The Prüfer code for k-trees

by

C. Rényi and A. Rényi
Budapest, Hungary

§ 1. INTRODUCTION

The notion of k-trees (k-dimensional trees) for $k \geqq 2$, which is a natural generalization of the notion of an ordinary (one-dimensional) tree (a "1-tree"), has been introduced by F. HARARY and E.M. PALMER [1]. A k-tree can be defined either as a k-dimensional simplicial complex with certain properties, or as a graph; in what follows, we take the second point of view.

The simplest way to define a k-tree of order n ($k = 1,2,\ldots; n \geqq k+1$)

The results of this paper have been obtained during June and July 1969. The results of § 2 - on which everything else in this paper is based - are due almost exclusively to Catherine Rényi. Alas, Catherine Rényi died on August 23, 1969. It remained the duty of the author named second to prepare the final text of the present paper; thus all possible shortcomings of the paper are his responsibility.

is to do this recursively. A k-tree of order $k+1$ is a complete graph of order $k+1$. A k-tree $G_{n+1}^{(k)}$ od order $n+1$ $(n \geqq k+1)$ is obtained by taking an arbitrary k-tree $G_n^{(k)}$ of order n and selecting any of its points which form a complete k-graph in $G_n^{(k)}$ and by connecting these k points with a new point*. Thus a k-tree of order n is a graph having n points, $\binom{k+1}{2}+k(n-k-1)$ edges and, in general, for every j with $1 \leqq j \leqq k+1$ containing $\binom{k+1}{j}+(n-k-1)\cdot\binom{k}{j-1}$ complete j-graphs: thus in particular it contains $n-k$ complete $(k+1)$-graphs and $k(n-k)+1$ complete k-graphs.

A point of a k-tree is called an endpoint if it belongs only to a single complete $(k+1)$-graph of the k-tree. Clearly each of the $k+1$ points of a k-tree of order $k+1$ is an endpoint. It is easy to see that every k-tree of order $n \geqq k+2$ contains at least 2 and at most $n-k$ endpoints; this follows immediately from the recursive definition, because a k-tree of order $k+2$ contains evidently 2 endpoints, and by forming from a given k-tree $G_n^{(k)}$ of order n, a k-tree of order $n+1$ by adding to $G_n^{(k)}$ a new point, and joining it to the points of a complete k-graph in $G_n^{(k)}$, the number of endpoints is never decreased.

In what follows we shall consider labelled k-trees, i.e. we suppose that the n points of a k-tree of order n are labelled by the numbers $1, 2, \dots, n$. For the sake of brevity we shall call the point labelled by the number j "the point j". We shall deal mainly with rooted k-trees; a k-tree of order n is called rooted at the root $(\varrho_1, \varrho_2, \dots, \varrho_k)$ where $(\varrho_1, \varrho_2, \dots, \varrho_k)$ is an unordered k-tuple of different integers chosen from the integers $1, 2, \dots, n$, if the complete k-graph consisting of the points $\varrho_1, \dots, \varrho_k$ is contained in the k-tree in question. Notice that a 1-tree is rooted at any one of its points; however a k-tree with $k \geqq 2$ is rooted only at $k(n-k)+1$ k-tuples among the possible $\binom{n}{k}$ k-tuples which can be formed from the integers $1, 2, \dots, n$. In the case $k=1$ the only consequence of looking at a 1-tree as rooted at one of

*In [1] a complete k-graph is called a k-tree of order k; we found it more convenient not to consider such a graph as a k-tree, i.e. to define k-trees of order n only for $n > k+1$.

its points, say ϱ, is that the root ϱ is not considered as an endpoint, even if it is connected with a single point. Similarly for $k \geq 2$ the k points belonging to the root are not considered as endpoints even if they belong to a single complete $(k+1)$-graph.

Notice that every rooted k-tree contains at least one endpoint (not belonging to the root). This is trivial for $k=1$ because every 1-tree contains at least 2 endpoints, of which at least one is different from the root. For $k \geq 2$, the statement follows from the remark (proved easily by induction) that every tree contains at least two endpoints which are not contained in the same complete k-graph, and thus if the k-tree is rooted, at least one of them does not belong to the root.

The aim of the present paper is to extend to k-trees the method by which A. PRÜFER [2] has coded 1-trees. In what follows we shall define the Prüfer code of a 1-tree of order n for 1-trees rooted at some selected point: for the sake of simplicity we suppose that the tree is rooted at the point n.

Prüfer's code is defined for labelled 1-trees of order n (the n points of which are labelled by the numbers $1, 2, \ldots, n$) as follows: we remove from the tree the endpoint which is labelled by the least number among all endpoints of the tree, and write down the number by which the unique point of the tree, to which the removed endpoint was connected, is labelled; we repeat the same procedure with the remaining 1-tree of order $n-1$, and continue this process until there remains only a single edge (a 1-tree of order 2). In this way we obtain a sequence of length $n-2$, each element of which is one of the numbers $1, 2, \ldots, n$. It can be shown that each of such n^{n-2} sequences is the Prüfer codeword of a tree of order n, and this tree can be reconstructed, given the codeword. In the case $k \geq 2$ the Prüfer code of a labelled k-tree of order n (the points of which are labelled by the numbers $1, 2, \ldots, n$) and rooted at a given k-tuple $(\varrho_1, \varrho_2, \ldots, \varrho_k)$ is defined similarly: we remove from the k-tree that endpoint which is labelled by the least number among all endpoints of the k-tree, and write down the (unordered) k-tuple of those numbers by which the k points connected to the removed endpoint are labelled. We repeat the same procedure with the remaining k-tree of order $n-1$, and continue this process until there remains only a complete $(k+1)$-graph (i.e. a k-tree of order $k+1$). In this way we obtain a sequence of $n-k-1$ unordered k-tuples of the numbers $1,2,\dots,n$; however, if $k \geqq 2$, not all such sequences are the codewords of a k-tree of order n, but only those which satisfy certain conditions of admissibility. The problem to be solved consists just in finding these conditions, i.e. in characterizing those sequences of $n-k-1$ unordered k-tuples, formed from the numbers $1,2,\dots,n$, which are the Prüfer codewords of a k-tree of order n; the solution of this characterization problem will lead also to a decoding procedure by which the k-tree can be reconstructed from its Prüfer codeword.

While the coding procedure can be generalized in a straightforward way, it will be seen that the characterization of those sequences which are the codewords of a k-tree, and the decoding procedure are much more involved for $k \geqq 2$. It appeared that to solve this problem, a much more thorough study of the original Prüfer code in the case $k=1$ was needed than was done previously. This study involved a certain coding of permutations which seems not to have been used before. In § 2 we start by discussing this coding of permutations; § 3 contains a detailed study of the Prüfer code for ordinary trees, based on the introduction of the notion of the "redundant Prüfer code" of a 1-tree. In § 4 it will be shown that generalizing for any $k \geqq 2$ for (rooted) k-trees the notion of the redundant Prüfer code the problem of characterization as well as of decoding are solved without encountering any further difficulty. In § 5 we show how the Prüfer code can be used for counting all labelled k-trees of order n as well as for enumerating k-trees of order n satisfying certain conditions.

