The computational landscape of permutation patterns

Marie-Louise Bruner

Joint work with Martin Lackner

Permutation Patterns 2013, July 4th
Types of permutation patterns

- classical patterns
- vincular patterns
- bivincular patterns
- consecutive patterns
- mesh patterns
- boxed mesh patterns
Permutation Pattern Matching Problems

Every type of permutation pattern naturally defines a corresponding computational problem. Let $C$ denote any type of permutation pattern, i.e., let $C \in \{\text{classical, vincular, bivincular, mesh, boxed mesh, consecutive}\}$.

\[ C \text{ Permutation Pattern Matching (C PPM)} \]

**Instance:** A permutation $T$ (the text) and a $C$ pattern $P$

**Question:** Does the $C$ pattern $P$ occur in $T$?
## Definitions and Examples

<table>
<thead>
<tr>
<th>Pattern</th>
<th>Classical</th>
<th>Vincular</th>
<th>Bivincular</th>
</tr>
</thead>
<tbody>
<tr>
<td>$P = 132$</td>
<td><img src="image1" alt="Classical Pattern" /></td>
<td><img src="image2" alt="Vincular Pattern" /></td>
<td><img src="image3" alt="Bivincular Pattern" /></td>
</tr>
</tbody>
</table>

**Text**

- **Classical**
- **Vincular**
- **Bivincular**
## Definitions and Examples 2

<table>
<thead>
<tr>
<th>Pattern</th>
<th>Mesh</th>
<th>Boxed mesh</th>
<th>Consecutive</th>
</tr>
</thead>
<tbody>
<tr>
<td>$P = (\pi, R) = 132$</td>
<td>$P = 132$</td>
<td>$P = 132$</td>
<td></td>
</tr>
</tbody>
</table>

### Text

![Mesh Diagram](image1)

![Boxed Mesh Diagram](image2)

![Consecutive Pattern Diagram](image3)
Hierarchy of pattern types

mesh

boxed mesh

bivincular

vincular

consecutive

classical
Theorem [Bose, Buss, and Lubiw, 1993]

Classical permutation Pattern Matching is NP-complete.
Theorem [Bose, Buss, and Lubiw, 1993]

**Classical permutation Pattern Matching** is NP-complete.

The following problems are also NP-hard:

- **Vincular PPM**
- **Bivincular PPM**
- **Mesh PPM**
Theorem [Bose, Buss, and Lubiw, 1993]

**Classical permutation Pattern Matching** is NP-complete.

⇒ The following problems are also NP-hard:

- **Vincular PPM**
- **Bivincular PPM**
- **Mesh PPM**

How about **Boxed Mesh PPM and Consecutive PPM**?
Boxed Mesh PPM is in P

It is easy to see that Boxed Mesh PPM can be solved in $O(n^3)$-time.
Boxed Mesh PPM is in P

It is easy to see that Boxed Mesh PPM can be solved in $O(n^3)$-time.
It is easy to see that **Boxed Mesh PPM** can be solved in $O(n^3)$-time.
**Boxed Mesh PPM is in P**

It is easy to see that **Boxed Mesh PPM** can be solved in $O(n^3)$-time.

There are $O(n^2)$ pairs $(i,j)$ that have to be checked.
Consecutive PPM is in $P$

Consecutive PPM can be solved in linear time in the length of the text:


Consecutive PPM can be solved in linear time in the length of the text:


It is very easy to see that Consecutive PPM can be solved in $O((n - k) \cdot k)$-time.
Permutation patterns

Polynomial time

NP-complete

Marie-Louise Bruner

The computational landscape of PPM

June 10th, 2013
A parameterized point of view

Idea: Which *parameter* a PPM instance makes this problem computationally hard?
A parameterized point of view

Idea: Which *parameter* a PPM instance makes this problem computationally hard?

Parameterized Problem: \( L \subseteq \Sigma^* \times \mathbb{N} \)
A parameterized point of view

Idea: Which parameter a PPM instance makes this problem computationally hard?

Parameterized Problem: \( L \subseteq \Sigma^* \times \mathbb{N} \)

Fixed-parameter tractability

A parameterized problem \( L \) is \textit{fixed-parameter tractable} if there is a computable function \( f \) and an integer \( c \) such that there is an algorithm solving \( L \) in time \( O(f(p) \cdot |l|^c) \).

