Generation of Permutations by Using an Input Restricted-deque or an Output Restricted-deque

ICHIRO SEMBA*

We consider the problems of generating all permutations on {1, 2, . . . , n} by using an output (input) restricted-deque. The average running time per permutation obtained is proved to be bounded by a constant.

1. Introduction

We consider the problem of generating all permutations on {1,2, . . . , n} by using an output (input) restricted-deque.

An output restricted-deque (ORDQ) is a list for which insertions are made at both ends and deletions are made at one end. An input restricted-deque (IRDQ) is a list for which insertions are made at one end and deletions are made at both ends.

ORDQ

insertion→ ←insertion
→deletion

IRDQ

←insertion
deletion← →deletion

We consider four kinds of permutations, output restricted-deque realizable permutation, output restricted-deque sortable permutation, input restricted-deque realizable permutation and input restricted-deque sortable permutation.

An output restricted-deque realizable permutation on {i, $i+1$, . . . , j} ($i \leqq j$) is defined to be a permutation which is constructed from a permutation $i \ldots j$ by using ORDQ.

An output restricted-deque sortable permutation on {i, $i+1$, . . . , j} ($i \leqq j$) is defined to be a permutation from which a permutation $i \ldots j$ is constructed by using ORDQ.

An input restricted-deque realizable permutation on {i, $i+1$, . . . , j} ($i \leqq j$) is defined to be a permutation which is constructed from a permutation $i \ldots j$ by using IRDQ.

An input restricted-deque sortable permutation on {i, $i+1$, . . . , j} ($i \leqq j$) is defined to be a permutation from which a permutation $i \ldots j$ is constructed by using IRDQ.

Example. For $n=4$, four kinds of permutations are presented.

ORDQ realizable permutation	ORDQ sortable permutation	IRDQ realizable permutation	IRDQ sortable permutation
1 2 3 4	1 2 3 4	1 2 3 4	1 2 3 4
1 2 4 3	1 2 4 3	1 2 4 3	1 2 4 3
1 3 2 4	1 3 2 4	1 3 2 4	1 3 2 4
1 3 4 2	1 3 4 2	1 3 4 2	1 3 4 2
1 4 2 3	1 4 2 3	1 4 2 3	1 4 2 3
1 4 3 2	1 4 3 2	1 4 3 2	1 4 3 2
2 1 3 4	2 1 3 4	2 1 3 4	2 1 3 4
2 1 4 3	2 1 4 3	2 1 4 3	2 1 4 3
2 3 1 4	2 3 1 4	2 3 1 4	2 3 1 4
2 3 4 1	2 3 4 1	2 3 4 1	2 3 4 1
2 4 1 3	2 4 1 3	2 4 1 3	2 4 1 3
2 4 3 1	(2 4 3 1)	2 4 3 1	2 4 3 1
3 1 2 4	3 1 2 4	3 1 2 4	3 1 2 4
3 1 4 2	3 1 4 2	3 1 4 2	3 1 4 2
3 2 1 4	3 2 1 4	3 2 1 4	3 2 1 4
3 2 4 1	3 2 4 1	3 2 4 1	(3 2 4 1)
3 4 1 2	3 4 1 2	3 4 1 2	3 4 1 2
3 4 2 1	3 4 2 1	3 4 2 1	3 4 2 1
4 1 2 3	4 1 2 3	4 1 2 3	4 1 2 3
(4 1 3 2)	4 1 3 2	4 1 3 2	4 1 3 2
4 2 1 3	4 2 1 3	(4 2 1 3)	4 2 1 3
(4 2 3 1)	(4 2 3 1)	(4 2 3 1)	(4 2 3 1)
4 3 1 2	4 3 1 2	4 3 1 2	4 3 1 2
4 3 2 1	4 3 2 1	4 3 2 1	4 3 2 1

Imamiya and Nozaki [1] have shown the necessary and sufficient condition for an output (input) restricted-deque realizable (sortable) permutation. However, the problem of generating these permutations is left unsolved.

The algorithm for generating all output (input) restricted-deque realizable permutations is based on this necessary and sufficient condition and the direct inser-

*Department of Pure and Applied Sciences, The College of Arts and Sciences, University of Tokyo.

tion order on the set of all permutations on $\{1, 2, \ldots, n\}$, proposed by Trojanowski [2]. The algorithm for generating all output (input) restricted-deque sortable permutations is based on this necessary and sufficient condition and another direct insertion order on the set of all permutations on $\{1, 2, \ldots, n\}$, proposed by Semba [3].