§ 2. CODING OF PERMUTATIONS

Let $\pi_n = (a_1, a_2, \dots, a_n)$ be a permutation of the numbers

$1, 2, \dots, n$. We call an element a_{k+1} of the permutation π_n $(1 \leqq k \leqq n-1)$ a weak element, if it is preceded by a larger number, i.e. if $\max_{1 \leqq j \leqq k} a_j > a_{k+1}$; otherwise we call a_{k+1} a strong element of the permutation π_n; a_1 is always a strong element.*

Let $e_k(\pi_n)$ denote the number of elements a_j $(j = 1, 2, \dots, k)$ which are larger than a_{k+1} $(k = 1, 2, \dots, n-1)$; thus for any permutation π_n of order n we have $0 \leqq e_k(\pi_n) \leqq k$ and a_{k+1} is a weak element if and only if $e_k(\pi_n) \neq 0$. We now shall prove a series of lemmas.

LEMMA 1. To every sequence $e_1, e_2, \dots, e_{n-1}$ of integers for which $0 \leqq e_k \leqq k$ $(k = 1, 2, \dots, n-1)$ there corresponds a uniquely determined permutation π_n of integers $1, 2, \dots, n$ such that $e_k(\pi_n) = e_k$ $(k = 1, 2, \dots, n-1)$.

PROOF. The permutation π_n in question can be successively constructed starting from its last element and working successively backwards. As a matter of fact, as a_n has to be preceded by e_{n-1} larger numbers, evidently $a_n = n - e_{n-1}$. Let us now omit a_n from the sequence $1, 2, \dots, n$; evidently a_{n-1} is obtained by taking the element having the rank $n-1-e_{n-2}$ in the remaining sequence; similarly, if we have already determined $a_n, a_{n-1}, \dots, a_{k+2}$, let us omit these numbers from the sequence $1, 2, \dots, n$; a_{k+1} $(1 \leqq k \leqq n-1)$ will be equal to the element having the rank $k+1-e_k$ in the remaining sequence, while a_1 will be the unique number of the sequence $1, 2, \dots, n$ which remains after removing $a_n, a_{n-1}, \dots, a_2$.

Thus we obtained that any permutation π_n of the elements $1, 2, \dots, n$ can be coded by the sequence $e_k(\pi_n)$ $(k = 1, 2, \dots, n-1)$.

Let this code be called in what follows the CR-code of the permutation π_n. This code is closely related to another code, due to M. HALL (see [4]), which is defined as follows: Let $u_k(\pi_n)$ denote the number of those numbers less than $k+1$ which follow the number $k+1$ in π_n; then we have

* A strong element is sometimes called an upper record, see e.g. [3].

$0 \leqq u_k(\pi_n) \leqq k$ and the permutation π_n is uniquely determined by the codeword $u_1(\pi_n), u_2(\pi_n), \ldots, u_{n-1}(\pi_n)$; we shall call this coding the MH-code.

It is easy to show that the CR code of a permutation π_n is equal to the MH code of its conjugate permutation π_n^* ; as a matter of fact if in π_n $a_{k+1} = j$, then in π_n^* $a_j^* = k+1$ and if there are in π_n e_k numbers greater than $a_{k+1} = j$ among the numbers $a_1, a_2, \ldots, a_k$, then in π_n^* there are e_k numbers less than $k+1$ among the numbers $a_{j+1}^*, \ldots, a_n^*$.

It is pointed out in [4] that the coding of permutations is of importance in computer programming.

Now we prove

LEMMA 2. The number of those permutations $(a_1, a_2, \ldots, a_n)$ of the numbers $1, 2, \ldots, n$ in which the elements a_{h_i+1} $(i = 1, 2, \ldots, r;$ $1 \leqq h_1 < h_2 < \ldots < h_r < n)$ are weak and all others strong $(r \leqq n-1)$ is equal to $h_1 h_2 \ldots h_r$.

PROOF of LEMMA 2. Let $(e_1, e_2, \ldots, e_{n-1})$ be the CR codeword of a permutation satisfying the requirements of Lemma 2. Then clearly $1 \leqq e_{h_i} \leqq h_i$ $(i = 1, 2, \ldots, r)$ while if j does not belong to the sequence $h_1, h_2, \ldots, h_r$ then $e_j = 0$. Thus the total number of such permutations is equal to the number of ways in which the numbers $e_{h_1}, e_{h_2}, \ldots, e_{h_r}$ can be chosen, i.e. is equal to $h_1 h_2 \ldots h_r$ which was to be proved.

REMARK 1. Let us consider the trivial indentity

$$n! = \prod_{h=1}^{n-1} (h+1) = \sum_{1 \leqq h_1 < h_2 < \cdots < h_r \leqq n-1} h_1 h_2 \ldots h_r \tag{2.1}$$

Lemma 2 furnishes a combinatorial interpretation of the identity (2.1): the left-hand side is equal to the total number of permutations of the elements $1, 2, \ldots, n$ while the term $h_1 h_2 \ldots h_r$ on the right-hand side gives the number of those permutations in which the elements a_{h_i+1} $(i=1,2,\dots,r)$, and only these, are weak.

REMARK 2. It follows from the proof of Lemma 1, that if we want to list all permutations satisfying the requirements of Lemma 2, we may proceed as follows: a_{h_r+1} can take the values $1,2,\dots,h_r$. If the value of a_{h_r+1} is fixed, remove this value from the sequence $1,2,\dots,n$; $a_{h_{r-1}+1}$ can be equal to any of the first h_{r-1} elements of the remaining sequence; in general if the values of $a_{h_r+1},\dots,a_{h_{i+1}+1}$ $(1\leqq i\leqq r-1)$ are already fixed, remove these from the sequence $1,2,\dots,n$, and take a_{h_i+1} to be any one of the first h_i terms of the remaining sequence. Thus working backwards we find all possible choices of the weak elements. The strong elements are then uniquely determined, as they are equal to the remaining $n-r$ numbers in natural order.

