Kernelization lower bound for Permutation Pattern Matching

Ivan Bliznets† Marek Cygan‡ Paweł Komosa§ Lukáš Mach¶

June 5, 2014

Abstract

A permutation $\pi$ contains a permutation $\sigma$ as a pattern if it contains a subsequence of length $|\sigma|$ whose elements are in the same relative order as in the permutation $\sigma$. This notion plays a major role in enumerative combinatorics. We prove that the problem does not have a polynomial kernel (under the widely believed complexity assumption $\text{NP} \not\subseteq \text{co-NP}/\text{poly}$) by introducing a new polynomial reduction from the clique problem to permutation pattern matching.

1 Introduction

Counting permutations of size $n$ avoiding a fixed pattern is an established and active area of enumerative combinatorics. Knuth [11] has shown that the number of permutations avoiding $(2, 3, 1)$ is the $n^{th}$ Catalan number. Various choices of prohibited patterns have been studied among others by Lovász [12], Rotem [14], and Simion and Schmid [15]. This culminated in the Stanley-Wilf conjecture stating that for every fixed prohibited pattern, the number of permutations of length $n$ avoiding it can be bounded by $c^n$ for some constant $c$. Klazar [10] reduced

---

*Partially supported by Warsaw Center of Mathematics and Computer Science.
†St. Petersburg Academic University of the Russian Academy of Sciences, Russia. E-mail: ivanbliznets@tut.by.
‡Institute of Informatics, University of Warsaw, Poland. E-mail: cygan@mimuw.edu.pl. Partially supported by NCN grant DEC-2012/05/D/ST6/03214.
§Institute of Informatics, University of Warsaw, Poland. E-mail: kompaw01@gmail.com.
¶DIMAP and Department of Computer Science, University of Warwick, United Kingdom. E-mail: lukas.mach@gmail.com. Partially supported by the European Research Council under the European Union’s Seventh Framework Programme (FP7/2007-2013)/ERC grant agreement no. 259385.
the question to the Füredi-Hajnal conjecture, which was ultimately proved by Marcus and Tardos in 2004 [13].

Herbert Wilf also posed the algorithmic question of whether finding a given pattern (of length \(\ell\)) in a given permutation (of length \(n\)) can be done in subexponential time. However, the problem was shown to be \(\text{NP}\)-hard in [6]. Ahal and Rabinovich have obtained an \(O(n^{0.47\ell + o(\ell)})\) time algorithm [1]. Fast algorithms have been found for certain restricted versions of the problem.

Pattern matching has also received interest in the context of parameterized complexity. Several groups of researchers have obtained \(W[1]\)-hardness results for generalizations of the problem [4,9]. In [5] it was shown that the problem is in \(\text{FPT}\) when parameterized by the number of runs (maximal monotonous subsequences) in the target permutation. Here, the authors raise the issue of whether their problem has a polynomial size kernel as an open problem. The central question of whether the problem is in \(\text{FPT}\) when parameterized by \(\ell\) has been resolved by Guillemot and Marx [9], who obtained an algorithm with runtime of \(2^{O(\ell \log \ell)} \cdot n\). This implies the existence of a kernel for the problem. Obtaining kernel size lower bounds was posed as an open question during a plenary talk at Permutation Patterns 2013 by Stéphane Vialette.

**Contribution:** We prove that the pattern problem under the standard parameterization by \(\ell\) does not have a polynomial size kernel, assuming \(\text{NP} \not\subseteq \text{co-NP/poly}\).

### 2 Preliminaries

The set \(\{i, i + 1, \ldots, j - 1, j\}\) is denoted by \([i, j]\). We set \([n] := [1, n]\). Permutation \(\pi\) is a bijection from \([n]\) to \([n]\). The value \(\pi(i)\) is called the entry of \(\pi\) at position \(i\). We use \(|\pi|\) to denote the size of the domain of \(\pi\). Two common representations of a permutation \(\pi\) are used: using a vector \((\pi(1), \pi(2), \ldots, \pi(n))\) and using a permutation matrix. The latter is a \(|\pi| \times |\pi|\) binary matrix with 1-entries precisely on coordinates \((\pi(i), i)\). A vector obtained from the vector representation by omitting some entries is a subsequence of the permutation. A subsequence of the permutation is a consecutive subsequence if it contains precisely the entries with indexes from \([i, j]\) for some \(i, j \in \mathbb{N}\). We use \(\pi[i, j]\) to denote the set of entries \(\{\pi(i), \pi(i + 1), \ldots, \pi(j)\}\). A maximal consecutive subsequence of a permutation is called a run if the entries form a monotonous sequence. For example, \((4, 5, 3, 1, 2)\) contains a (decreasing) run of length 3.

