Discrete Mathematics 113 (1993) 263–268
North-Holland

Note

Labelled trees and factorizations of a cycle into transpositions

I.P. Goulden and S. Pepper

Department of Combinatorics and Optimization, University of Waterloo, Waterloo, Ont., Canada N2L 3G1

Received 27 July 1990

Abstract

Goulden, I.P. and S. Pepper, Labelled trees and factorizations of a cycle into transpositions, Discrete Mathematics 113 (1993) 263–268.

Moszkowski has previously given a direct bijection between labelled trees on n vertices and factorizations of the cycle $(12\cdots n)$ in S_n into $n-1$ transpositions. By considering a quadratic recurrence equation and its combinatorial interpretation for trees and for transposition factorizations, we derive another such bijection in a straightforward manner.

Let A_n be the set of labelled trees on n vertices and B_n be the set of $(n-1)$-tuples of transpositions in S_n whose ordered product is the cycle $(12\cdots n)$. Cayley [1] proved that $|A_n|=n^{n-2}$, and Dénes [2] proved that $|B_n|=|A_n|$, by giving a bijection between sets of cardinality $(n-1)!\,|A_n|$ and $(n-1)!\,|B_n|$. Dénes posed the problem of finding a direct bijection between A_n and B_n; the first such bijection was given by Moszkowski [4].

Jackson [3] was able to enumerate factorizations in S_n in several more general situations. The method was based on symmetric group characters, and yielded simple binomial summations as solutions. For example, the number of k-tuples of transpositions in S_n whose ordered product is the cycle $(12\cdots n)$ was shown to be

$$t(n,k)=\frac{1}{n!}\binom{n}{2}^{k}\sum_{i=0}^{n-1}(-1)^{i}\binom{n-1}{i}(n-2i-1)^{k}.$$

The form of this summation suggests that a direct combinatorial explanation should exist, though none is known. Since $t(n,n-1)=|B_n|$, in looking for this

Correspondence to: I.P. Goulden, Department of Combinatorics and Optimization, University of Waterloo, Waterloo, Ont., Canada N2L 3G1.

direct explanation, it is of interest to know as much about the combinatorics of B_n as possible. Accordingly, in this paper we give a second bijection between A_n and B_n, one which is easy to implement and has a very simple proof.

Our method is to give direct combinatorial proofs that $|A_n|$ and $|B_n|$ satisfy the recurrence equation

$$c_n = \sum_{k=1}^{n-1} (n-k)\binom{n-2}{k-1} c_k c_{n-k}, \qquad n \geqslant 2; \qquad c_1 = 1. \tag{*}$$

A comparison of these proofs yields the required bijection.

To prove that $|A_n|$ satisfies the recurrence relation, it is convenient to define a particular edge-deletion operation. Let t be a labelled tree on m vertices, with vertex labels $\{\beta_1, \ldots, \beta_m\}$, where $\beta_1 < \cdots < \beta_m$ and $m \geqslant 2$. Consider the path in t from vertex β_m to β_{m-1}. Suppose that vertex j is the vertex adjacent to vertex β_m in this path. Now remove edge (β_m, j) from t to obtain two trees. One of these trees, call it t', contains vertex β_m, and the other tree, call it t'', contains vertex β_{m-1}. Define

$$F(t) = (t', t'', j).$$

Proposition 1. $|A_n|$, $n \geqslant 1$ *satisfies* (*).

Proof. Consider an arbitrary $a \in A_n$. If $F(a) = (a', a'', j)$ then a' has vertex labels $\alpha \cup \{n\}$ for some $\alpha \subseteq \{1, 2, \ldots, n-2\}$, and a'' has vertex labels $\bar{\alpha} \cup \{n-1\}$.

Clearly $|\alpha| = k-1$ for some $k = 1, \ldots, n-1$. There are then $\binom{n-2}{k-1}$ choices for α, $|A_k|$ choices for a', $|A_{n-k}|$ choices for a'' and $n-k$ choices for j. This is reversible and the result follows. □

Proposition 2. $|B_n|$, $n \geqslant 1$ *satisfies* (*).