REMARK 3. Let us choose at random with uniform distribution one of the $n!$ permutations of the numbers $1,2,\dots,n$; let this permutation be $\pi_n=(\alpha_1,\alpha_2,\dots,\alpha_n)$. Let $P(n;m_1,m_2,\dots,m_s)$ denote the probability that $\alpha_1,\alpha_{m_1+1},\dots,\alpha_{m_s+1}$ $\quad(1\leqq m_1<m_2<\dots<m_s\leqq n-1)$ and only these, are the strong elements (upper records) of the random permutation π_n. Denote by $h_1+1,\dots,h_r+1$ $(r=n-s-1)$ those among the numbers $1,2,\dots,n$ which are not contained in the sequence $1,m_1+1,\dots,m_s+1$. Then we have by Lemma 2.

$$P(n;m_1,m_2,\dots,m_s)=\frac{h_1h_2\cdots h_r}{n!}=\frac{1}{nm_1m_2\cdots m_s}. \tag{2.2}$$

This formula can be found in DAVID and BARTON [3]; the proof given here by means of Lemma 2 is, however, much simpler than the proof in [3].

REMARK 4. The result of Lemma 2 can be also interpreted as follows: let us choose at random with uniform distribution one of the permutations of order n. Let this be $\pi_n=(\alpha_1,\alpha_2,\dots,\alpha_n)$.

Put $\varepsilon_h = 0$ if α_{h+1} is a weak element and $\varepsilon_h = 1$ if α_{h+1} is a strong element of π_n $(h = 0,1,\dots, n-1)$.

Then we have

$$P(\varepsilon_h = 1) = \frac{1}{h+1} \quad \text{and} \quad P(\varepsilon_h = 0) = \frac{h}{h+1} \tag{2.3}$$

$(h = 0,1,\dots,n-1)$ and the random variables $\varepsilon_0, \varepsilon_1, \dots, \varepsilon_{n-1}$ are independent. It follows that putting $S_n = \varepsilon_0 + \varepsilon_1 + \dots + \varepsilon_{n-1}$ i.e. denoting by S_n the total number of strong elements of the random permutation π_n, we have, denoting by $E(\xi)$ the expectation and by $D^2(\xi)$ the variance of a random variable ξ, and by $P(A)$ the probability of an event A,

$$E(S_n) = 1 + \frac{1}{2} + \dots + \frac{1}{n} \sim \log n \tag{2.4}$$

and

$$D^2(S_n) = \sum_{k=1}^{n} \left(\frac{1}{k} - \frac{1}{k^2}\right) \sim \log n \tag{2.5}$$

and it is easy to show that the distribution of $\frac{S_n - \log n}{\sqrt{\log n}}$ tends for $n \to +\infty$ to the standard normal distribution, i.e.

$$\lim_{n \to +\infty} P\left(\frac{S_n - \log n}{\sqrt{\log n}} < x\right) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{x} e^{-\frac{u^2}{2}} du \tag{2.6}$$

It is interesting to compare this result with a related result on sequences of independent observations (see [5]).

§ 3. THE REDUNDANT PRÜFER CODE FOR 1-TREES

Let us consider a labelled 1-tree T_n of order n, rooted at n, and let us perform the process by which its Prüfer codeword is obtained as described in the introduction. Let the least endpoint of T_n be a_1, and the point with which this endpoint is connected be b_1; after removing the point a_1, let a_2 be the least endpoint of the remaining tree, and b_2 the point to which it is connected; in general, after removing $j-1$ points, let a_j be the least endpoint, and b_j the point with which a_j is connected $(1 \leqq j \leqq n-2)$; after removing $n-2$ points only a single edge remains, one of the endpoints of which is the root n; let a_{n-1} be its other endpoint and put $b_{n-1} = n$.

Let us consider the matrix consisting of two rows and $n-1$ columns

$$\begin{pmatrix} a_1, a_2, \dots, a_{n-1} \\ b_1, b_2, \dots, b_{n-1} \end{pmatrix} \tag{3.1}$$

We call the matrix (3.1) <u>the redundant Prüfer codeword</u> of the 1-tree T_n rooted at n, clearly $(b_1, b_2, \dots, b_{n-2})$ is the usual Prüfer codeword which we call the primitive Prüfer codeword of the tree T_n. We shall prove a series of lemmas about the redundant Prüfer code.

LEMMA 3. The redundant Prüfer codeword (3.1) is uniquely determined by the primitive Prüfer codeword $(b_1, b_2, \dots, b_{n-2})$ and can be obtained from the latter in the following way:

1) $b_{n-1} = n$,

2) a_1 is the least integer not contained in the sequence $b_1, b_2, \dots, b_{n-1}$.

3) for every j with $1 \leqq j \leqq n-1$ a_j is the least integer not contained among the numbers $a_1, a_2, \dots, a_{j-1}, b_j, b_{j+1}, \dots, b_{n-1}$.

PROOF. Lemma 3 follows immediately from the rule of construction of the primitive Prüfer code and the definition of the redundant Prüfer code.

REMARK. Lemma 3 contains the rule for decoding the primitive Prüfer code, i.e. the algorithm for reconstructing the tree, given its primitive Prüfer code. We first construct the redundant Prüfer code according to Lemma 3; as clearly (a_j, b_j) $(j = 1, 2, \ldots, n-1)$ are all the edges of the tree in question, once the redundant Prüfer code is given, the tree can be immediately obtained from it.

As usual, we define the degree of a point of a graph as the number of other points of the graph to which it is connected by an edge. We prove now

LEMMA 4. Every number j $(1 \leqq j \leqq n)$ occurs d_j times in the redundant Prüfer code of a tree, where d_j denotes the degree of the point in the tree. If $1 \leqq j \leqq n-1$ then j occurs just once in the upper row and $d_j - 1$ times in the lower row, at places with indices less than the index of its occurrence in the upper row. The number n occurs d_n times in the lower row.

PROOF. Lemma 4 follows from Lemma 3, and from the definition of the redundant Prüfer code.

REMARK. As $b_{n-1} = n$, Lemma 3 implies the well known fact that every number j $(1 \leqq j \leqq n)$ occurs $d_j - 1$ times in the primitive Prüfer code $(b_1, b_2, \ldots, b_{n-2})$.

LEMMA 5. The upper row $(a_1, a_2, \ldots, a_{n-1})$ of the redundant Prüfer code of a 1-tree of order n rooted at the point n, is a permutation of the numbers $1, 2, \ldots, n-1$. If a_{h+1} is a weak element of this permutation, then we have

$$b_h = a_{h+1} \qquad (1 \leqq h \leqq n-2) \tag{3.1}$$

PROOF. The first statement of Lemma 5 follows immediately from Lemma 4. As regards the second statement let a_{h+1} be a weak element of the permutation $(a_1, a_2, \ldots, a_{n-1})$ and let j be the largest natural number not exceeding h such that $a_j > a_{h+1}$. We shall prove the statement $a_{h+1} = b_h$

by induction on the value of $t = h+1-j$. If $t=1$, i.e. if $a_h > a_{h+1}$, let us put $a_{h+1} = v$ and $a_h = u$.

