphane Vialette

CPM, July 9 to 11, 2007

Outline

Definitions

"σ *contains* π *as a pattern*"

A permutation $\pi \in \mathfrak{S}_k$ is said to be a pattern (or to occur) whithin a permutation $\sigma \in \mathfrak{S}_n$ if σ has a subsequence that is order-isomorphic to π, *i.e.*, there exist $1 \leq x_1 \leq x_2 \leq \ldots \leq x_k \leq n$ such that for $1 \leq i, j \leq k$,

$$\sigma(x_i) < \sigma(x_j) \quad \text{iff} \quad \pi(i) < \pi(j).$$

Example

A permutation contains the pattern $(1, 2)$ iff it has an ascending subsequence of length two. Note that members need not actually be consecutive, merely ascending

Therefore, of the $3! = 6$ partitions of $\{1, 2, 3\}$, all but $(3, 2, 1)$ contain the pattern (12) , *i.e.*, an increasing subsequence of length two.

Definitions

“σ *contains* π *as a pattern*”

A permutation $\pi \in \mathfrak{S}_k$ is said to be a pattern (or to occur) whithin a permutation $\sigma \in \mathfrak{S}_n$ if σ has a subsequence that is order-isomorphic to π, *i.e.*, there exist $1 \leq x_1 \leq x_2 \leq \ldots \leq x_k \leq n$ such that for $1 \leq i, j \leq k$,

$$\sigma(x_i) < \sigma(x_j) \quad \text{iff} \quad \pi(i) < \pi(j).$$

Example

The permutation $\sigma = (7, 2, 5, 3, 8, 6, 1, 4, 9)$ contains the permutation $\pi = (1, 3, 2, 4)$ as a pattern:

$$\sigma = (7, 2, 5, 3, 8, 6, 1, 4, 9)$$
$$\pi = (\quad 1, \qquad 3, 2, \qquad 4)$$

Pattern avoiding

Pattern avoiding

A permutation σ that does not contain a permutation π as a pattern is said to avoid π.

The set of all permutations avoiding the patterns $\pi_1, \pi_2, \ldots, \pi_k$ is denoted $\mathfrak{S}(\pi_1, \pi_2, \ldots, \pi_k)$ and $\mathfrak{S}_n(\pi_1, \pi_2, \ldots, \pi_k)$ denotes the set of all permutations of length n avoiding $\pi_1, \pi_2, \ldots, \pi_k$.

Example

- $\sigma = (1, 4, 2, 5, 6, 3)$ contains the pattern $(1, 3, 4, 2)$ $(1, 5, 6, 3)$, $(1, 4, 6, 3)$, $(2, 5, 6, 3)$ and $(1, 4, 5, 3)$ are the occurrences of this pattern in σ.
- $\sigma \in \mathfrak{S}((3, 2, 1))$, *i.e.*, σ avoids the pattern $(3, 2, 1)$ as no subsequence of length 3 is decreasing.

Combinatorics *versus* Algorithmic

Context

- In combinatorics, much research focused on closed classes of permutations:
  - permutations that do not contain the pattern $(3, 1, 2)$,
  - permutations that do not contain the pattern $(2, 1, 3)$,
  - permutations that do not contain the pattern $(2, 3, 1)$ nor $(3, 1, 2)$,
  - permutations that do not contain the pattern $(1, 2, 4, 3)$ nor $(2, 1, 4, 3)$,
  - ...
- There is considerably less research on algorithmic aspects of pattern involvement.

Pattern-matching permutations

Numbers of pattern-matching permutations of $k, k+1, \ldots, n$ for various patterns $(a_1, \ldots, a_k)$ of length k.

pattern	**Sloane**	**nb. of pattern-matching permutations**
(1)	A000142	1, 2, 6, 24, 120, 720, 5040, ...
$(1,2)$	A033312	1, 5, 23, 119, 719, 5039, 40319, ...
$(1,2,3,4)$	A000000	1, 17, 207, ...
$(1,3,4,2)$	A000000	1, 17, 208, ...

Pattern-avoiding permutations

Numbers of pattern-avoiding permutations of $1, 2, \ldots, n$ for various sets of patterns.

Wilf class	Sloane	nb. of pattern-avoiding permutations
$\{(1,2,3),$ $(1,3,2),$ $(2,1,3)\}$	A000027	$1, 2, 3, 4, 5, 6, 7, 8, 9, 10, \ldots$
$\{(1,3,2),$ $(2,3,1),$ $(3,2,1)\}$	A000027	$1, 2, 3, 4, 5, 6, 7, 8, 9, 10, \ldots$
$\{(1,2,3),$ $(1,3,2),$ $(3,2,1,4)\}$	A000073	$1, 2, 4, 7, 13, 24, 44, 81, 149, \ldots$

Two patterns P_1 and P_2 belong to the same Wilf class if $|\mathfrak{S}_n(P_1)| = |\mathfrak{S}_n(P_2)|$ for all n.