Proof. Consider an arbitrary $(b_1, b_2, \ldots, b_{n-1}) \in B_n$, so $b_1 b_2 \cdots b_{n-1} = (12 \cdots n)$. Now $b_{n-1} = (i, i+k)$ for some $k = 1, \ldots, n-1$ and $i = 1, \ldots, n-k$. Thus, multiplying from left to right,

$$b_1 b_2 \cdots b_{n-2} = (12 \cdots n)(i, i+k) = c_1 c_2$$

where c_1 is the k-cycle $(i, i+1, \ldots, i+k-1)$ and c_2 is the $(n-k)$-cycle $(i+k, i+k+1, \ldots, n, 1, \ldots, i-1)$. Since c_1 and c_2 consist of disjoint elements, we have

$$\prod_{j \in \alpha} b_j = c_1, \qquad \prod_{l \in \bar{\alpha}} b_l = c_2$$

for some $\alpha \subseteq \{1, 2, \ldots, n-2\}$, $|\alpha| = k-1$, where b_j and b_l commute for $j \in \alpha$, $l \in \bar{\alpha}$.

Thus there are $\binom{n-2}{k-1}$ choices for α, $|B_k|$ choices for $(b_{\alpha_1}, \ldots, b_{\alpha_{k-1}})$, $|B_{n-k}|$ choices for $(b_{\bar{\alpha}_1}, \ldots, b_{\bar{\alpha}_{n-k-1}})$ and $n-k$ choices for i, where $k = 1, \ldots, n-1$. This is reversible and the result follows. □

After comparing these combinatorial proofs, we are in a position to describe a recursive algorithm that provides the bijection.

Algorithm 3. The inputs are a labelled tree t on m vertices, with vertex labels $\{\beta_1, \ldots, \beta_m\}$, $\beta_1 < \cdots < \beta_m$, $m \geq 1$, and an m-tuple of integers $(l_1, \ldots, l_m)$. If $m = 1$ then STOP. Otherwise let $F(t) = (t', t'', j)$, let k be the number of vertices in t', and suppose that j is the rth smallest label in t''. Then output the transposition $b_{\beta_{m-1}} = (l_r, l_{r+k})$, perform Algorithm 3 with tree t' and k-tuple $(l_r, l_{r+1}, \ldots, l_{r+k-1})$ as inputs, and perform Algorithm 3 with tree t'' and $(m-k)$-tuple $(l_{r+k}, \ldots, l_m, l_1, \ldots, l_{r-1})$ as inputs.

Theorem 4. *For arbitrary $a \in A_n$, if we perform Algorithm 3 with tree a and n-tuple $(1, 2, \ldots, n)$ as inputs, we obtain $(b_1, \ldots, b_{n-1}) \in B_n$, and this is reversible, providing the required bijection.*

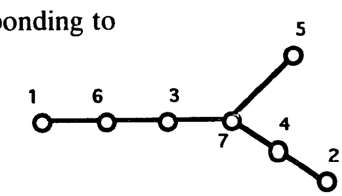
Proof. Every $i = 1, \ldots, n-1$ will appear as the second largest label in one of the sub-trees on which the algorithm is performed, so $b_1, \ldots, b_{n-1}$ are all defined.

Comparison of Proposition 1 and 2 shows that in the bijection, removal of an edge from a tree and the two resulting sub-trees correspond to factoring out the last transposition and the two resulting cycles. In applying Algorithm 3, we output this last transposition and recursively consider the sub-trees and the list of elements on the corresponding cycle. $\square$

We now give an example of the bijection, in which the operation of Algorithm 3 is represented by a binary tree. Each node represents one application of the algorithm, giving the input tree t and list, and output transposition. The edge to be deleted from t is doubled. The left offspring treats the sub-tree t' and the right offspring treats the sub-tree t''. Nodes representing applications of the algorithm which STOP have been deleted.

Example 5. See Fig. 1.

This tells us that corresponding to



$\in A_7$ there is $((6,7), (4,2), (6,1), (3,4), (2,5), (2,6)) \in B_7$, and indeed, to check, multiplying these transpositions in this order yields the cycle $(12 \cdots 7)$, as required.

It is straightforward to reverse this bijection, as follows. Given an $(n-1)$-tuple of transpositions $(b_1, \ldots, b_{n-1})$ such that $b_1 \cdots b_{n-1} = (12 \cdots n)$, we can re-