According to Lemma 3, u is the least number not occurring among the numbers $a_1, a_2, \ldots, a_{h+1}, b_h, b_{h+1}, \ldots, b_{n-1}$; b_{n-1} ; and v the least number not occurring among $a_1, a_2, \ldots, a_{h-1}, u, b_{h+1}, \ldots, b_{n-1}$. It follows that v occurs among the numbers $a_1, \ldots, a_{h-1}, b_h, b_{h+1}, \ldots, b_{n-1}$ but it does not occur among $a_1, \ldots, a_{h-1}, b_{h+1}, \ldots, b_{n-1}$; this is possible only if $b_h = v = a_{h+1}$. Thus our statement holds for $t=1$. Suppose it holds for $t = 1, 2, \ldots, m$; we shall show that it holds for $t = m+1$ too. Suppose that $a_{h-m} > a_{h+1}$ and $a_j < a_{h+1}$ for $j = h-m+1, \ldots, h$.

It follows that $a_h, a_{h-1}, \ldots, a_{h-m+1}$ are all weak elements of the permutation $(a_1, a_2, \ldots, a_{n-1})$ and by the induction hypothesis it follows that $b_j = a_{j+1}$ for $j = h-1, \ldots, h-m$. Now the value $a_{h+1} = v$ has to occur among the numbers $a_1, \ldots, a_{h-m+1}, b_{h-m}, \ldots, b_{n-1}$ because otherwise a_{h-m} would have the value v and not a larger value; as, however, the numbers a_i are all different, v can be equal only to b_h, i.e. $b_h = v = a_{h+1}$; this means that $b_h = a_{h+1}$ holds for $t = m+1$ too, and thus the second statement of Lemma 5 follows by induction.

REMARK 1. Notice that the converse of the second statement of Lemma 5 does not hold: $b_h = a_{h+1}$ does not imply that a_{n+1} is a weak element of the permutation $(a_1, a_2, \ldots, a_{n-1})$.

REMARK 2. Clearly $a_{h+1} = j$ is a weak element of the permutation $(a_1, \ldots, a_{n-1})$ if and only if there exists in the tree a path starting from the root, passing through the point j and arriving to a point $k > j$. In this case we call the point j a weak point of the tree.

Now we are in the position to prove

LEMMA 6. Let $\pi_{n-1} = (a_1, a_2, \ldots, a_{n-1})$ be a permutation of the numbers $1, 2, \ldots, n-1$. Let $T_1(n, \pi_{n-1})$ denote the number of those labelled 1-trees of order n, rooted at n, the upper row of the redundant Prüfer codeword of which is equal to the sequence π_{n-1}. Let $a_{h_1+1}, a_{h_2+1}, \dots, a_{h_r+1}$ be the weak elements of the permutation π_{n-1}. Then we have

$$(3.2) \qquad T_1(n, \pi_{n-1}) = \prod_{\substack{h=1 \\ h \neq h_i\ (1 \leq i \leq r)}}^{n-2} (n-h)$$

PROOF. According to Lemma 5 we have $b_{h_i} = a_{h_i+1}$ for $i = 1, 2, \dots, r$ further we have by definition $b_{n-1} = n$.

If h $(1 \leqq h \leqq n-2)$ is not one of the numbers $h_1, \dots, h_r$ then b_h can take on any value different from $a_1, a_2, \dots, a_h$ because according to Lemma 4 the value a_j can occur in the lower row of the redundant Prüfer code only at a place with index $h < j$, i.e. we have $b_h \neq a_j$ if $j < h$. Thus there are $n-h$ possible choices for the value of b_h if h is not one of the numbers $h_1, h_2, \dots, h_r$; this implies that (3.2) holds.

REMARK. Evidently one has, denoting by S_{n-1} the set of all $(n-1)!$ permutations of the numbers $1, 2, \dots, n-1$ and by $T_1(n)$ the total number of labelled 1-trees of order n,

$$(3.3) \qquad T_1(n) = \sum_{\pi_{n-1} \in S_{n-1}} T_1(n, \pi_{n-1}).$$

According to Lemma 2 the total number of permutations is $\pi_{n-1} \in S_{n-1}$ in which the weak elements are $a_{h_1+1}, \dots, a_{h_r+1}$ $(1 \leqq h_1 < h_2 < \dots < h_r \leqq n-2)$ is $h_1 h_2 \dots h_r$. Thus it follows that

$$(3.4) \qquad \begin{aligned} T_1(n) &= \sum_{1 \leqq h_1 < h_2 < \dots < h_r \leqq n-1} h_1 h_2 \dots h_r \prod_{\substack{h=1 \\ h \neq h_i\ (1 \leqq i \leqq n)}} (n-h) = \\ &= \prod_{h=1}^{n-2} (h + n - h) = n^{n-2}. \end{aligned}$$

i.e. we obtain in this way a proof of Cayley's well-known formula

$$T_1(n) = n^{n-2} \tag{3.5}$$

for the total number of labelled trees of order n. Of course, for 1-trees this proof of (3.5) is unnecessarily complicated, and it is much easier to enumerate trees of order n by remarking that each element of the primitive Prüfer codeword can be chosen in n ways, and thus there are n^{n-2} possibilities; however, it will be seen in § 4 that in counting k-trees of order n in the case $k \geqq 2$ this simple method of counting cannot be easily generalized, while Lemma 6 can be easily generalized for k-trees for every k (see Lemma 6^*) and this leads to a method for enumerating all k-trees of order n as well for counting the number of k-trees having certain prescribed properties.

From what has been said we get immediately the following result on the number of weak points of a random 1-tree. As usual, a random 1-tree of order n is defined as a 1-tree chosen at random, with uniform distribution, among the n^{n-2} possible labelled 1-trees of order n. (For the definition of weak points, see Remark 2 to Lemma 5.)

THEOREM 1. Let $W(n,r)$ denote the probability that a random labelled 1-tree of order n, rooted at the point n, should contain exactly r weak points; then we have

$$\sum_{r=0}^{n-2} W(n,r)x^r = \prod_{h=1}^{n-2} \left(1 + \frac{h}{n}(x-1)\right).$$

Thus if V_n is the number of weak points in a random labelled tree of order n we have

$$E(V_n) = \frac{n}{2}\left(1-\frac{1}{n}\right)\left(1-\frac{2}{n}\right)$$

$$D^2(V_n) = \frac{n}{6}\left(1-\frac{1}{n}\right)\left(1-\frac{2}{n}\right)\left(1+\frac{3}{n}\right)$$

and

$$\lim_{n \to +\infty} P\left(\frac{V_n - \frac{n}{2}}{\sqrt{\frac{n}{6}}} < x\right) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{x} e^{-\frac{u^2}{2}} du ,$$

i.e. the number of weak points of a random 1-tree of order n (rooted at the point n) is for large n approximately normally distributed around $\frac{n}{2}$ with variance $\frac{n}{6}$.

