Efficient Enumeration of All Ladder Lotteries with k Bars

Katsuhisa Yamanaka[1] and Shin-ichi Nakano[2]

[1] Graduate School of Information Systems, University of Electro-Communications, 1-5-1 Chofugaoka, Chofu, Tokyo 182-8585, Japan. `yamanaka@is.uec.ac.jp`

[2] Department of Computer Science, Gunma University, 1-5-1 Tenjin-cho, Kiryu, Gunma 376-8515, Japan. `nakano@cs.gunma-u.ac.jp`

A *ladder lottery*, known as the "Amidakuji" in Japan, is a common way to choose an assignment randomly. Formally, a ladder lottery of a permutation $\pi = (p_1, p_2, \ldots, p_n)$ is a network with n vertical lines (*lines* for short) and many horizontal lines (*bars* for short) as follows (see Fig. 1). The i-th line from the left is called *line i*. The top ends of the n lines correspond to π. The bottom ends of the n lines correspond to the identical permutation $(1, 2, \ldots, n)$. Each bar connects two consecutive lines. Each number p_i in π starts at the top end of line i, and goes down along the line, then whenever p_i comes to an end of a bar, p_i goes horizontally along the bar to the other end, then goes down again. Finally p_i reaches the bottom end of line p_i. We can regard a bar as a modification of the "current" permutation. In a ladder lottery a sequence of such modifications always results in the identical permutation $(1, 2, \ldots, n)$.

The ladder lotteries are strongly related to primitive sorting networks, which are deeply investigated by Knuth [2]. A comparator in a primitive sorting network replaces p_i and p_j by $\min(p_i, p_j)$ and $\max(p_i, p_j)$, while a bar in a ladder lottery always exchanges them.

Given a permutation $\pi = (p_1, p_2, \ldots, p_n)$ the minimum number of bars to construct ladder lotteries of π is equal to the number of "reverse pairs" in π, which are pairs (p_i, p_j) in π with $p_i > p_j$ and $i < j$. A ladder lottery of π with the minimum number of bars is *optimal*. The ladder lottery in Fig. 2 has eight bars, and its corresponding permutation (2,6,4,1,5,3) has eight reverse pairs: (2,1), (6,4), (6,1), (6,5), (6,3), (4,1), (4,3) and (5,3), so the ladder lottery is optimal.

In [5] we gave an algorithm to enumerate all optimal ladder lotteries of a given permutation π. The algorithm generates all optimal ladder lotteries of π in $O(1)$ time for each. The idea of our algorithm in [5] is as follows. We first define a tree structure T_π, called *the family tree*, among optimal ladder lotteries, in which each vertex of T_π corresponds to each optimal ladder lottery and each edge of T_π corresponds to a relation between two optimal ladder lotteries. Then we design an efficient algorithm to generate all child vertices

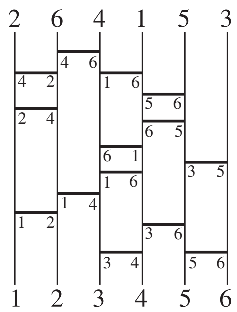

Fig. 1. A ladder lottery of the permutation (2,6,4,1,5,3) with 14 bars.

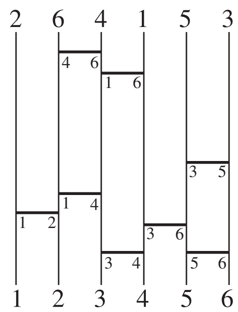

Fig. 2. An optimal ladder lottery of the permutation (2,6,4,1,5,3).

