Stud. Univ. Babeş-Bolyai Math. 59(2014), No. 4, 435–442

Reconstructibility of trees from subtree size frequencies

Dénes Bartha and Péter Burcsi

Abstract. Let T be a tree on n vertices. The subtree frequency vector (STF-vector) of T, denoted by $\mathrm{stf}(T)$ is a vector of length n whose kth coordinate is the number of subtrees of T that have exactly k vertices. We present algorithms for calculating the subtree frequencies. We give a combinatorial interpretation for the first few and last few entries of the STF-vector. The main question we investigate – originally motivated by the problem of determining molecule structure from mass spectrometry data – is whether T can be reconstructed from $\mathrm{stf}(T)$. We show that there exist examples of non-isomorphic pairs of unlabeled free (i.e. unrooted) trees that are STF-equivalent, i.e. have identical subtree frequency vectors. Using exhaustive computer search, we determine all such pairs for small sizes. We show that there are infinitely many non-isomorphic STF-equivalent pairs of trees by constructing infinite families of examples. We also show that for special kinds of trees (e.g. paths, stars and trees containing a single vertex of degree larger than 2), the tree is reconstructible from the subtree frequencies. We consider a version of the problem for rooted trees, where only subtrees containing the root are counted. Finally, we formulate some conjectures and open problems and outline further research directions.

Mathematics Subject Classification (2010): 05C05.

Keywords: Tree reconstruction, subtree size frequencies.

1. Introduction

Reconstruction of combinatorial structures from partial information is a widely discussed topic in the literature, full of intriguing and notoriously hard problems. Our present paper falls in the domain of reconstructibility investigations. Similar problems include reconstructibility of strings from factors or subsequences [2, 4], reconstructibility of graphs from vertex- or edge-deleted subtrees [6, 7], reconstructibility of matrices [3, 5], reconstruction of strings from Parikh vectors [1] and others.

This paper was presented at the 10th Joint Conference on Mathematics and Computer Science (MaCS 2014), May 21-25, 2014, Cluj-Napoca, Romania.

The problem we investigate is the possibility of reconstruction of an unlabeled free (i.e. unrooted) tree with n vertices, given the number of subtrees of size $1, 2, \ldots, n$, which we call the STF-vector of the tree. The motivation of the questions comes from the interpretation of mass spectrometry data.

The paper is structured as follows: in Section 2, we give the definition of the subtree frequency vector, and discuss some of its properties. In Section 3, we introduce methods for calculating the STF-vectors. Our two main tools are a version of the STF-vector for rooted trees and a polynomial representation of the STF-vector. In Section 4, we show that in some cases, the STF-vector uniquely determines the tree (up to isomorphism). In Section 5, we present examples where the STF information is insufficient for reconstructing the tree. In the Conclusion we present open questions and propose new research directions.

All symbols – if not stated otherwise – represent nonnegative integers, x is used for the variable of univariate polynomials and n usually denotes the number of vertices in a tree.

2. Basic definitions and properties

Definition 2.1. *Let T be a tree on n vertices. The subtree frequency vector (STF-vector) of T, denoted by* $\mathrm{stf}(T)$ *is a vector of length n whose entry at position k is the number of subtrees of T that have exactly k vertices.*

Remark 2.2. *Note that* stf *is clearly invariant for isomorphism. Thus in the reconstruction problem mentioned later, we are only interested in reconstructing the (unlabeled) tree up to isomorphism. Note however, that in the calculation of the STF-vector, all subtrees are considered, and isomorphic subtrees are counted with multiplicity.*

For example, if P_5 denotes a path of length 5 and S_4 a star with 4 leaves, then we have $\mathrm{stf}(P_5) = [6, 5, 4, 3, 2, 1]$ and $\mathrm{stf}(S_4) = [5, 4, 6, 4, 1]$.