Separable permutations

Separable permutation

The class of *separable permutations*, denoted Sep, is defined by $\text{Sep} = \mathfrak{S}((2,4,1,3),(3,1,4,2))$.

Other characterizations

- Permutation graphs [Bose, Buss and Lubiw, 1998],
- Interval decomposition [Bui-Xuan, Habib and Paul, 2005],
- Separating trees [Bose, Buss and Lubiw, 1998].

Separating trees

Definition

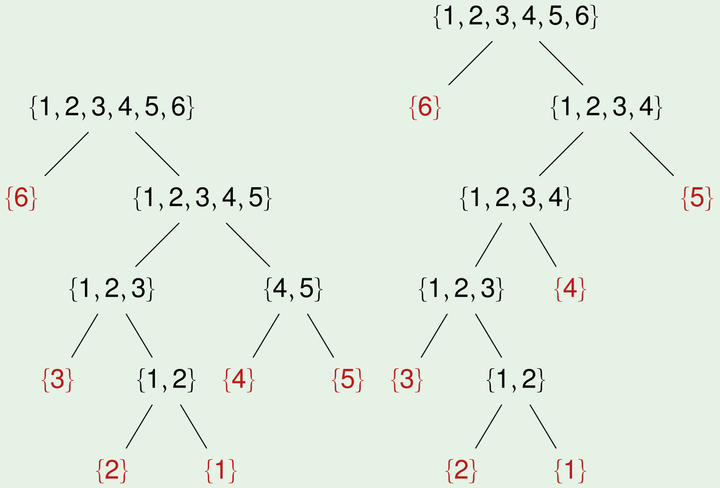
A *separating tree* for a separable permutation $\pi = (\pi_1, \pi_2, \ldots, \pi_k)$ is a binary tree T with leaves labeled $\pi_1, \pi_2, \ldots, \pi_k$ in that order, such that for each node u of T, the set $\{\pi_i, \pi_{i+1}, \ldots, \pi_{i+j}\}$ of the leaves of the subtree rooted at u forms a contiguous subrange of $[1, 2, \ldots, k]$.

Remarks

- The separating tree may not be unique.
- There exists a linear-time to decide if a permutation π is separable and if so, to construct a separating tree for π [Bose, Buss and Lubiw, 1998].

Separating trees

Example ($\pi = (6, 3, 2, 1, 4, 5)$)



Separating trees

Easy properties

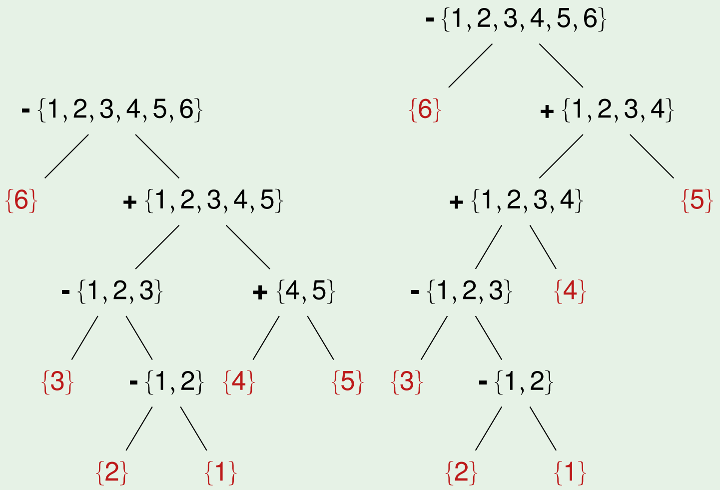
The set of leaves in the subtree rooted at u is the range of u.

If a node u has left child u_{left} and right child u_{right} then

- the range of u_{left} immediatly precedes the range of u_{right} (u is a positive node), or
- the range of u_{left} immediatly succeeds the range of u_{right} (u is a negative node).

Separating trees

Example ($\pi = (6, 3, 2, 1, 4, 5)$)



Outline

Pattern matching

PATTERN-MATCHING-FOR-PERMUTATIONS (PATTERN INVOLVEMENT)

- **Input** : A permutation $\sigma \in \mathfrak{S}_n$ and a permutation $\pi \in \mathfrak{S}_k$.
- **Question** : Is π a pattern of σ?

