International Journal of Pure and Applied Mathematics
Volume 71 No. 1 2011, 49-56

A NOTE ON TREES AND CODES

Elena Rubei
Dipartimento di Matematica "U. Dini"
Viale Morgagni 67/A
50134 Firenze, ITALY

Abstract: We propose a code based on weighted trees and on the difficulty to reconstruct the weights of the tree from the distances $D_{i,j}$ among the leaves; precisely we consider an ordered rooted tree as code key; we put the letters of the message, translated into numbers, on the edges of the tree as their weights, by starting from the edges coming out from the root, then the edges coming out from the vertices of intrinsic distance 1 from the root and so on. Then we forecast the numbers $D_{i,j}$ for i, j leaves of the tree.

AMS Subject Classification: 05C05, 05C22, 94B25
Key Words: weighted trees, codes

1. Introduction

Let T be a positive-weighted tree, that is a tree such that every edge is endowed with a positive real number, called weight. If i and j are two leaves, we call $D_{i,j}(T)$ the distance beteween i and j, i.e. the sum of the weights of the edges of the path from i to j. The problem to reconstruct the positive-weighted tree from the $D_{i,j}$ has been widely studied. In particular we quote the Neighbourhood-Joining algorithm, which reconstructs a positive-weighted tree with n leaves in $O(n^3)$ elementary operations (see [1] and [2]).

In this short note we propose a code based on weighted trees and on the difficulty to reconstruct the weights of the tree from the distances among the leaves. Shortly speaking, the idea is the following: the code key is an ordered rooted tree (known only by the ally); we put the letters of the message, translated into numbers, on the edges of the tree as their weights, by starting from the edges coming out from the root, then the edges coming out from the vertices

Received: April 14, 2011

of intrinsic distance 1 from the root and so on. Then we forecast the numbers $D_{i,j}$ for any i, j leaves of the tree. The ally can easily reconstruct the message from the $D_{i,j}$ and the tree. For the enemy this is much more difficult (in §3 we specify how much more difficult). We can make the reconstruction of the weighted tree, and thus of the message, even more difficult for the enemy, by forecasting not all the $D_{i,j}$, but only some of them.

2. Preliminaries

Definition 1. For x, y vertices of a tree T, we call *intrinsic distance* of x and y the minimum of the following set:

$$\{\#edges\ of\ p |\ p\ path\ from\ x\ to\ y\}$$

We say that a vertex of a rooted tree is a *child* of a subgraph of the tree if the path from the vertex to the root intersects the subgraph.

A *weighted tree* is a tree such that every edge is endowed with a real number called "weight" or "length" of the edge. If the weights are positive we say that the tree is *positive-weighted*.

An *ordered rooted tree* is a rooted tree for which an ordering is specified for the children of each vertex with different degrees.

Definition 2. A *cherry* B in a tree T is a subtree B such that only one of the inner vertices is not bivalent; we call this vertex *stalk* of the cherry and we say that the leaves of a cherry are *neighbours*. We call the path from a leaf of a cherry to its stalk *twig* of this leaf.

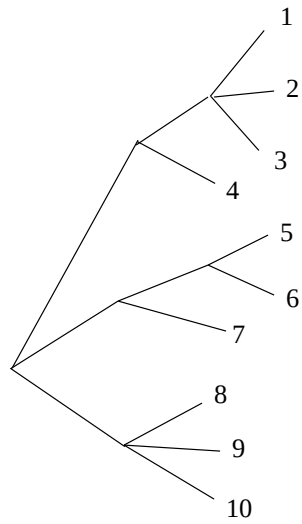
If T is a rooted tree, we define the *pseudocherries of distance* s in the following way: let r be the maximum of the intrinsic distances between the leaves and the root;

the pseudocherries of distance r are the cherries whose leaves have intrinsic distance r from the root

the pseudocherries of distance $r-1$ are, among the cherries of the tree you get by "pruning" the pseudocherries of distance r, the ones whose leaves have intrinsic distance $r-1$ from the root and so on.