REMARK 1. It is interesting to compare Theorem 1 with previous results on the number of inversions of a random 1-tree (see [8] and [9]). The points j_1 and j_2 of a random 1-tree are said to be in inversion if $j_1 < j_2$ and the path from the point j_2 to the point n passes through the point j_1.

REMARK 2. Notice that an endpoint of a 1-tree is never weak. As it is known (see [10]) that the number of endpoints of a random 1-tree of order n is for large n approximately normally distributed with mean value $\frac{n}{e}$ and variance $\frac{n}{e}(1-\frac{2}{e})$, it follows e.g. that the expected number of points of a random 1-tree, which are neither endpoints nor weak points, is asymtptotically equal to $n(\frac{1}{2}-\frac{1}{e})$ for $n \to +\infty$.

§ 4. THE CHARACTERIZATION OF PRÜFER CODEWORDS FOR k-TREES

Let us now generalize the results of the previous § for k-trees with $k \geqq 2$. We consider labelled k-trees of order n rooted at a given complete k-graph, which can be taken without restricting the generality as consisting of the points labelled by the numbers $n-k+1, n-k+2, \ldots, n$. In other words, we consider the set of all k-trees of order n, the points of which are labelled by the numbers $1, 2, \ldots, n$ and which contain the complete graph formed by the points $n-k+1, n-k+2, \ldots, n$. First we define the redundant Prüfer codeword of such a tree. This codeword is a matrix

$$(4.1)\qquad \begin{pmatrix} a_1, a_2, \dots, a_{n-k} \\ B_1, B_2, \dots, B_{n-k} \end{pmatrix}$$

where $(a_1, a_2, \dots, a_{n-k})$ is a permutation of the numbers $1, 2, \dots, n-k$, while B_i $(1 \leq i \leq n-k)$ is a k-tuple of different integers taken from the numbers $1, 2, \dots, n$. The redundant Prüfer codeword of a k-tree T, rooted at the k-tuple $(n-k+1, n-k+2, \dots, n)$, is obtained as follows: a_1 is the least endpoint of the tree, and B_1 is the k-tuple of those points to which a_1 is connected; a_2 is the least endpoint of the tree $T^{(1)}$ obtained after removing from T the point a_1 (and of course all k edges connecting a_1 with a point belonging to B_1) and B_2 is the k-tuple of those points of $T^{(1)}$ which are connected in $T^{(1)}$ to a_2; in general a_{j+1} is the least endpoint of the tree $T^{(j)}$ obtained by removing from T the points $a_1, a_2, \dots, a_j$ and all edges starting at these points, while B_{j+1} is the k-tuple of those points in $T^{(j)}$ which are connected to a_{j+1} (in $T^{(j)}$) $(j = 1, 2, \dots, n-k+1)$. (Remember that the points of the root are not endpoints.) It is easy to see that Lemmas 3, 4 and 5 can be generalized to k-trees for any k; the corresponding statements are as follows:

Lemma 3*. The redundant Prüfer codeword (4.1) of a k-tree is uniquely determined by the primitive Prüfer codeword $(B_1, B_2, \dots, B_{n-k-1})$ and can be obtained from the latter in the following way:

1) B_{n-k} is the k-tuple $(n-k+1, n-k+2, \dots, n)$ (i.e. the root);

2) a_1 is the least integer not contained among the elements of the k-tuples B_i $(i = 1, 2, \dots, n-k)$;

3) For every j, with $1 < j \leq n-k$, a_j is the least integer different from the numbers $a_1, a_2, \dots, a_{j-1}$ and not contained among the elements of the k-tuples $B_j, B_{j+1}, \dots, B_{n-k}$.

In the same way as in the special case $k=1$ Lemma 3* contains the algorithm for decoding the primitive Prüfer code of a k-tree: first one has to construct from the primitive codeword the redundant codeword according to the rules laid down in Lemma 3*; if this is done, the k-tree in question is already defined because evidently the edges of the k-tree are those connecting two points belonging to at least one $(k+1)$-tuple C_j where C_j consists of a_j and of the k points forming the k-tuple B_j $(j = 1, 2, \dots, n-k)$. (Notice that the C_j $(j = 1, 2, \dots, n-k)$ are all complete $(k+1)$-graphs contained in the k-tree in question, and the same edge may belong to several of these complete $(k+1)$-graphs.)

Before generalizing Lemma 4 for $k \geqq 2$, we introduce a definition: the <u>degree</u> of a complete k-graph G formed by k of the points $1, 2, \dots, n$ in a k-tree T, is defined as the number of those complete $(k+1)$-graphs contained in T which contain G.

Now we can formulate

Lemma 4*. Every k-tuple B of k of the integers $1, 2, \dots, n$ belongs to $d(B)$ of the $(k+1)$-tuples $C_j = (a_j, B_j)$ $(j = 1, 2, \dots, n-k)$ where $d(B)$ is the degree of the complete k-graph consisting of the k points of B. The root $R = (n-k+1, n-k+2, \dots, n)$ occurs $d(R)$ times among the k-tuples $B_1, B_2, \dots, B_{n-k}$, while every other k-tuple B for which $d(B) \geqq 1$ occurs just once as the subset of a $(k+1)$-tuple $C_j = (a_j, B_j)$ such that B is not identical with B_j (i.e. such that $a_j \in B$), and B occurs $d(B)-1$ times among the k-tuples B_i with $i < j$.

Finally the generalization of Lemma 5 for $k \geqq 2$ is as follows:

Lemma 5*. The upper row $(a_1, a_2, \dots, a_{n-k})$ of the redundant Prüfer codeword of a k-tree, rooted at the k-tuple $(n-k+1, \dots, n)$ is a permutation of the numbers $1, 2, \dots, n-k$. If a_{n+1} is a weak element of the permutation $(a_1, \dots, a_{n-k})$ then a_{n+1} is contained in the k-tuple B_h $(1 \leqq h \leqq n-k-1)$.

The proofs of Lemmas 3*, 4*, 5* are almost word for word the same in the case $k \geqq 2$ as for $k=1$; therefore there is no need to go into details, except for the second statement of Lemma 5*, the proof of which in the general case is slightly more involved. This proof is as follows: let a_{h+1} be a weak element of the permutation $(a_1, \dots, a_{n-k})$ and let j be the largest number not exceeding h such that $a_j > a_{h+1}$, and put $t = h+1-j$; we apply induction on t as we have done for the special case $k=1$. If $t=1$, i.e. if $a_h > a_{h+1}$ put $a_{h+1}=v$ and $a_h = u > v$. According to Lemma 3*, u is the least number different from $a_1, \dots, a_{h-1}$ and not belonging to any of the k-tuples $B_h, B_{h+1}, \dots, B_{n-k}$ and v the least number different from $a_1, a_2, \dots, a_{h-1}, u$ and not occurring among the elements of the k-tuples $B_{h+1}, \dots, B_{n-k}$; as $v<u$ this is possible only if v belongs to B_h. Thus our statement holds for $t=1$. Suppose our statement holds for $t = 1, 2, \dots, m$; we show that it holds for $t=m+1$ too. Suppose that $a_{h-m} > a_{h+1}$ and $a_j < a_{h+1}$ for $j = h-m+1, \dots, h$. It follows that $a_h, a_{h-1}, \dots, a_{h-m+1}$ are all weak elements of the permutation $(a_1, a_2, \dots, a_{n-k})$ and thus by the induction hypothesis it follows that $a_{j+1} \in B_j$ for $j = h-1, \dots, h-m$.