The key notion of a permutation pattern is introduced below:

**Definition 1.** A permutation \(\sigma\) on the set \([l]\) is a pattern of a permutation \(\pi\) on
the set $[n]$ if there exists an increasing function $\phi : [l] \to [n]$ such that

$$\forall x, y \in [l] : \sigma(x) < \sigma(y) \text{ if and only if } \pi(\phi(x)) < \pi(\phi(y)).$$

We say that the function $\phi$ certifies the pattern.

A parameterized problem is a language $Q \subseteq \Sigma^* \times \mathbb{N}$, where $\Sigma$ is a fixed alphabet. The value $k$ of the instance $(x, k) \in Q$ is its parameter. **Permutation Pattern** is the following parameterized algorithmic problem:

| **Input:** | a permutation $\sigma$ on [l], a permutation $\pi$ on [n]. |
|------------|------------------------------------------------------------|
| **Parameter:** | $\ell$. |
| **Question:** | is $\sigma$ a pattern of $\pi$? |

A problem $Q$ is in **FPT** if there is an algorithm deciding $(x, k) \in Q$ in time $f(k)|x|^{O(1)}$, where $f$ is a computable function.

**Definition 2.** Kernelization algorithm for a parameterized problem $Q$ is an algorithm that given an instance $(x, k) \in \Sigma^* \times \mathbb{N}$ produces in $p(|x| + k)$ steps an instance $(x', k')$ such that

1. $(x, k) \in Q \iff (x', k') \in Q$
2. $|x'|, k' \leq f(k),$

where $p(\cdot)$ is a polynomial and $f(\cdot)$ a computable function.

If there is a kernelization algorithm for $Q$, we say that $Q$ has a kernel. If the function $f(\cdot)$ in the above definition can be bounded by a polynomial, we say that $Q$ has a polynomial kernel.

We utilize the standard machinery of Bodlaender et al. [3], which builds on [2, 7], to derive kernelization lower-bounds under the widely believed complexity assumption $\text{NP} \not\subseteq \text{co-NP}/\text{poly}$. The failure of this assumption implies that the polynomial hierarchy collapses to the third level. Some basic definitions are necessary:

**Definition 3** (Bodlaender et al. [3]). An equivalence relation $R$ on $\Sigma^*$ is called a polynomial equivalence relation if the following two conditions hold:

1. There is an algorithm that given two strings $x, y \in \Sigma^*$ decides whether $x$ and $y$ belong to the same equivalence class in $(|x| + |y|)^{O(1)}$ time.
2. For any finite set $S \in \Sigma^*$ the equivalence relation $R$ partitions the elements of $S$ into at most $(\max_{x \in S} |x|)^{O(1)}$ equivalence classes.
An example of such relation is the grouping of instances of the same size.

**Definition 4** (Bodlaender et al. [3]). Let $L \subseteq \Sigma^*$ be a set and let $Q \subseteq \Sigma^* \times \mathbb{N}$ be a parameterized problem. We say that $L$ cross-composes into $Q$ if there is a polynomial equivalence relation $R$ and an algorithm which, given $t$ strings $x_1, x_2, \ldots, x_t$ belonging to the same equivalence class of $R$, computes an instance $(x^*, k^*) \in \Sigma^* \times \mathbb{N}$ in time polynomial in $\sum_{i=1}^{t} |x_i|$ such that:

1. $(x^*, k^*) \in Q \iff x_i \in L$ for some $1 \leq i \leq t$,
2. $k^*$ is bounded by a polynomial in $\max_{i=1}^{t} |x_i| + \log t$.

**Theorem 5** (Bodlaender et al. [3]). Let $L \subseteq \Sigma^*$ be an NP-hard language. If $L$ cross-composes into the parameterized problem $Q$ and $Q$ has a polynomial kernel then $NP \subseteq \text{co-NP/poly}$.

### 3 Kernelization lower bound for PERMUTATION PATTERN

We show that the PERMUTATION PATTERN problem likely does not have a polynomial kernel:

**Theorem 6.** Unless $NP \subseteq \text{co-NP/poly}$, the PERMUTATION PATTERN problem does not admit a polynomial kernel.

We prove Theorem 6 using Theorem 5. However, this requires a polynomial time reduction that allows cross-composition without significantly increasing the parameter value. The reductions previously described in the literature [4, 6] do not have such property. We introduce a reduction from the well known CLIQUE problem to the PERMUTATION PATTERN problem. We then apply the cross-composition framework.

