Egerváry Research Group
on Combinatorial Optimization

Technical reportS

TR-2005-16. Published by the Egerváry Research Group, Pázmány P. sétány 1/C, H–1117, Budapest, Hungary. Web site: www.cs.elte.hu/egres . ISSN 1587–4451.

On Prüfer codes

Tamás Fleiner

2005 December

On Prüfer codes

Tamás Fleiner*

Abstract

We give an alternative proof of Cayley's theorem on the number of labelled trees. Essentially, we use the Prüfer code, but the method seems to be novel.

Keywords: labelled trees, Prüfer code

1 Introduction

Cayley's well known theorem [1] states that there are exactly n^{n-2} different trees on n labelled vertices. Prüfer's proof [2] is based on the so-called Prüfer code that describes each tree T by a sequence $p(T)$ of $n-2$ labels. More precisely, we delete the smallest labelled leaves one by one and the label sequence of the neigbours of the first $n-2$ leaves is the (unique) Prüfer code $p(T)$ of T. From a Prüfer-code $p(T)$ of a tree, it is not so difficult to reconstruct T. But it takes some effort to prove that for any sequence s of $n-2$ labels, the graph G we ,,reconstruct" is indeed a tree with $p(G)=s$. In section 2, we give an alternative coding of labelled trees that turns out to be closely related to the Prüfer code and show a formula for the number of labelled trees with prescribed distance between two vertices.

2 An alternative proof of Cayley's theorem

Let T be a tree on vertices $v_1, v_2, \ldots, v_n$, and call the index of a vertex its *label*. Orient the edges of T away from v_1, and let us label each edge e of T by the label of the tail of $\vec{e}$. Let T_i be the subtree spanned by the first i vertices of T, that is, T_i is the unique inclusionwise minimal subtree of T that contain all vertices $v_1, v_2, \ldots, v_i$. Let $P_i = (u_1, u_2, \ldots, u_k = v_i)$ be the unique path from T_{i-1} to vertex i, such that

*Budapest University of Technology and Economics, Department of Computer Science and Information Theory, Magyar tudósok körútja 2. H-1117, Budapest, Hungary. The author is a member of the Egerváry Research Group (EGRES). Research is supported by the János Bolyai Research Fellowship of the Hungarian Academy of Sciences OTKA grants F 037301 and T 037547, by the European MCRTN Adonet, Contract Grant No. 504438 and by the Egerváry Research Group of the Hungarian Academy of Sciences. e-mail: `fleiner@cs.bme.hu`.

$V(P_i) \cap V(T_{i-1}) = \{u_1\}$ and $T_i = T_{i-1} \cup P_i$. (If $v_i \in V(T_{i-1})$ then $P_i = (v_i)$ is a trivial path.) Number the edges of T as $e_1, e_2, \ldots, e_{n-1}$ in the order as they appear on directed paths $P_1, P_2, \ldots, P_n$. The code $C(T)$ of tree T is the label sequence of $e_1, e_2, \ldots, e_{n-1}$. So the code $C(T) = (c(1), c(2), \ldots, c(n-1))$ of T is uniqe and

$$c(i) \in \{1, 2, \ldots, n\} \quad \text{for all} \quad i \in \{1, 2, \ldots, n-1\} \qquad \text{and} \qquad c(1) = 1. \quad (1)$$

Our goal is to show that the above coding is a bijection. To this end, let us mark the last edges of each directed path P_i. Observe that if e_j is unmarked then $e_j = v_{c(j)}v_{c(j+1)}$ and its head $v_{c(j+1)}$ is not a vertex of the subtree T^{j-1} formed by edges $e_1, e_2, \ldots, e_{j-1}$. Otherwise, if e_j is marked, then $e_j = v_{c(j)}v_i$ is the last edge of some path P_i and $v_{c(j+1)}$ is the first vertex of the next nontrivial path P_{i+k}. Hence either $v_{c(j+1)} = v_i \notin V(T^{j-1})$ (and thus $e_j = v_{c(j)}v_{c(j+1)}$ again) or $v_{c(j+1)} \in V(T^{j-1})$. As $v_1, v_2, \ldots, v_{i-1} \in V(T^{j-1})$, i is the minimal vertex label that is not used for $V(T^{j-1})$.

Lemma 2.1. *For any sequence $C = (c(1), c(2), \ldots, c(n-1))$ with property (1) (where $1 \leq n \in \mathbb{N}$), there is a unique tree T on vertices $v_1, v_2, \ldots, v_n$ such that $C(T) = C$.*

Proof. The observation before the lemma shows how to find the edges $e_1, e_2, \ldots, e_{n-1}$ from C. Namely, if we have constructed tree T^{j-1} that contains the first $j-1$ edges then in case of $v_{c(j+1)} \notin V(T^{j-1})$ we must choose $e_j = v_{c(j)}v_{c(j+1)}$. Otherwise $e_j = v_{c(j)}v_i$ where i is the minimal vertex label that does not appear in $V(T^{j-1})$.

As each e_j connects a new vertex to T^{j-1}, graph T^{n-1} is a tree on $v_1, v_2, \ldots, v_n$. By the construction, we build up T^{n-1} from paths. Moreover, if v_i is the end of such a path then $v_1, v_2, \ldots, v_{i-1}$ are already vertices of the actual graph T^j, and the leaves of T^j belong to $\{v_1, v_2, \ldots, v_i\}$. Hence $C(T^{n-1}) = C$, as we claimed. □

From the above proof, it follows that the greatest labelled leaf of T^j is hanging on e_j. So if we relabel T by replacing each label with its negative, then the negated reverse of the Prüfer code of the relabelled tree is the last $n-2$ digits of $C(T)$.

Theorem 2.2 (Cayley [1]). *There are exactly n^{n-2} trees on n vertices, for $1 \leq n$.*

Proof. By Theorem 2.1, we have to count the possible codes. Each code is a result of $n-2$ independent choices of n possible values. □

We can also enumerate trees with a prescribed distance between two given vertices.

Corollary 2.3. *There are exactly $(n-2) \cdot (n-3) \cdot \ldots \cdot (n-k) \cdot (k+1) \cdot n^{n-k-2}$ trees on vertices $\{v_1, v_2, \ldots, v_n\}$ such that the distance of v_1 and v_2 is k. ($1 < k < n-1$)*

Proof. C is the code of such a tree if and only if C starts with 1 and follows with $k-1$ pairwise different integers (between 3 and n), and the $(k+1)$st digit is either 2 or equals with one of the first k digits. □

References

[1] Cayley, A. A Theorem on Trees. *Quart. J. Math.* **23**, 376-378, 1889.

[2] Prüfer, H. Neuer Beweis eines Satzes ber Permutationen. *Arch. Math. Phys.* **2**, 742-744, 1918.