Now the value $a_{h+1} = v$ has to occur among the elements of at least one of the k-tuples $B_{h-m}, B_{h-m+1}, \dots, B_h$; let j be the largest integer such that $v \in B_j$ $(j \leqq h)$ if we would have $j \leqq h-1$ then both a_{j+1} and v would belong to B_j. Now the k-tuple B_j has to occur in just one $(k+1)$-tuple $C_i = (a_i, B_i)$ $(i > j)$ so that $B_j \neq B_i$, i.e. $a_i \in B_j$; if this would happen for $i = h+1$ in which case $a_i = a_{h+1} = v$, we would have $a_{j+1} \in B_{h+1}$ which is however impossible as a_{j+1} cannot be an element of B_{h+1} because $h+1 > j+1$; or we may have $i \leqq h$ in which case $v \in B_i$ with $i > j$, but this again contradicts the definition of j as the largest index $\leqq h$ such that $v \in B_j$; thus the supposition $j \leqq h-1$ is impossible; thus $j = h$ that is, $v = a_{h+1} \in B_h$, which was to be proved.

Lemmas 3*, 4*, 5* contain a complete characterization of those sequences $B_1, B_2, \dots, B_{n-k-1}$ of $n-k-1$ k-tuples, which are the primitive

Prüfer codewords of a k-tree rooted at the k-tuple $n-k+1, n-k+k, \dots, n$. This characterization is as follows:

THEOREM 2. A sequence $(B_1, B_2, \dots, B_{n-k-1})$ of $n-k-1$ k-tuples from the integers $1, 2, \dots, n$ is the primitive Prüfer code of a k-tree of order n, labelled by the numbers $1, 2, \dots, n$ and rooted at the k-tuple $(n-k+1, \dots, n)$, if and only if it has the following properties:

a) Putting $B_{n-k} = (n-k+1, n-k+2, \dots, n)$, denoting by a_1 the least natural number not contained in any of the k-tuples $B_1, B_2, \dots, B_{n-k}$ and for every $j \leqq n-k$ denoting by a_j the least natural number different from $a_1, \dots, a_{j-1}$ and not contained in any of the k-tuples $B_j, B_{j+1}, \dots, B_{n-k}$ the numbers $a_1, \dots, a_{n-k}$ form a permutation of the numbers $1, 2, \dots, n-k$.

b) Denoting by C_j the $(k+1)$-tuple consisting of a_j and the k elements of the k-tuple B_j, each B_i is the subset of at least one C_j with $i < j \leqq n-k$.

PROOF of THEOREM 2. The conditions a) and b) are necessary because of Lemmas 3*, 4* and 5*. To see that they are sufficient it is enough to point out that if these conditions are satisfied the corresponding k-tree can be recursively constructed starting from the complete $(k+1)$-graph the points of which are the elements of C_{n-k} and working backwards: at the j-th step the new point is $a_{n-k-j+1}$ and the k-tuple of points to which it is connected is $B_{n-k-j+1}$, this k-tuple being a subset of a $(k+1)$-tuple C_i such that $i > n-k-j+1$ $(j = 1, 2, \dots, n-k)$.

We give now some examples which may help to familiarize the reader with the meaning of Theorem 2.

EXAMPLE 1. Let us have $n = 7$, $k = 2$ and consider the sequence of $n-k-1 = 4$ pairs of the integers $1, 2, 3, 4, 5, 6, 7$;

$$(6,7), (6,7), (1,6), (5,6).$$

We shall verify that this is the primitive Prüfer codeword of a 2-

tree of order 7, rooted at the edge $(6,7)$. As a matter of fact condition a) is satisfied with $a_1 = 2$, $a_2 = 3$, $a_3 = 4$, $a_4 = 1$ and $a_5 = 5$. Further the matrix

$$\begin{pmatrix} 2 & 3 & 4 & 1 & 5 \\ (6,7) & (6,7) & (1,6) & (5,6) & (6,7) \end{pmatrix}$$

clearly satisfies condition b).

EXAMPLE 2. Let us have $n=5$, $k=2$. The sequence $(1,2)$, $(4,5)$ is not the primitive Prüfer codeword of a 2-tree of order 5 rooted at the edge $(4,5)$. As a matter of fact while condition a) is satisfied with $a_1 = 3$, $a_2 = 1$, $a_3 = 2$ the matrix

$$\begin{pmatrix} 3 & 1 & 2 \\ (1,2) & (4,5) & (4,5) \end{pmatrix}$$

does not satisfy condition b) because the pair $(1,2)$ is not a subset of either of the triples $(1,4,5)$ and $(2,4,5)$.

In connection with this example it should be pointed out that a sequence of $n-k-1$ k-tuples of the numbers $1,2,\dots,n$ which satisfies condition a) but violates condition b) of Theorem 2 corresponds to a graph the edges of which are the edges of $n-k$ such complete $(k+1)$-graphs which do not form a k-tree

REMARK. It is easy to obtain from Theorem 2 a characterization of the primitive Prüfer codewords of k-trees of order n rooted at any k-tuple. The obvious modification of Theorem 1 for this case is as follows:

THEOREM 2b. A sequence $(B_1, B_2, \dots, B_{n-k-1})$ of $n-k-1$ k-tuples of the integers $1,2,\dots,n$ is the primitive Prüfer codeword of a k-tree of order n the points of which are labelled by the numbers $1,2,\dots,n$ and which is rooted at some k-tuple B of the numbers $1,2,\dots,n$ if it has the following properties:

a) Put $B_{n-k} = B$; let a_1 be the least natural number not contained in any of the k-tuples $B_1, B_2, \dots, B_{n-k}$ and for every j with $1 < j \leqq n-k$; let a_j be the least natural number different from $a_1, \dots, a_{j-1}$ and not contained in any of the k-tuples $B_j, B_{j+1}, \dots, B_{n-k}$. Then $a_1, a_2, \dots, a_{n-k}$ is a permutation of the $n-k$ numbers obtained by omitting from the sequence $1, 2, \dots, n$ the elements of the k-tuple B.

b) Denoting by C_j the $(k+1)$-tuple consisting of a_j and the elements of the k-tuple B_j, each B_i $(1 \leqq i \leqq n-k-1)$ is contained in at least one C_j with $i < j \leqq n-k$.