The average running time per output (input) restricted-deque realizable (sortable) permutation is proved to be bounded by a constant.

We note that permutations on $\{1, 2\}$ and $\{1, 2, 3\}$ are output (input) restricted-deque realizable (sortable) permutations. In this paper we discuss in detail the problem of generating all output restricted-deque realizable permutations. The other problems are omitted because they are solved in a similar way.

2. Generation of All Output Restricted-deque Realizable Permutations

In this section, we will present a generating algorithm for all output restricted-deque realizable permutations on $\{1, 2, \ldots, n\}$ and prove that the average running time per output restricted-deque realizable permutation is bounded by a constant. The following property is fundamental and has been shown by Imamiya and Nozaki [1].

Property 2.1 Let $n \geqq 4$.
A permutation on $\{1, 2, \ldots, n\}$ $a_1 \ldots a_n$ is an output restricted-deque realizable permutation, if and only if there are no subsequences $a_i a_j a_k a_l$ such that $a_i > a_k > a_l > a_j$ or $a_i > a_k > a_j > a_l$ $(1 \leqq i < j < k < l \leqq n)$.

We propose constructing an output restricted-deque realizable permutation on $\{1, 2, \ldots, n\}$ by inserting the new element n into possible positions of a permutation on $\{1, 2, \ldots, n-1\}$. If a permutation on $\{1, 2, \ldots, n-1\}$ is not an output restricted-deque realizable permutation, then no output restricted-deque realizable permutations on $\{1, 2, \ldots, n\}$ can be constructed by inserting the new element n into any possible positions. Thus, it is sufficient to consider an output restricted-deque realizable permutation on $\{1, 2, \ldots, n-1\}$. We will determine the positions into which the new element n is to be inserted. By Property 2.1, we have to insert the new element n so that a subsequence $n a_j a_k a_l$ such that $n > a_k > a_l > a_j$ or $n > a_k > a_j > a_l$ $(1 \leqq j < k < l \leqq n-1)$ will not be formed.

Property 2.2 Let $n \geqq 4$. Suppose that the new element n is inserted into an output restricted-deque realizable permutation $a_1 \ldots a_{n-1}$ on $\{1, 2, \ldots, n-1\}$.

If a subsequence $n a_j a_k a_l$ satisfies $n > a_k > a_l > a_j$ or $n > a_k > a_j > a_l$ $(1 \leqq j < k < l \leqq n-1)$, then a subsequence $n a_m a_{m+1} a_l$ such that $n > a_{m+1} > a_l > a_m$ or $n > a_{m+1} > a_m > a_l$ $(j \leqq m < k)$ has been found.

Proof. Obvious.

From now on, we pay attention to a subsequence $n a_i a_{i+1} a_j$ such that $n > a_{i+1} > a_j > a_i$ or $n > a_{i+1} > a_i > a_j$ $(1 \leqq i < j \leqq n-1)$. By the following property, we know the positions into which the new element n is to be inserted.

Let $P(a_1 \ldots a_{n-1})$ be the rightmost index i satisfying $1 \leqq i \leqq n-3$, $a_i < a_{i+1}$ and $a_{i+1} > \min\{a_{i+2}, \ldots, a_{n-1}\}$ ($P(a_1 \ldots a_{n-1}) = 0$, if the rightmost index is not found).

Property 2.3 Let $n \geqq 4$. Let a permutation $a_1 \ldots a_{n-1}$ on $\{1, 2, \ldots, n-1\}$ be an output restricted-deque realizable permutation.
(1) If $P(a_1 \ldots a_{n-1}) > 0$, then a permutation $a_1 \ldots a_j n \ldots a_{n-1}$ $(P(a_1 \ldots a_{n-1}) \leqq j \leqq n-1)$ is an output restricted-deque realizable permutation and a permutation $a_1 \ldots n a_j \ldots a_{n-1}$ $(1 \leqq j \leqq P(a_1 \ldots a_{n-1}))$ is not an output restricted-deque realizable permutation.
(2) If $P(a_1 \ldots a_{n-1}) = 0$, then a permutation on $\{1, 2, \ldots, n\}$ obtained by inserting the new element n into any possible position of $a_1 \ldots a_{n-1}$ is an output restricted-deque realizable permutation.