Marie-Louise Bruner

The computational landscape of PPM

June 10th, 2013
Proving intractability

Definition

The class $W[1]$ is defined as the class of all problems that are fpt-reducible to the following problem.

**Clique**

*Instance:* A graph $G = (V, E)$ and a positive integer $k$.

*Parameter:* $k$

*Question:* Is there a subset of vertices $S \subseteq V$ of size $k$ such that $S$ forms a clique, i.e., the induced subgraph $G[S]$ is complete?
Proving intractability

Definition

The class $W[1]$ is defined as the class of all problems that are fpt-reducible to the following problem.

**Clique**

*Instance:* A graph $G = (V, E)$ and a positive integer $k$.

*Parameter:* $k$

*Question:* Is there a subset of vertices $S \subseteq V$ of size $k$ such that $S$ forms a clique, i.e., the induced subgraph $G[S]$ is complete?

If we can prove $W[1]$-hardness, this implies, under standard complexity theoretic assumptions, that no fpt-algorithm exists.
Fpt-reductions

Definition

Let $L_1, L_2 \subseteq \Sigma^* \times \mathbb{N}$ be two parameterized problems. An \textit{fpt-reduction} from $L_1$ to $L_2$ is a mapping $R : \Sigma^* \times \mathbb{N} \rightarrow \Sigma^* \times \mathbb{N}$ such that

- $(I, k) \in L_1$ iff $R(I, k) \in L_2$.
- $R$ is computable by an fpt-algorithm.
- There is a computable function $g$ such that for $R(I, k) = (I', k')$, $k' \leq g(k)$ holds.
Segregated PPM

**Segregated Permutation Pattern Matching (SPPM)**

**Instance:** A permutation $T$ (the text) of length $n$, a permutation $P$ (the pattern) of length $k \leq n$ and two positive integers $p \in [k]$, $t \in [n]$.

**Parameter:** $k$

**Question:** Is there a matching $\mu$ of $P$ into $T$ such that $\mu(i) \leq t$ iff $i \leq p$?

Consider the pattern $P = 132$ and the text $T = 53142$. As shown by the matching $\mu(2) = 3$, $\mu(1) = 1$ and $\mu(3) = 4$, the instance $(P, T, 2, 3)$ is a yes-instance of the SPPM problem. However, $(P, T, 2, 4)$ is a NO-instance, since no matching of $P$ into $T$ can be found where $\mu(3) > 4$. 

Marie-Louise Bruner
**Segregated PPM**

**Segregated Permutation Pattern Matching (SPPM)**

**Instance:** A permutation $T$ (the text) of length $n$, a permutation $P$ (the pattern) of length $k \leq n$ and two positive integers $p \in [k]$, $t \in [n]$.

**Parameter:** $k$

**Question:** Is there a matching $\mu$ of $P$ into $T$ such that $\mu(i) \leq t$ iff $i \leq p$?

Consider the pattern $P = 132$ and the text $T = 53142$. As shown by the matching $\mu(2) = 3$, $\mu(1) = 1$ and $\mu(3) = 4$, the instance $(P, T, 2, 3)$ is a yes-instance of the SPPM problem. However, $(P, T, 2, 4)$ is a NO-instance, since no matching of $P$ into $T$ can be found where $\mu(3) > 4$. 
Segregated PPM is W[1]-hard.

Proof idea: reduction from Clique.
Mesh PPM is \( W[1] \)-hard.

Proof idea: Let \((P, T, p, t)\) be a Segregated PPM instance. We define \(p' = p + 0.5\) and \(t' = t + 0.5\).
Mesh PPM is $W[1]$-hard.

Proof idea: Let $(P, T, p, t)$ be a Segregated PPM instance. We define $p' = p + 0.5$ and $t' = t + 0.5$. A Mesh PPM instance $(P', T')$ is constructed as follows:
Bivincular and Vincular PPM are also $W[1]$-hard.

*Proof idea:* reduction from Segregated PPM.
Permutation patterns

Marie-Louise Bruner

The computational landscape of PPM

June 10th, 2013 19
Further directions

- Is PPM fixed-parameter tractable with respect to the length of the pattern?
- Other parameters than the length of the pattern
- Find polynomial fragments
- Partially ordered patterns: a POPPM instance can be reduced to at most $k!$ many PPM instances
- Marked mesh and decorated patterns
- Patterns in words or in partitions