Let us first define encoding $\pi_z(G)$ taking a graph $G$ and $z \in \mathbb{N}$ and producing a permutation, as illustrated on Figure 1. The key property of the encoding is that for any $G, H$ we have $G \subseteq H$ if and only if $\pi_z(G)$ is a pattern of $\pi_z(H)$ for a particular choice of $z$. This allows us to express the clique problem in terms of permutation pattern matching. The encoding permutation itself consists of two different types of entries: encoding entries and separating entries. The former ones encode the edges of $G$, while the latter separate encoding entries corresponding to different vertices. Separating entries form pairs of decreasing runs inserted for each vertex of $G$. 


Figure 1: Permutation matrix representation of the encoding permutation $\pi_3(G)$ of a graph $G = (\{1, 2, 3, 4\}, \{\{1, 2\}, \{2, 3\}, \{2, 4\}, \{3, 4\}\})$. White positions correspond to 0-entries of the permutation matrix, non-white positions are 1-entries. Columns are indexed from left to right, rows from bottom to top. Therefore, the $(1, 1)$ entry of the matrix is in the bottom-left corner of the grid. Separating runs are colored in gray, encoding entries in red. Vertical red lines denote indexes $L(2)$ and $R(2)$. Horizontal red lines represent values attained at positions $L(4)$ and $R(4)$. Notice that these four red lines induce a rectangle containing a single 1-entry encoding the edge $\{2, 4\}$.

We start constructing $\pi_z(G)$ by imposing a total ordering on $V(G)$ placing vertices from the same connected component of $G$ consecutively. Thus, we can assume that $V(G) = [n]$ and set

$$
N^+_G(v) := \{ u : u > v \land \{u, v\} \in E(G) \},
$$

$$
N^-_G(v) := \{ u : u < v \land \{u, v\} \in E(G) \},
$$

$$
deg^+_G(v) := |N^+_G(v)|,
$$

$$
deg^-_G(v) := |N^-_G(v)|.
$$

Let us first introduce a notation for positions and values of the resulting permutation’s entries that are of special importance. As stated above, the permutation $\pi_z(G)$ includes a pair of decreasing runs of length $z$ for each vertex $v$ of $G$. These two runs surround the positions encoding the neighbourhood $N^+_G(v)$ of $v$, one from left and the other from right. We use $p_L(v)$ and $p_R(v)$ as a shorthand for the positions on which the left and right decreasing run of $v$ starts, respectively. The first
position of the segment encoding $N^+_G(v)$ is denoted by $p_M(v)$. Specifically, we set
$p_L(1) := 1, p_M(1) := z + 1$, and $p_R(1) := z + 1 + \deg^+_G(1)$. For $v \geq 2$, we have:
\[
\begin{align*}
p_L(v) &:= p_R(v - 1) + z, \\
p_M(v) &:= p_L(v) + z, \\
p_R(v) &:= p_M(v) + \deg^+_G(v).
\end{align*}
\]
For convenience, we also introduce notation for the values used by the separating runs. The left separating run of $v$ starts at position $p_L(v)$ with value $q_L(v)$. The right separating run starts at $p_R(v)$ with value $q_R(v)$. Finally, $q_M(v)$ is the least value used to determine $N^-_G(v)$. Specifically, the encoding entries of vertices from $N^-_G(v)$ use the values $[q_M(v), q_M(v) + \deg^-_G(v) - 1]$ to determine their connection to $v$. (When $N^-_G(v)$ is empty – which corresponds to the situation that no vertex preceding $v$ is connected to it – we let $q_M(v) := q_R(v) + 1$. Note that this is always the case with the first vertex of $G$.) We set $q_L(1) := 2z, q_M(1) := z + 1$, and $q_R(1) := z$. For $v \geq 2$ let
\[
\begin{align*}
q_L(v) &:= q_M(v) + z + \deg^-_G(v) - 1, \\
q_M(v) &:= q_R(v) + 1, \\
q_R(v) &:= q_L(v - 1) + z.
\end{align*}
\]
We now define the values of $\pi = \pi_z(G)$. For each $v$, we introduce a decreasing run of length $z$ starting at position $p_L(v)$:
\[
\begin{align*}
\pi(p_L(v)) &:= q_L(v), \\
\pi(p_L(v) + 1) &:= q_L(v) - 1, \\
\pi(p_L(v) + 2) &:= q_L(v) - 2, \\
&\quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \qv
The remaining values are used to encode the graph $G$. The neighbourhood $N^+_G(v)$ is encoded by an increasing run on positions $p_M(v), p_M(v) + 1, \ldots, p_M(v) + |N^+_G(v)| - 1$. We fix a vertex $v \in V(G)$ and iterate through the neighbours \( \{u_1, u_2, \ldots, u_k\} \in N^+_G(v) \). Assume $u_1 < u_2 < \ldots < u_k$. For $i \in [k]$, we set:

$$\pi(p_M(v) + i - 1) := q_M(u_i) + j(i),$$

where $j(i) = |\{w : w < v \land \{w, u_i\} \in E(G)\}|$. The term $j(i)$ ensures that no value in $\pi$ is repeated and the resulting function is indeed a permutation.

The above procedure is repeated for each $v \in V(G)$. After iterating through all vertices, the entries of $\pi$ are assigned a value and no value is used twice. Thus, we obtain a permutation. This finishes the construction of $\pi_z(G)$. For the purpose of the proof of the lemma below, we define the following notation:

$$C(v) := [p_M(v), p_R(v) - 1],$$

$$L(v) := p_L(v) + \lfloor \frac{z}{2} \rfloor,$$

$$R(v) := p_R(v) + \lfloor \frac{z}{2} \rfloor.$$

Therefore, $C(v)$ is the set of entries of $\pi$ encoding the vertex $v$, $L(v)$ denotes the middle entry of the left separating run of $v$, and $R(v)$ is the middle entry of the right separating run of $v$.

The following claim immediately implies \textbf{NP}-hardness of \textsc{Permutation Pattern}:

\textbf{Lemma 7.} For every graph $G$ without isolated vertices, $K_l$ is a subgraph of $G$ if and only if $\pi_z(K_l)$ is a pattern of $\pi_z(G)$, for $z = 4n' + 4$, where $n'$ is the number of vertices in the largest connected component of $G$.

\textbf{Proof.} We let $\sigma := \pi_z(K_l)$ and $\pi := \pi_z(G)$.

Clearly, if $G$ contains a clique of size $l$ as a subgraph, then $\pi$ contains the pattern $\sigma$ by construction.

For the other direction, assume there is a function $\phi : [\sigma] \to [\pi]$ certifying the pattern. We start by noting there are no decreasing subsequences of length $\frac{z}{4}$ in $\pi$ avoiding all pairs of separating runs. This is because such sequence contains at most one entry from $C(i)$ for each $i \in [n]$, where $n := |V(G)|$. At the same time, it cannot simultaneously contain an entry from $C(i)$ and $C(j)$ for $i,j$ chosen from different connected components. This is because the construction of the encoding permutation places vertices from the same component consecutively and the entries encoding a component placed earlier in the ordering have strictly smaller values than those from a later component. Together, this bounds the length of such subsequence by $n' < \frac{1}{4}z$.

Furthermore, \textit{any} decreasing subsequence can contain entries from at most one pair of separating runs. This is because once the sequence hits such run,
the subsequent entries included in the decreasing subsequence can only be from this one pair of separating runs. A decreasing subsequence of length $z$ therefore contains at least $\frac{3}{4}z$ of consecutive entries of a single pair of separating runs.

Consider now a vertex $v \in K_l$. Since $\phi$ maps the subsequence of $\sigma$ formed by the pair of separating runs of $v$ to a decreasing subsequence of $\pi$ of the same length, we know that the middle entry $L(v)$ of the left separating run of $v$ in $\sigma$ needs to be mapped to the left separating run of some vertex $u \in G$. Additionally, the middle entry $R(v)$ of the right separating run of $v$ in $\sigma$ needs to be mapped by $\phi$ somewhere in the right separating run of the same vertex $u$. This establishes a mapping from $K_l$ to $G$ denoted by $f_\phi$. We claim $f_\phi$ to be a graph homomorphism.

Fix any pair of vertices $v_1$, $v_2$ of $K_l$ such that $v_1 < v_2$. We show that $f_\phi(v_1), f_\phi(v_2)$ are connected by an edge in $G$. Note that $\sigma[L(v_1), R(v_1)]$ contains precisely one number $p$ with $\sigma(R(v_2)) \leq p \leq \sigma(L(v_2))$. Thus, there needs to be an entry of $\pi$ with an index between $\phi(L(v_1))$ and $\phi(R(v_1))$ and value between $\phi(R(v_2))$ and $\phi(L(v_2))$. Recall, however, that $\pi$ was constructed from the incidence matrix of $G$ and this value is present only when the entry on column $v_1$, row $v_2$ is non-zero. We conclude there is an edge between $v_1$ and $v_2$ in $G$. This shows $f_\phi$ is a homomorphism and $G$ contains a clique of size $l$.