If we have numbered the leaves of T, we name a pseudocherry C by taking, for every of its twig, the leaf of T which is child of C and has the lowest number.

Example.



The pseudocherries of distance 3 are: $(1, 2, 3)$, $(5, 6)$.

The pseudocherries of distance 2 are $(1, 4)$, $(5, 7)$, $(8, 9, 10)$.

The unique pseudocherry of distance 1 is $(1, 5, 8)$.

Remark 3. Let X be a finite sequence of natural numbers such that:

- the first number is greater or equal than 3

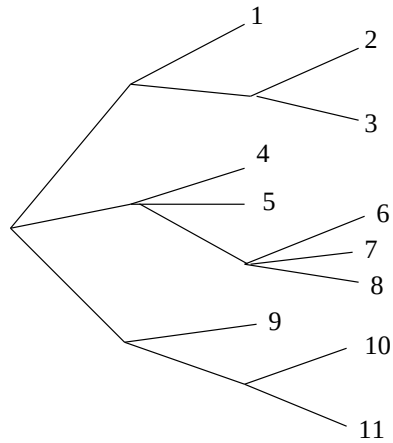
- X can be written as $X = (X^1, X^2, ..., X^r)$ where X^i are sequences of natural number, the length of X^1 is 1 and the length of X^i is equal to the sum of the elements of X^{i-1} for every $i \geq 2$.

Then we can associate to X an ordered rooted tree T in the following way: let X^1 be the number of the edges going out from the root; let X^2_1, X^2_2.... be the numbers of the edges going out from the vertices with intrinsic distance 1 from the root; let X^3_1, X^3_2.... be the numbers of the edges going out from the vertices with intrinsic distance 2 from the root and so on...

Viceversa to any ordered rooted tree T we can associate a sequence X as above.

We can number the leaves as in the following example, i.e. respecting the order of the ordered rooted tree.

Example. Let $X = (3, 2, 3, 2, 0, 2, 0, 0, 3, 0, 2)$. To be more clear, we could write it also as $X = (3|2, 3, 2|0, 2, 0, 0, 3, 0, 2)$, but the | are redundant (in fact the first | must be after the first number, the second | must be after further X^1 numbers and so on...). The corresponding tree is the one shown in the figure (where we have also numbered the leaves).



Remark 4. Let T be a tree with n leaves and k edges.
- If $n \geq 3$, then $k \geq n$.
- If there are no bivalent vertices, then $k \leq 2n-3$.
- If there are only 1 or 3-valent vertices, then $k = 2n-3$.

Proof. The three statements can be proved easily on induction on k. □

Remark 5. Let T be a rooted tree with n leaves. Then the number of the pseudocherries of T is less or equal than $n-2$.

Proof. Obviously we get the maximum number of pseudocherries when there are only 1-valent or 3-valent vertices. In this case, the number of the edges is $2n-3$ by the previous remark. So the number of the edges apart from the ones with the root as vertex is $2n-6$. Obviously the number of the pseudocherries, apart from the one with the root as stalk, is $\frac{1}{2}(2n-6) = n-3$. □

Remark 6. Let T be an ordered rooted tree without bivalent vertices and with n leaves. Then the length of the sequence X associated to T as in Remark 3 is less or equal than $2n-4$ (and this bound can be achieved).

Proof. Obviously the length x of X is

$$x = 1 + e - m$$

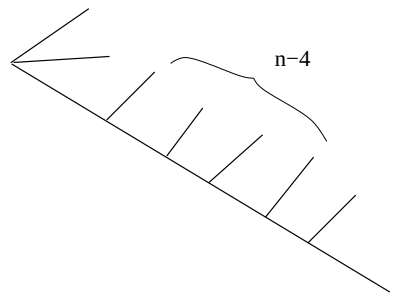
where e is the number of the edges and m is the number of the leaves with maximum intrinsic distance from the root.