Proposition 2.3. *Let T be a tree on n vertices with* $\mathrm{stf}(T) = [a_1, a_2, \ldots, a_n]$. *Then the following holds:*

i) $a_1 = n$,
ii) $a_2 = n - 1$,
iii) $a_3 = \sum_{v \in V} \binom{d(v)}{2}$, *where V is the set of vertices and $d(v)$ denotes the degree of v.*
iv) a_{n-1} *equals the number of leaves,*
v) $a_n = 1$.

Proof. For iii) note that a tree with 3 vertices is a path, so we can calculate such subtrees by counting how many of them are centered at each vertex of T, giving the formula. For iv) note that omitting a vertex v from T is connected if and only if v is a leaf. The other statements are trivial. □

We also introduce the rooted version of the STF-vector, partly because it is interesting on its own, but it also helps in calculating the unrooted STF-vector.

Definition 2.4. *Let T be a tree on n vertices and v a vertex of T. The rooted subtree frequency vector (RSTF-vector) of T with root v, denoted by* $\mathrm{rstf}(T, v)$ *is a vector of length n whose entry at position k is the number of subtrees of T that contain v and have exactly k vertices.*

For example if T is a path on 5 vertices, and v is its center, then $\mathrm{rstf}(T, v) = [1, 2, 3, 2, 1]$. If v' is a leaf in T, then $\mathrm{rstf}(T, v') = [1, 1, 1, 1, 1]$.

Proposition 2.5. *Let T be a rooted tree on n vertices, v the root of T, and for all vertices v' denote by $T_{v'}$ the subtree rooted at v'. Then* $\mathrm{stf}(T) = \sum_{v'} \mathrm{rstf}(T_{v'}, v')$.

Proof. Simply observe that each subtree has a unique node v' highest up in the tree, and is thus counted exactly once on the right side. □

3. Methods for calculating subtree frequencies

One possible solution to calculate the STF-vector of an unlabeled free (i.e. unrooted) tree with n vertices is to generate all the subtrees of the given tree and count their sizes in a vector. Since there can be exponentially many subtrees, this is not always applicable.

Another possibility is to use Proposition 2.5 and apply recursion. The problem then reduces to calculating RSTF-vectors for arbitrary $T_{v'}$ and v'. RSTF vectors can also be calculated using recursion. We could give a combinatorial description of the process, but it would be essentially equivalent to the polynomial method given below. We introduce polynomials for representing STF-vectors. It turns out that they are useful in both calculation of STF-vectors and in proving results about reconstructibility.

Definition 3.1. *Let T be a tree, v a vertex of T. Let* $\mathrm{stf}(T) = [a_1, a_2, \ldots, a_n]$ *and* $\mathrm{rstf}(T, v) = [b_1, b_2, \ldots, b_n]$. *The STF-polynomial of T, denoted by $s(T)$ is defined by* $s(T) = a_1 + a_2x + a_3x^2 + \cdots + a_nx^{n-1}$. *The RSTF-polynomial of T with root v, denoted by $r(T, v)$ is defined by* $r(T, v) = b_1 + b_2x + b_3x^2 + \cdots + b_nx^{n-1}$.

Remark 3.2. *Note that the degree k coefficient of the polynomial corresponds to the number of subtrees with k edges rather than k vertices and is a degree $n-1$ polynomial. This will yield simpler formulas later.*

We prove a few results which together allow a recursive calculation of s and r.

Lemma 3.3. *Let T_1, T_2 be rooted trees with roots v_1 and v_2 respectively. Let T be the rooted tree obtained by joining the two trees by identifying v_1 and v_2 as a new vertex v. Then* $r(T, v) = r(T_1, v_1)r(T_2, v_2)$.