The above reduction can be directly used within the cross-composition framework to show our result:

**Proof of Theorem 6**

Denote the **Permutation Pattern** problem by $Q$. We set $L$ to be the set of all pairs $(K_l, G)$, where $K_l$ is a clique, $G$ is a connected graph containing $K_l$ as a subgraph. It is widely known that deciding $x \in L$ is **NP-complete**.

We introduce a cross-composition of $L$ into $Q$. Let $R$ be the relation such that $(K^1, G^1)R(K^2, G^2)$ if and only if $|V(K^1)| = |V(K^2)|$ and $|V(G^1)| = |V(G^2)|$. For instances $(K_1, G_1), (K_1, G_2), (K_1, G_3), \ldots, (K_l, G_i)$ from the same equivalence class of $R$, we produce an instance of the **Permutation Pattern** problem where we ask if $\pi_z(K_l)$ is in $\pi_z(G)$, where $G$ is a disjoint union of graphs $G_1, \ldots, G_i$ and $z$ is set to $4 \cdot |V(G_1)| + 4$. Lemma 4 shows that the answer to this problem is YES if and only if at least one of the instances $(K_l, G_i)$ belong to $L$. Since the parameter of the pattern matching instance is $|\pi_z(K_l)|$, which can be bounded by $|V(G_i)| \cdot 2z + |V(G_i)|^2$ for any $i$, we can apply Theorem 5. This finishes the proof.

4 Conclusion

Guillemot and Marx [9] have shown that the **Permutation Pattern** problem can be solved in $2^{O(\ell^2 \log \ell)} \cdot n$ time. They raised the question of whether a faster
FPT algorithm could be obtained and outlined a strategy for achieving this using their notion of decompositions of permutations. This relied on the bound from the Stanley-Wilf conjecture not being tight. However, Fox [8] has shown this is not the case and only a minor speed-up can be achieved in this way. The non-existence of a polynomial kernel is a further indication of the difficulty of the problem.

References

[1] S. Ahal, Y. Rabinovich: On Complexity of the Subpattern Problem, SIAM J. Discrete Math., 22(2), 629-649, 2008.

[2] H. L. Bodlaender, R. G. Downey, M. R. Fellows, D. Hermelin: On problems without polynomial kernels, J. Comput. Syst. Sci., 75(8), 423-434, 2009.

[3] H. L. Bodlaender, B. M. P. Jansen, S. Kratsch: Cross-Composition: A New Technique for Kernelization Lower Bounds, Proc. 28th STACS, 165–176, 2011.

[4] M.-L. Bruner, M. Lackner: The computational landscape of permutation patterns, CoRR, abs/1301.0340, 2013.

[5] M.-L. Bruner, M. Lackner: A fast algorithm for permutation pattern matching based on alternating runs, F. V. Fomin and P. Kaski, editors, SWAT, 7357, Lecture Notes in Computer Science, 261-270, Springer, 2012.

[6] P. Bose, J. F. Buss, A. Lubiw: Pattern Matching for Permutations, Inf. Process. Lett., 65(5), 277-283, 1998.

[7] L. Fortnow, R. Santhanam: Infeasibility of instance compression and succinct PCPs for NP, J. Comput. Syst. Sci., 77(1), 91-106, 2011.

[8] J. Fox: Stanley-Wilf limits are typically exponential (preprint), arXiv:1310.8378, 2014.

[9] S. Guillemot, D. Marx: Finding Small Patterns in Permutations in Linear Time, Proc. SODA 2014, 2014.

[10] M. Klazar: The F"uredi-Hajnal conjecture implies the Stanley-Wilf conjecture, Formal Power Series and Algebraic Combinatorics, 250-255, 2000.

[11] D. E. Knuth: The Art of Computer Programming, Vol. 1: Fundamental Algorithms, 2nd edition, Addison-Wesley, 1973.

[12] L. Lovász: Combinatorial Problems and Exercises, North-Holland, 1979.
[13] A. Marcus, G. Tardos: Excluded permutation matrices and the Stanley-Wilf conjecture, J. Comb. Theory, Ser. A, 107(1), 153-160, 2004.

[14] D. Rotem: Stack-sortable permutations, Discrete Math, 33, 185-196, 1981.

[15] R. Simion, F. W. Schmidt: Restricted permutations, European J. Combinatorics, 6, 383-405, 1985.