We have

$$x \leq 1 + e - 2 \leq 2n - 3 + 1 - 2 = 2n - 4$$

where the first disequality is due to the fact that the number of the leaves with maximum intrinsic distance from the root is at least 2 and the second is due to Remark 4.

The sequence $(3|2,0,0|2,0|....|2,0)$, where $(2,0)$ is repeated $n-4$ times, achieves the bound, in fact its length is $2n-4$ and the corresponding tree has n leaves.

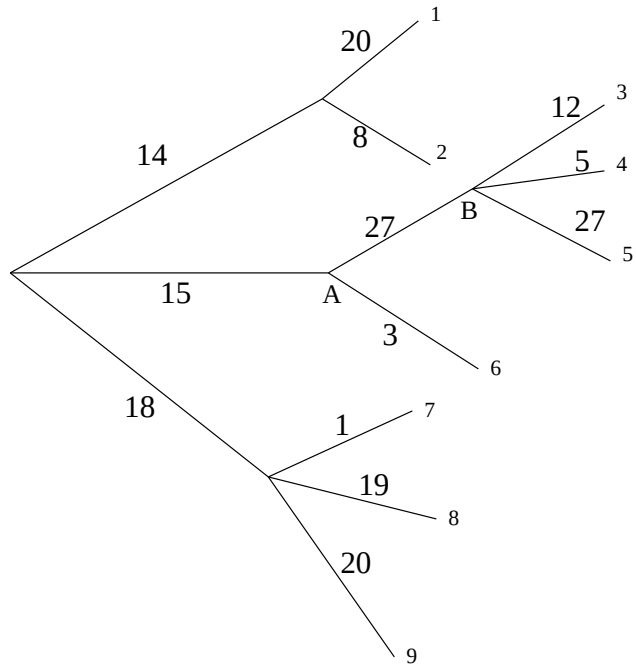


□

Remark 7. The number of ordered rooted trees with k edges, with only 3-valent and 1-valent vertices and such that they have at most one pseudocherry of distance t for any $t \in \mathbf{N}$, is $3 \cdot 2^{[\frac{k-5}{2}]}$.

3. The Code

The idea is the following. The code key is an ordered rooted tree T without bivalent vertices (known only by the ally); it can be given in the form of a sequence X of natural numbers as in Remark 3. For instance let $X = (3,2,2,3,0,0,3,0,0,0,0)$. We can translate the letters of a message into the numbers from 1 to 26 and the space into 27. For instance, if we want to send the message "north castle", first we translate it into numbers: $14,15,18,20,8,27,3,1,19,20,12,5$. Then we put these numbers on the edges of the tree by starting from the edges coming out from the root, then the edges coming out from the vertices of intrinsic distance 1 from the root and so on. If the length of the message is less than the number of the edges, we put 27 on the remained edges.



Then we forecast the numbers $D_{i,j}$ for i, j leaves of the tree. Since our ally knows the tree, it is easy for him to reconstruct the message, while for the enemy is more difficult.

Precisely we describe two possibilities: the second is less simple to be described, but better. We have decided to describe also the first, above all because we think that describing it can help to understand the idea and so also the second code, in other words, we describe the first code as "first step towords the construction" of the second.

1) Let the key code be an ordered rooted tree T without bivalent vertices and with n leaves and k edges. So the key code can be given as a sequence X of natural numbers of length at most $2n-4$ (see Remark 6).

We recall that $n \leq k \leq 2n-3$ by Remark 4; therefore k is $O(n)$.

We forecast all $D_{i,j}$ in lexicographic order with respect to the order $1 < 2... < n$. So the cardinality of the data we forecast is obviously $\left(\begin{array}{c} n \\ 2 \end{array} \right)$.

Evidently every time we can forecast a message of length less or equal than k.

