Counting permutation classes quickly using tilings

Jay Pantone Marquette University

This talk is based on joint work with Christian Bean

A *tiling* is a way of representing a set of gridded permutations that is defined by the avoidance of some gridded patterns (obstructions) and the containment of others (requirements). Tilings were introduced by Albert, Bean, Claesson, Nadeau, Pantone, and Ulfarsson [ABC+25] in order to algorithmically discover the structure of permutation classes using a method called Combinatorial Exploration.

The figure below shows, on the left, a tiling with obstruction patterns shown in red with circle points and solid edges and requirement patterns shown in blue with square points and dashed edges. This tiling represents all 2×3 gridded permutations that avoid 132 in cell $(0,0)$ and in cell $(2,0)$, avoid the pattern 123 with the 1 in cell $(0,0)$ and the 2 and the 3 in cell $(2,0)$, and contain exactly one point in cell $(1,1)$. The tiling on the right is just visually simplified, representing the same set.

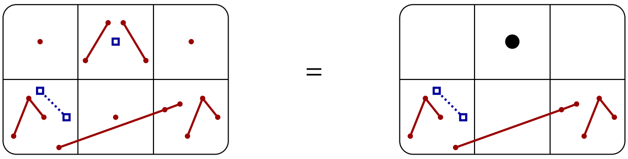

Figure 1: ([ABC+25]) On the left, a tiling with all obstructions and requirements depicted. On the right, the same tiling shown using visual shortcuts.

We show in this talk that tilings have utility beyond their role in Combinatorial Exploration. We will present an algorithm that uses tilings to describe how permutations in any given class can be built by inserting new maximum entries into existing permutations in the class. This is essentially the *insertion encoding* of Albert, Linton, and Ruškuc [ALR05] and of Vatter [Vat12], using tilings to keep track of where new entries can be inserted. This is also similar to the method used by Conway, Guttmann, and Zinn-Justin [CGZJ18] to compute the number of 4231-avoiding permutations up length 50, except that it can be used on any permutation class, and does not require any up-front work of understanding the structure of the class.

For some permutation classes, our method automatically discovers a polynomial-time counting algorithm. For others it is exponential, but with a lower base than the growth rate of the class. The method is also easily parallelizable.

References

[ABC+25] Michael H. Albert, Christian Bean, Anders Claesson, Émile Nadeau, Jay Pantone, and Henning Ulfarsson. Combinatorial Exploration: An algorithmic framework for enumeration. *Memoirs of the American Mathematical Society (to appear)*, 2025+.

[ALR05] Michael H. Albert, Steve Linton, and Nik Ruškuc. The insertion encoding of permutations. *Electron. J. Combin.*, 12:Research Paper 47, 31, 2005.

[CGZJ18] Andrew R. Conway, Anthony J. Guttmann, and Paul Zinn-Justin. 1324-avoiding permutations revisited. *Advances in Applied Mathematics*, 96:312 – 333, 2018.

[Vat12] Vincent Vatter. Finding regular insertion encodings for permutation classes. *J. Symbolic Comput.*, 47(3):259–265, 2012.