REMARK. The same sequence $B_1, B_2, \dots, B_{n-k-1}$ may be the primitive Prüfer code of different k-trees rooted at different roots, as is shown by the following.

EXAMPLE 3. The sequence $(3,4), (3,5), (4,5)$ is the primitive Prüfer code of a 2-tree of order 6, rooted at the edge $(4,6)$, the corresponding redundant Prüfer code being

$$\begin{pmatrix} 1 & 2 & 3 & 5 \\ (3,4) & (3,5) & (4,5) & (4,6) \end{pmatrix}.$$

At the same time, the sequence $(3,4), (3,5), (4,5)$ is also the primitive Prüfer code of an other 2-tree of order 6 rooted at the edge $(1,5)$ the corresppnding redundant Prüfer code being

$$\begin{pmatrix} 2 & 6 & 3 & 4 \\ (3,4) & (3,5) & (4,5) & (1,5) \end{pmatrix}.$$

§ 5. ENUMERATION OF k-TREES

We first prove the generalization of Lemma 6 to the case of k-trees for $k \geqq 2$.

LEMMA 6.* Let $\pi_{n-k} = (a_1, a_2, \dots, a_{n-k})$ be a permutation of the numbers $1, 2, \ldots, n-k$. Let $R_k(n, \pi_{n-k})$ denote the number of those labelled k-trees of order n, rooted at the k-tuple $(n-k+1, n-k+2, \ldots, n)$, the upper row of the redundant Prüfer codeword of which is equal to the sequence π_{n-k}. Let $a_{h_1+1}, \ldots, a_{h_r+1}$ be the weak elements of the permutation π_{n-k}. Then we have

$$(5.1) \qquad R_k(n, \pi_{n-k}) = k^r \prod_{\substack{h=1 \\ h \neq h_i \ (1 \leq i \leq r)}}^{n-k-1} [\,k(n-k-h)+1\,]$$

PROOF. According to Lemma 5* we have $a_{h_i+1} \in B_{h_i}$ for $i = 1, 2, \ldots, r$. As by Lemma 4*, B_{h_i} has to be a subset of some $(k+1)$-tuple C_j with $j > h_i$, and a_{h_i+1} is not an element of C_j with $j > h_i+1$, it follows that B_{h_i} has to be a subset of C_{h_i+1}. As the number of k-tuples contained in the $(k+1)$-tuple $C_{h_i}+1$ and containing $a_{h_i}+1$ is equal to k, the number of choices of B_{h_i} is equal to k. If h is not one of the numbers $h_1, h_2, \ldots, h_r$ then the number of possible choices of B_h is equal to the number of all k-tuples contained in at least one of the $(k+1)$-tuples C_j with $j > h$. It is easy to see that this number is equal to $k(n-k-h)+1$; this statement is most easily shown by induction on the value of $t = n-k-h$. If $t = 0$, i.e. if $h = n-k$, then $B_h = B_{n-k}$ is the root $(n-k+1, \ldots, n)$, i.e. the number of choices is one. Suppose our statement holds for some value of $t = n-k-h$, i.e. we know already that the $(k+1)$-tuples $C_{h+1}, \ldots, C_{n-k}$ contain together $kt+1$ k-tuples. As B_h is one of these, C_h contains k such k-tuples which are not contained in any of the $(k+1)$-tuples $C_{h+1}, \ldots, C_{n-k}$ i.e. the number of choices for B_{h-1} is $k(t+1)+1 = k(n-k-(h-1)) + 1$, thus our statement is proved by induction. It follows that the total number of choices of the lower row of the redundant Prüfer codeword with the given upper row π_{n-k} is

$$(5.2)\qquad R_k(n,\pi_{n-k}) = k^r \prod_{\substack{h=1 \\ h \neq h_i\ (1 \leqq i \leqq r)}}^{n-k-1} (k(n-k-h)+1)$$

Thus Lemma 6* is proved.

We now deduce from Lemma 6 the following result which has been proved first by BEINEKE and PIPPERT [6] and also by MOON [7] ; our proof is essentially different from both of these proofs.

THEOREM 3. The total number $R_k(n)$ of labelled k-trees of order n rooted at a given k-tuple is given by the formula

$$(5.3)\qquad R_k(n) = [k(n-k)+1]^{n-k-1},$$

while the total number $T_k(n)$ of labelled (unrooted) k-trees of order n is given by

$$(5.4)\qquad T_k(n) = \binom{n}{k}[k(n-k)+1]^{n-k-2}$$

PROOF. In proving (5.3) we may suppose without restricting the generality that the k-trees in question are rooted at the k-tuple $(n-k+1, n-k+2, \dots, n)$. We have evidently

$$(5.5)\qquad R_k(n) = \sum_{\pi_{n-k} \in S_{n-k}} R_k(n, \pi_{n-k}).$$

where S_{n-k} denotes the set of all permutations of the numbers $1, 2, \dots, n-k$. Using Lemma 6* and Lemma 2 we obtain from (5.5)

$$(5.6)\quad R_k(n) = \sum_{1\leq h_1<h_2<\dots<h_r\leq n-k-1} h_1h_2\dots h_r\cdot k^r \prod_{\substack{h=1\\ h\neq h_i\,(1\leq i\leq r)}}^{n-k-1} [k(n-k-h)+1] = [k(n-k)+1]^{n-k-1}$$

Thus (5.3) is proved.

As has been remarked both in [6] and [7], knowing the total number $R_k(n)$ of rooted labelled k-trees of order n, it is very easy to get the total number $T_k(n)$ of labelled unrooted k-trees of order n, using the obvious identity

$$(5.7)\qquad T_k(n)[k(n-k)+1] = \binom{n}{k} R_k(n)$$

(5.7) follows from the remark that every (unrooted) k-tree of order n contains $k(n-k)+1$ complete k-graphs and thus it can be rooted at any of these; on the other hand one can choose k points out of n in $\binom{n}{k}$ ways, and for each such choice by definition the total number of k-trees of order n rooted at this k-tuple is $R_k(n)$.

From (5.3.) and (5.7) we obtain immediately (5.4).

