FAST ALGORITHMS FOR FINDING PATTERN AVOIDERS AND COUNTING PATTERN OCCURRENCES IN PERMUTATIONS

WILLIAM KUSZMAUL
Stanford University
kuszmaul@stanford.edu

Abstract. Given a set Π of permutation patterns of length at most k, we present an algorithm for building $S_{\leq n}(\Pi)$, the set of permutations of length at most n avoiding the patterns in Π, in time $O(|S_{\leq n-1}(\Pi)| \cdot k + |S_n(\Pi)|)$. Additionally, we present an $O(n!k)$-time algorithm for counting the number of copies of patterns from Π in each permutation in S_n. Surprisingly, when $|\Pi| = 1$, this runtime can be improved to $O(n!)$, spending only constant time per permutation. Whereas the previous best algorithms, based on generate-and-check, take exponential time per permutation analyzed, all of our algorithms take time at most polynomial per outputted permutation.

If we want to solve only the enumerative variant of each problem, computing $|S_{\leq n}(\Pi)|$ or tallying permutations according to Π-patterns, rather than to store information about every permutation, then all of our algorithms can be implemented in $O(n^{k+1}k)$ space.

Using our algorithms, we generated $|S_5(\Pi)|, \ldots, |S_{16}(\Pi)|$ for each $\Pi \subseteq S_4$ with $|\Pi| > 4$, and analyzed OEIS matches. We obtained a number of potentially novel pattern-avoidance conjectures.

Our algorithms extend to considering permutations in any set closed under standardization of subsequences. Our algorithms also partially adapt to considering vincular patterns.

Citation: William Kuszmaul. "Fast Algorithms for Finding Pattern Avoiders and Counting Pattern Occurrences in Permutations." To appear in *Mathematics of Computation.* (2016).