Algorithmic issues

- **NP**–complete [Bose, Buss and Lubiw, 1998],
- solvable in $O(n^5 \log n)$ time for fixed separable pattern, [Albert, Aldred, Atkinson and Holton, 2001]
- solvable in $O(kn^4)$ time and $O(kn^3)$ space for separable permutations [Ibarra, 1997]

Finding a common pattern

LONGEST-COMMON-PATTERN

- **Input** : A collection of permutations $\sigma_1, \sigma_2, \ldots, \sigma_n$.
- **Solution** : A permutation π that occurs as a pattern in each input permutation σ_i, $1 \leq i \leq n$.
- **Measure** : The size of the common pattern, *i.e.*, $|\pi|$.

Algorithmic issues

- **NP**–complete [Bose, Buss and Lubiw, 1998],
- Solvable in polynomial-time provided that one of the two permutation is separable [Bouvel and Rossin, 2006].

Finding a common $\mathcal{C}$-pattern

LONGEST-$\mathcal{C}$-COMMON-PATTERN

- **Input** : A collection of permutations $\sigma_1, \sigma_2, \ldots, \sigma_n$.
- **Solution** : A permutation $\pi \in \mathcal{C}$ that occurs as a pattern in each input permutation σ_i, $1 \leq i \leq n$.
- **Measure** : The size of the common pattern, *i.e.*, $|\pi|$.

Algorithmic issues

- Complexity: **NP**–complete but widely unexplored.
- We focus here on $\mathcal{C} = $ Sep.

Outline

A polynomial-time algorithm

Finding a longest common pattern

- We focus here on separable permutations.
    - We do not require the input permutations to be separable.
    - Tree-like structure.
- Beside separable permutations, only a few restricted cases were considered (*e.g.*, finding a longest increasing or decreassing subpermutation).

A polynomial-time algorithm

Definition

Let $\pi = (\pi_i, \pi_2, \ldots, \pi_k)$ and $\pi'(\pi'_i, \pi'_2, \ldots, \pi'_{k'})$ be two permutations. The positive and negative concatenations of π and π' are defined by:

$$\pi \oplus \pi' = (\pi_1, \pi_2, \ldots, \pi_k, \pi'_1 + k, \pi'_2 + k, \ldots, \pi'_{k'} + k)$$
$$\pi \ominus \pi' = (\pi_1 + k', \pi_2 + k', \ldots, \pi_k + k', \pi'_1, \pi'_2, \ldots, \pi'_{k'})$$

Properties

- If both π and π' are separable, then so are $\pi \oplus \pi'$ and $\pi \ominus \pi'$.
- Any separable permutation π of length at least 2 might be written as $\pi = \pi_1 \oplus \pi_2$ or $\pi = \pi_1 \ominus \pi_2$ for some non empty separable permutations π_1 and π_2.

A polynomial-time algorithm

Definition

Let $\pi \in \mathfrak{S}_k$ and $\sigma \in \mathfrak{S}_n$ be such that π has the occurrence $\sigma_{i_1}, \sigma_{i_2}, \ldots, \sigma_{i_k}$ in σ. Let I and V be two subintervals of $[1 : n]$ such that

- $\{i_1, i_2, \ldots, i_k\} \subseteq I$ and
- $\{\sigma_{i_1}, \sigma_{i_2}, \ldots, \sigma_{i_k}\} \subseteq V$.

We say that π is a pattern of σ using the intervals I of indices and V of values in σ.

Decomposition theorem I

Theorem

Let π_1 *be a common separable pattern of* $\sigma_1, \sigma_2, \ldots, \sigma_n$ *that uses the intervals* $[i_q..h_q - 1]$ *of indices and* $[a_q..c_q - 1]$ *(resp.* $[c_q..b_q]$*) of values in* σ_q, $1 \leq q \leq n$.

Let π_2 *be a common separable pattern of* $\sigma_1, \sigma_2, \ldots, \sigma^n$ *that uses the intervals* $[h_q..j_q]$ *of indices and* $[c_q..b_q]$ *(resp.* $[a_q..c_q - 1]$*) of values in* σ_q, $1 \leq q \leq n$.

Then $\pi = \pi_1 \oplus \pi_2$ *(resp.* $\pi = \pi_1 \ominus \pi_2$*) is a common separable pattern of* $\sigma_1, \sigma_2, \ldots, \sigma^n$ *that uses the intervals* $[i_q..j_q]$ *of indices and* $[a_q..b_q]$ *of values in* σ_q, $1 \leq q \leq n$.