REMARK. Notice that $R_1(n) = T_1(n)$ and thus for $k=1$ both (5.3) and (5.4) reduce to Cayley's formula

In [7] MOON has also determined the number $R_{k,d}(n)$ of k-trees of order n in which a selected k-tuple of points has the degree d. We shall now deduce this result by our method, i.e. we prove

THEOREM 4. Denoting by $R_{k,d}(n)$ $(k\geq 1,\ n\geq k+1,\ d\geq 1)$ the number of labelled k-trees of order n in which a selected k-tuple has the degree d, we have

$$(5.8)\qquad R_{k,d}(n) = \binom{n-k-1}{d-1}[k(n-k)]^{n-k-d}$$

PROOF. It is no restriction to suppose that the selected k-tuple is

$B=(n-k+1, n-k+2, \dots, n)$. We consider the redundant Prüfer codewords of those k-trees which are rooted at B and in which B has the degree $d \geqq 1$. Clearly B occurs in the lower row of the redundant Prüfer codeword d times, one of these being the last place. If the other $d-1$ places have the indices $j_1, j_2, \dots, j_{d-1}$ then clearly the elements a_{j_i+1} $(i=1,2,\dots,d-1)$ of the permutation π_{n-k} in the upper row of the redundant Prüfer codeword cannot be weak. It follows in the same way as in the proof of Lemma 6* that if both the places $j_1, \dots, j_{d-1}$ and the weak elements $a_{h_1+1}, \dots, a_{h_r+1}$ of π_{n-k} are fixed, the total number of choices of the k-tuples in the lower row is equal to

$$(5.9) \qquad k^r \prod_{\substack{h=1 \\ h \neq j_\ell \ (1 \leqq \ell \leqq d-1) \\ h \neq h_i \ (1 \leqq i \leqq r)}}^{n-k-1} (k(n-k-h)) \; .$$

(Notice that we had to replace $k(n-k-h)+1$ by $k(n-k-h)$ because B is now not an admissible choice for B_h if $h \neq j_\ell$ $j_\ell = 1,2,\dots,d-1$, and $h \neq n-k$.)

Multiplying (5.9) by the number $h_1 h_2 \dots h_r$ of permutations π_{n-k} having their weak elements at the places with indices h_i+1 $(1 \leqq i \leqq r)$ and by summation over all admissible choices of the indices $h_1, \dots, h_r$, we obtain that to any choice of the indices $j_1, \dots, j_{d-1}$ there correspond $[k(n-k)]^{n-k-d}$ k-trees; as the indices $j_1, \dots, j_{d-1}$ can be chosen in $\binom{n-k-1}{d-1}$ ways, (5.8) follows.

REMARK. Notice that

$$(5.10) \qquad R_k(n) = \sum_{d=1}^{n-k} R_{k,d}(n)$$

thus (5.3) can be deduced from (5.8); as a matter of fact Moon proved (5.3) by proving first (5.8) - in a rather complicated way - and then carrying out the summation over d.

One can easily generalize Theorem 1 to k-trees for any $k \geqq 2$. We call the point $j \leqq n-k$ of a labelled k-tree T of order n a weak point if there exists in T a point $h > j$ such that the unique "path" from the point h to the root (i.e. the shortest sequence of different complete $(k+1)$-graphs $C_1, \ldots, C_s$, the first of which contains the point h and the last the root, and the intersection of C_i and C_{i+1} is a complete k-graph) passes through the point j (i.e. j belongs to one of the $(k+1)$-tuples $C_1, \ldots, C_s$).

With this definition we obtain the following generalization of Theorem 1.

THEOREM 1*. Let $W_k(n,r)$ denote the probability that a random* labelled k-tree of order n, rooted at the k-tuple $(n-k+1, n-k+2, \ldots, n)$ contains r weak points. We have

$$\sum_{r=0}^{n-k-1} W_k(n,r) x^r = \prod_{h=1}^{n-k-1} \left(1 - \frac{hk(x-1)}{k(n-k)+1}\right).$$

Thus if $V_n^{(k)}$ denotes the number of weak points of a random k-tree of order n, we have

$$E(V_n^{(k)}) = \frac{k(n-k)(n-k-1)}{2[k(n-k)+1]} \sim \frac{n}{2}$$

and

$$D^2(V_n^{(k)}) = \frac{k(n-k)(n-k-1)[k(n-k+1)+3]}{6(k(n-k)+1)^2} \sim \frac{n}{6}$$

*A random k-tree of order n rooted at the k-tuple $n-k+1, \ldots, n$ means a k-tree chosen at random, with uniform distribution among the $[k(n-k)+1]^{n-k-1}$ trees.

further

$$\lim_{n \to +\infty} P\left(\frac{V_n^{(k)} - \frac{n}{2}}{\sqrt{\frac{n}{6}}} < x\right) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{x} e^{-\frac{u^2}{2}} du .$$

REMARK. Notice that asymptotically the mean and variance of $V_n^{(k)}$ do not depend on k. It has been proved by the second author (see [11]) that the same is true as regards the number $\varepsilon_n^{(k)}$ of endpoints of a random k-tree of order n.

$$\frac{\varepsilon_n^{(k)} - \frac{n}{e}}{\sqrt{\frac{n}{e}\left(1 - \frac{2}{e}\right)}}$$

has in the limit for $n \to +\infty$ a standard normal distribution independently of the value of k. Thus it follows that for the majority of labelled k-trees of order n the number of points which are neither endpoints nor weak points is asymptotically equal to $n\left(\frac{1}{2} - \frac{1}{e}\right)$.

Other enumeration problems concerning k-trees can also be solved by the method of the present paper.

REFERENCES

[1] F. HARARY-E.M. PALMER, On acyclic simplicial complexes, Mathematika 15 (1968) 115-122.

[2] A. PRÜFER, Neuer Beweis eines Satzes über Permutationen. Archiv der Mathematik und Physik 27 (1918) 142-144.

[3] F.N. DAVID-D.E. BARTON, Combinatorial chance, Griffin, London 1962, p. 181-182.

[4] See D.H. LEHMER, The machine tools of combinatorics, Chapter 1, in Applied Combinatorial Analysis, edited by F. Beckenbach, Wiley, New York, 1964, see p. 20.

[5] A. RÉNYI, Théorie des éléments saillants d' une suite d' observations, Annales de la Faculté des Sciences de l' Université de Clermont-Ferrand, 8 (1962) 7-13.

[6] L.W. BEINEKE-R.E. PIPPERT, The number of labelled k-dimensional trees, Journal of Combinatorial Theory, 6 (1969) 200-205.

[7] J.W. MOON, The number of labelled k-trees, Journal of Combinatorial Theory, 6 (1969) 196-199.

[8] C.L. MALLOWS-J.RIORDAN, The inversion enumerator for labelled trees, Bulletin of the American Math. Soc. 74 (1968) 92-94.

[9] J.W. MOON, Counting labelled trees: a survey of methods and results. (Lecture notes.) Biennial Seminar of the Canadian Mathematical Congress, Vancouver, 1968.

[10] A. RÉNYI, Some remarks on the theory of trees, Publications of the Math. Inst. of the Hung. Acad. of Sci. 4 (1959) 385-388.

[11] A. RÉNYI, On the number of endpoints of a k-tree, Studia Scientiarum Mathematicarum Hungarica (in print).