Proof. (1) If a permutation $a_1 \ldots a_j n \ldots a_{n-1}$ is not an output restricted-deque realizable permutation, then by Property 2.2 a subsequence $n a_q a_{q+1} a_r$ such that $n > a_{q+1} > a_r > a_q$ or $n > a_{q+1} > a_q > a_r$ $(j < q < r \leqq n-1)$ is found in a permutation $a_1 \ldots a_j n \ldots a_{n-1}$. This means that $P(a_1 \ldots a_{n-1}) \geqq q > j$. This contradicts the assumption that $P(a_1 \ldots a_{n-1}) \leqq j$.

(2) If a permutation $a_1 \ldots n \ldots a_{n-1}$ is not an output restricted-deque realizable permutation, then by Property 2.2 a subsequence $n a_q a_{q+1} a_r$ such that $n > a_{q+1} > a_r > a_q$ or $n > a_{q+1} > a_q > a_r$ $(1 < q < r \leqq n-1)$ is found in a permutation $a_1 \ldots n \ldots a_{n-1}$. This means that $P(a_1 \ldots a_{n-1}) > 0$. This contradicts the assumption that $P(a_1 \ldots a_{n-1}) = 0$. This completes the proof.

This property indicates that the positions into which the new element n can be inserted are successive from right to left. By using this property, a generating algorithm is constructed recursively.

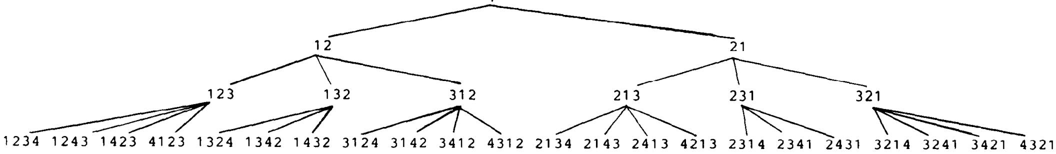

Fig. 1 Output restricted-deque realizable permutation tree.

Example. Two permutations 31425 and 53124 are output restricted-deque realizable permutations. Since $P(31425)=2$, we have four output restricted-deque realizable permutations, 314256, 314265, 314625 and 316425. Since $P(53124)=0$, we have six output restricted-deque realizable permutations, 531246, 531264, 531624, 536124, 563124 and 653124.

The process of generating all output restricted-deque realizable permutations on $\{1, 2, \ldots, n\}$ is represented by the tree (output restricted-deque realizable permutation tree) in Fig. 1. The root 1 is defined to be level 1. The new element n at level n is assumed to be inserted from right to left, according to the direct insertion order proposed by Trojanowski [2].

Now we will show the average running time per output restricted-deque realizable permutation on $\{1, 2, \ldots, n\}$ is bounded by a constant.

Theorem 2.1 Let $n \geqq 2$.
The average running time per output restricted-deque realizable permutation on $\{1, 2, \ldots, n\}$ is bounded by a constant.

Proof. Let B_n be the number of the output restricted-deque realizable permutations on $\{1, 2, \ldots, n\}$. We consider the output restricted-deque realizable permutation tree and assume that a permutation $a_1 \ldots a_{n-1}$ is an output restricted-deque realizable permutation. When a new element n is inserted, three permutations $a_1 \ldots a_{n-1}n$, $a_1 \ldots na_{n-1}$ and $a_1 \ldots na_{n-2}a_{n-1}$ are output restricted-deque realizable permutations. Therefore we have $B_n \geqq 3B_{n-1}$. It follows that $B_n \geqq B_{n-1} + \ldots + B_3$. This means that the number of leaves is greater than the number of other nodes in the tree. This completes the proof.

Note that we do not count the time needed to print out a permutation.

Acknowledgement

I wish to thank the referees for carefully reading this paper.

References

1. IMAMIYA, A. and NOZAKI, A. Generating and Sorting Permutations Using Restricted-deques (in Japanese), *Journal of IPSJ* **17**, 12 (1976), 1128-1134.
2. TROJANOWSKI, A. E. Ranking and Listing Algorithm for k-ary Trees, *SIAM J. Comput.* **7**, 4 (1978), 492-509.
3. SEMBA, I. Generation of Permutations by Using a Stack or a Queue, *JIP* **5**, 3 (1982), 162-171.

(Received July 5, 1982; revised September 28, 1984)