Decomposition theorem II

Theorem

Let π be a common separable pattern of maximal length of $\sigma_1, \sigma_2, \ldots, \sigma_n$ among those using the intervals $[i_q..j_q]$ of indices and $[a_q..b_q]$ of values in σ_q, $1 \leq q \leq n$.
If $\pi = \pi_1 \oplus \pi_2$ (resp. $\pi = \pi_1 \ominus \pi_2$), π_1 and π_2 being two non-empty separable patterns, then there exist indices $(h_q)_{1 \leq q \leq n}$ and values $(c_q)_{1 \leq q \leq n}$, $i_q < h_q \leq j_q$, $a_q < c_q \leq b_q$, $1 \leq q \leq n$, such that

- *π_1 is a common separable pattern of maximal length of $\sigma_1, \sigma_2, \ldots, \sigma_n$ among those using the intervals $[i_q..h_q - 1]$ of indices and $[a_q..c_q - 1]$ (resp. $[c_q..b_q]$) of values in σ_q, $1 \leq q \leq n$, and*
- *π_2 is a common separable pattern of maximal length of $\sigma_1, \sigma_2, \ldots, \sigma_n$ among those using the intervals $[h_q..j_q]$ of indices and $[c_q..b_q]$ (resp. $[a_q..c_q - 1]$) of values in σ_q, $1 \leq q \leq n$.*

A polynomial-time algorithm

sketch

- Dynamic programming algorithm.
- Merging and splitting separables patterns.
- Dynamic programming table M of size $4n$:

$$M[\overbrace{i_1, j_1, a_1, b_1}^{\sigma_1}, \overbrace{i_2, j_2, a_2, b_2}^{\sigma_2}, \ldots, \overbrace{i_n, j_n, a_n, b_n}^{\sigma_n}]$$

contains a common maximum size separable pattern using intervals $[i_q : j_q]$ of indices and $[a_q : b_q]$ of values in σ_q, $1 \leq q \leq n$.

A polynomial-time algorithm

Theorem

The LONGEST-SEP-COMMON-PATTERN *problem is solvable in* $O(m^{6n+1})$ *time and* $O(m^{4n+1})$ *space, where* n *is the number of permutations and* m *is the maximum size of a permutation.*

Remarks

- What about the computational complexity (**P** *versus* **NP**) of the LONGEST-SEP-COMMON-PATTERN problem ?
- Is the LONGEST-SEP-COMMON-PATTERN problem an issue for approximating the LONGEST-COMMON-PATTERN problem ?

Hardness result

Theorem

The LONGEST-$\mathcal{C}$-COMMON-PATTERN *problem, in its natural decision version, is* **NP***–complete for* $\mathcal{C} = Sep$*, even if each input permutation is itself separable.*

Remarks

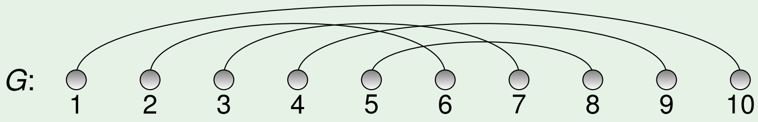
- The number of input permutations is not fixed.
- Reduction from the INDEPENDENT-SET problem: *find a subset of vertices in a graph no two of which are adjacent*.
- Matching diagrams.

From permutations to matching diagrams

Matching diagram

- Vertex labeled graph where vertices are labeled by distinct numbers in $\{1, 2, \ldots, n\}$.
- Each vertex $i \in \{1, 2, \ldots, n\}$ is connected by an edge to exactly one vertex $j \in \{n+1, n+2, \ldots, 2n\}$.
- Each vertex $j \in \{n+1, n+2, \ldots, 2n\}$ is connected by an edge to exactly one vertex $j \in \{1, 2, \ldots, n\}$.

Example



$\pi = (2, \quad 3, \quad 5, \quad 4, \quad 1)$

From permutations to matching diagrams

Definition

A matching diagram G' is said to occur in a matching diagram G if on can obtain G' from G by a sequence of edge deletion.