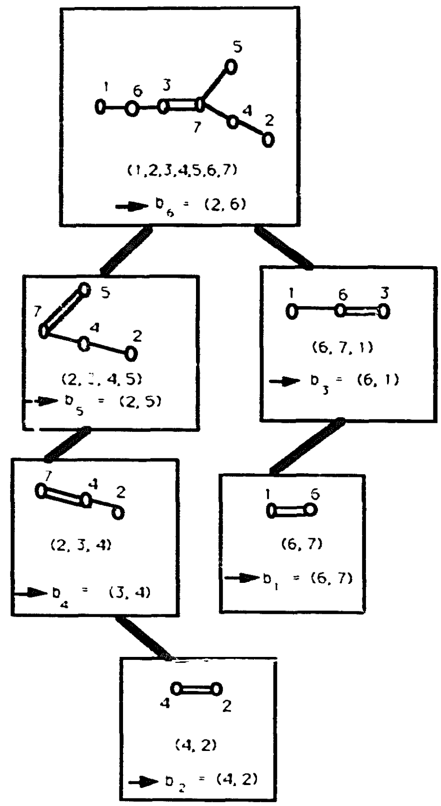


Fig. 1.

construct the binary tree created by Algorithm 3 in two passes, one down, and a second up.

On the first pass, for each vertex of the binary tree we identify the list of vertices in the cycle, the transposition, and the value of r. The list is passed from above, and the transposition is that of maximum subscript containing two elements in the list. We begin with the list $(1, 2, \ldots, n)$ (and thus the transposition b_{n-1}) at the root vertex. The value of r is straightforward at each vertex.

Simultaneously, we assign the labels $1, 2, \ldots, n$ to the 'STOP' vertices of the binary tree as follows. We pass each vertex a label from above, which is assigned to the vertex if it is a STOP vertex. Otherwise, the label is passed to the left offspring of the vertex, and the subscript of the transposition corresponding to the vertex is passed to the right offspring of the vertex. We begin by passing the label 'n' to the root vertex.

The second pass up the tree is now used to join the labelled STOP vertices as specified iteratively by the values of r at each internal vertex.

Finally, this reversal of the bijection is illustrated by an example.

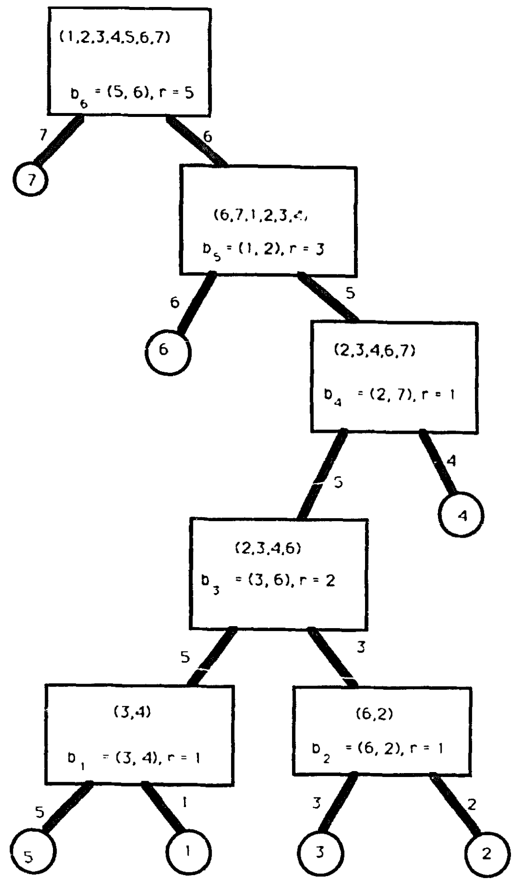
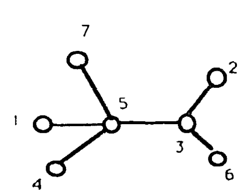


Fig. 2.

Example 6. Given $((3,4),(2,6),(3,6),(2,7),(1,2),(5,6)) \in B_7$, we complete the first pass down to give the binary tree in Fig. 2, in which the labels on the edges give the passed labels, and the circled vertices are the labelled STOP vertices.

The second pass up this binary tree immediately yields the corresponding tree in A_7:



Acknowledgements

This work was supported by grant A8907 from the Natural Sciences and Engineering Research Council of Canada.

References

[1] A. Cayley, A theorem on trees, Quart. J. Math. Oxford 23 (1889) 376–378.
[2] J. Dénes, The representation of a permutation as the product of a minimal number of transpositions and its connection with the theory of graphs, Publ. Math. Institute Hung. Acad. Sci. 4 (1959) 63–70.
[3] D.M. Jackson, Some combinatorial problems associated with products of conjugacy classes of the symmetric group, J. Combin. Theory Ser. A 49 (1988) 363–369.
[4] P. Moszkowski, A solution to a problem of Dénes: a bijection between trees and factorizations of cyclic permutations, European J. Combin. 10 (1989) 13–16.