We state that the number of the elementary operations the ally needs to reconstruct the message is $O(n)$. Let r be the maximum intrinsic distance of a leaf from the root. For every cherry $i_1, ..., i_s$, the sum of the weights of the

twigs i_l, i_m is D_{i_l,i_m} and the difference is $D_{i_l,j} - D_{i_m,j}$ for any other leaf j. Then, for every cherry, the ally can recover the weights of the twigs of the cherry with $O(\#twigs\ of\ the\ cherry)$ elementary operations. Analogously for the other pseudocherries, but, for a pseudocherry of distance $p < r$, to get the weights of the edges, further elementary operations are needed: for instance in the example above, from $D_{3,6}, D_{3,9}D_{6,9}$ one can get the distance from A to 3 and the weight of the edge $A, 6$; to get the weight of the edge A, B we need to subtract the weight of the edge $B, 3$ from the distance from A to 3. So to get the weights of the edges of the pseudocherries of distance $p < r$, the ally needs

$$O(\#twigs\ of\ the\ pseudocherries\ of\ distance\ p) + O(\#pseudocherries\ of\ distance\ p+1)$$

elementary operations. Therefore, in all, the ally needs

$$O(\#edges) + O(\#pseudocherries)$$

elementary operations, that is $O(n)$ elementary operations, by Remarks 6 and 5.

The number of the elementary operations the enemy needs to reconstruct the message is $O(n^3)$ by using neighbourhood joinig algorithm, see [1] and [2].

2) Let the key code be an ordered rooted tree X without bivalent vertices, with n leaves and k edges and with leaves 1 and n not in the same cherry.

We forecast for every pseudocherry $i_1, ..., i_s$ of maximum distance, the numbers $D_{i_1,n}, ..., D_{i_s,n}, D_{i_1,i_2}$ unless $\{i_1, ..., i_s\} \ni n$; in such a case we forecast $D_{i_1,1}, ..., D_{i_s,1}, D_{i_1,i_2}$. Analogously for the other pseudocherries (which we recall we name after their leaf with minimum number). Obviously we forecast every $D_{i,j}$ at most once. We forecast the $D_{i,j}$ in lexicographic order.

So, again, the key code is a sequence of length at most $2n - 4$ (see Remark 6), while the cardinality of the data we forecast is exactly k (which is $O(n)$). We can prove this for instance by induction on the number of cherries: if we add a cherry with s leaves $i_1, ..., i_s$, we increase the number of the edges by s and we have to forecast s numbers more: D_{i_1,i_2}, $D_{i_2,n}....D_{i_s,n}$.

Evidently every time we can forecast a message of length at most k.

As in 1), the number of the elementary operations the ally needs to reconstruct the message is $O(n)$.

As to the number of the elementary operations the enemy needs to reconstruct the message, we don't see other ways except making a list of the possible trees and then, supposing the tree is one after the other of the list, trying to reconstruct the message; but we recall that the number of the ordered rooted trees

with k edges is at least exponential in k, by Remark 7, and then exponential in n.

We resume the two possibilities in the following table:

	1	2
Length of the code key	$O(n)$	$O(n)$
Number of the data we have to forecast	$O(n^2)$	$O(n)$
Length of the message we can forecast	$O(n)$	$O(n)$
Number of operation for the ally	$O(n)$	$O(n)$
Number of the operation for the enemy	$O(n^3)$	at least exponential

Obviously we can use this method (both in the form 1 and 2) in two ways:

A) we can take a very large tree with a number of edges greater than the length of every message we have to sent and then forecast every message all at the same time.

B) we can take a smaller tree and divide every message in parts of length less than the number of the edges and then forecast them one after the other.

References

[1] M. Nei, N. Saitou, The neighbor joining method: A new method for reconstructing phylogenetic trees, *Mol. Biol. Evol.*, **4**, No. 4 (1987), 406-425

[2] J.A. Studier, K.J. Keppler, A note on the neighbor-joining algorithm of Saitou and Nei, *Mol. Biol. Evol.*, **5**, No. 6 (1988), 729-731.