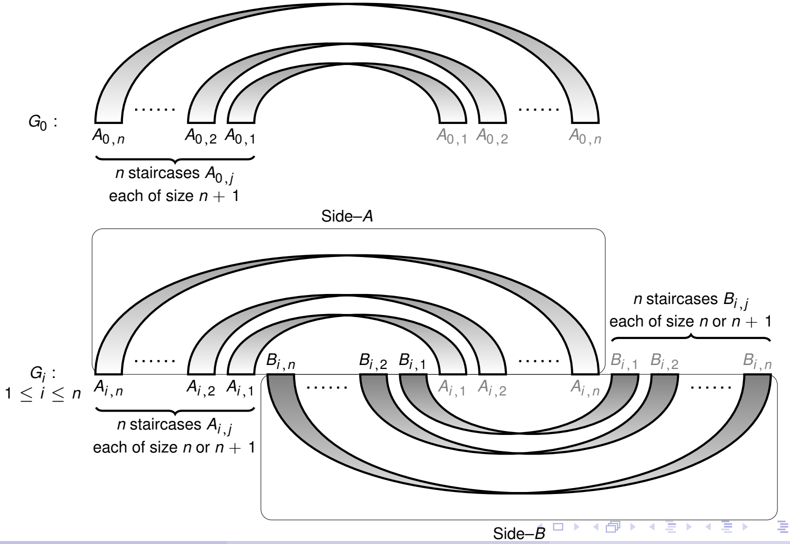
Theorem

The LONGEST-$\mathcal{C}$-COMMON-PATTERN *problem for* $\mathcal{C} = Sep$ *is computationally equivalent to the following problem: "**Given a set of matching diagrams, find a largest matching diagram that occurs in each input matching diagram**".*

Remarks

- $\{\sqsubset, \between\}$-linear graphs.
- $\{\sqsubset\}$-matching diagram *versus* increasing subsequence.
- $\{\between\}$-matching diagram *versus* decreasing subsequence.

Key ideas


G_0 :
$A_{0,n}$ …… $A_{0,2}$ $A_{0,1}$ $A_{0,1}$ $A_{0,2}$ …… $A_{0,n}$
n staircases $A_{0,j}$
each of size $n+1$
Side–A
n staircases $B_{i,j}$
each of size n or $n+1$
G_i :
$1 \leq i \leq n$
$B_{i,n}$ …… $B_{i,2}$ $B_{i,1}$ $B_{i,1}$ $B_{i,2}$ …… $B_{i,n}$
$A_{i,n}$ …… $A_{i,2}$ $A_{i,1}$ $A_{i,1}$ $A_{i,2}$ …… $A_{i,n}$
n staircases $A_{i,j}$
each of size n or $n+1$
Side–B

Stanley-Wilf Conjecture

Definition

Let $F(n, \sigma)$ denote the number of permutations on the symmetric group $\mathfrak{S}_n$ which avoid σ as a subpattern.

Stanley-Wilf Conjecture

Stanley and Wilf conjectured that, for every permutation pattern σ, there is a constant $c(\sigma) < \infty$ such that for all n, $F(n, \sigma) \leq (c(\sigma))^n$.

- A related conjecture stated that for every sigma, the limit

$$\lim_{n \to \infty} (F(n, \sigma))^{1/n}$$

  exists and is finite.
- Arratia (1999) showed that these two conjectures are equivalent.
- The conjecture was proved by Marcus and Tardos (2004).

Inapproximabilty result

Theorem

For any $\varepsilon > 0$ *and any pattern avoiding permutation class* $\mathcal{C}$*, there exists a sequence* $(\sigma_n)_{n \in \mathbb{N}}$, $\sigma_n \in \mathfrak{S}_n$*, such that* $|\pi_n| = o(n^{0.5+\varepsilon})$*, where* π_n *is a longest permutation of class* $\mathcal{C}$ *involved as a pattern in* σ_n*.*

Theorem

The LONGEST-COMMON-PATTERN *problem cannot be approximated within a tighter ratio that* $\sqrt{\mathbf{opt}}$ *by solving the* LONGEST-$\mathcal{C}$-COMMON-PATTERN *problem, where* $\mathcal{C}$ *if a pattern-avoiding permutation class and* $\mathbf{opt}$ *is the size on the optimal solution for the* LONGEST-COMMON-PATTERN *problem.*

Outline

Open problems [1]

Parameterized complexity

Is the PATTERN-MATCHING-FOR-PERMUTATIONS problem **fixed-parameter tractable** ?

Equivalently, does there exist an algorithm solving the PATTERN-MATCHING-FOR-PERMUTATIONS problem that runs in time $n^c\ f(k)$ for some constant c and some arbitrary function $f(\cdot)$?

Open problems [2]

Finitely based classes

Is the LONGEST-$\mathcal{C}$-COMMON-PATTERN problem polynomial-time solvable for any *finitely based* permutation class $\mathcal{C}$ (in case of a fixed number of input permutations).

Remarks

- For a permutation class $\mathcal{C}$, deciding whether a permutation belongs to $\mathcal{C}$ might be an **NP**–complete problem (*e.g.*, 4-stack sortable permutations).
- We are not aware of any finitely based pattern-avoiding permutation class for which the recognition problem is **NP**–hard.