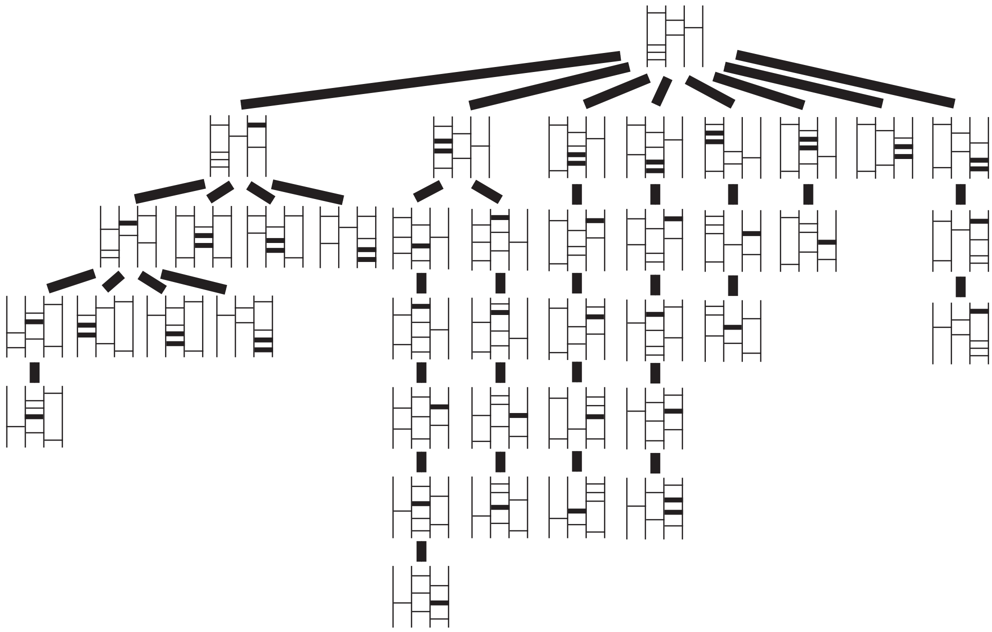

Fig. 3. The family tree $T_{\pi,k}$, where $\pi = (4, 2, 3, 1)$ and $k = 7$.

of a given vertex in T_π. Applying the algorithm recursively from the root of T_π, we can generate all vertices in T_π, and also corresponding optimal ladder lotteries. Based on such tree structure, but with some other ideas, a lot of efficient enumeration algorithms are designed $[1, 3, 4]$.

However the algorithm in [5] works only if the number of bars is minimum. In this paper we generalize the algorithm to enumerate all ladder lotteries in $S_{\pi,k}$, which is the set of all ladder lotteries of a given permutation π with exactly k bars. The algorithm enumerates all ladder lotteries in $S_{\pi,k}$ in $O(1)$ time for each. Note that if k is smaller than the number of reverse pairs in π then $S_{\pi,k} = \phi$. Also if the parity of k does not match the parity of the number of reverse pairs in π then $S_{\pi,k} = \phi$. In this paper we design a new family tree (see Fig. 3) to enumerate all ladder lotteries in $S_{\pi,k}$ for any k. Our main result is as follows.

Theorem 1. *There is an algorithm to enumerate all ladder lotteries in $S_{\pi,k}$ in $O(1)$ time for each.*

References

1. D. Avis and K. Fukuda, *Reverse search for enumeration*, Discrete Applied Mathematics, Vol.65, pp.21–46, 1996.
2. D.E. Knuth, *Axioms and Hulls, LNCS 606*, Springer-Verlag, 1992.
3. Z. Li and S. Nakano, Efficient generation of plane triangulations without repetitions, *Proc. The 28th International Colloquium on Automata, Languages and Programming (ICALP 2001), LNCS 2076*, pp.433–443, 2001.
4. S. Nakano, Efficient generation of triconnected plane triangulations, *Comput. Geom. Theory and Appl.*, Vol.27, No.7, pp.109–122, 2004.
5. K. Yamanaka and S. Nakano and Y. Matsui and R. Uehara and K. Nakada, Efficient enumeration of all ladder lotteries, *Proc. The 20th Workshop on Topological Graph Theory (TGT20)*, pp.150–151, 2008.