Proof. A subtree of T containing v and exactly k edges is obtained by joining a a subtree of T_1 containing v_1 and i edges with a subtree of T_2 containing v_2 and j edges, where $i+j = k$. The number of such pairs is $\sum_{i+j=k} \mathrm{rstf}(T_1, v_1)[i+1] \cdot \mathrm{rstf}(T_2, v_2)[j+1]$, which is exactly the kth coefficient in the polynomial product. (We denote by $v[i]$ the ith component of vector v). □

Example 3.4. *Let T_1 and T_2 be paths of length 2 with v_1 and v_2 leaves of T_1 and T_2 respectively. Then T is a path of length 4 rooted at its center v. The polynomials $r(T_1, v_1) = r(T_2, v_2) = 1 + x + x^2$, while $r(T, v) = 1 + 2x + 3x^2 + 2x^3 + x^4 = r(T_1, v_1)r(T_2, v_2)$.*

Lemma 3.5. *Let T_1, T_2 be rooted trees with roots v_1 and v_2 respectively. Let T be the rooted tree obtained by joining the two trees by identifying v_1 and v_2 as a new vertex v. Then $s(T) = r(T_1, v_1)r(T_2, v_2) + s(T_1) - r(T_1, v_1) + s(T_2) - r(T_2, v_2)$.*

Proof. Observe that a subtree not containing v is either a subtree of T_1 not containing v_1, or a subtree of T_2 not containing v_2. The number of such subtrees is counted by the polynomials $s(T_1) - r(T_1, v_1)$ and $s(T_2) - r(T_2, v_2)$, respectively. This gives the desired result. □

This latter statement allows one to calculate r and s of a rooted tree recursively if the root is not a leaf. Take the subtrees that are obtained by taking the root and all nodes below one child of the root, and join them at the root. Since these subtrees are smaller than the original tree, the calculation can proceed recursively using the following proposition. Note that the base cases of the recursion are trees with 1 or 2 vertices for which the calculation is trivial.

Lemma 3.6. *Let T be a tree and v a leaf of T. Denote by v' the only neighbor of v and by T' the subtree obtained by removing v. Then $r(T, v) = 1 + xr(T', v')$ and $s(T) = s(T') + r(T)$.*

Proof. Apart from the single-node subtree consisting of v itself, all subtrees containing v also contain v', and such subtrees of T of size k are in bijection with subtrees of T' containing v' and of size $k-1$. This proves the first statement. The second statement is trivial. □

4. Reconstructibility results

We present a few results which show that in some cases, $\mathrm{stf}(T)$ uniquely determines T. The first two are trivial observations, the third one requires deeper analysis.

Proposition 4.1. *If $\mathrm{stf}(T) = [n, n-1, \ldots, 1]$ for some n, then T is a path on n vertices.*

Proof. From the vector we deduce that the tree has n vertices and that it contains $\binom{n}{2}$ subtrees. Every tree on n vertices contains at least $\binom{n}{2}$ subtrees, namely the paths between pairs of vertices. The only tree that does not contain any further subtrees is a path. □

Proposition 4.2. *Let S_k be a star with $k \geq 2$ leaves. If $\mathrm{stf}(T) = \mathrm{stf}(S_k)$, then T is isomorphic to S_k.*

Proof. By Proposition 2.3, the number of vertices is $k+1$ and the number of leaves is k, which implies the claim. □

Definition 4.3. *Let $a_1, a_2, \ldots, a_k \geq 1$ and let $SL(a_1, a_2, \ldots, a_k)$ denote the star-like graph obtained by joining paths of length $a_1, a_2, \ldots, a_k$ at their endpoints. See Figure 5 for an illustration.*

Theorem 4.4. *Let $k, l \geq 3$, and $1 \leq a_1 \leq a_2 \leq \ldots \leq a_k$, $1 \leq b_1 \leq b_2 \leq \ldots \leq b_l$. If $\mathrm{stf}(SL(a_1, a_2, \ldots, a_k)) = \mathrm{stf}(SL(b_1, b_2, \ldots, b_l))$, then $k = l$ and $a_i = b_i$ for $i = 1, 2, \ldots, k$.*

Proof. Let $T_1 = SL(a_1, \ldots, a_k)$ and $T_2 = SL(b_1, \ldots, b_k)$. By Proposition 2.3, the number of leaves is the same in the two graphs, which implies $k = l$. By Lemma 3.5 we obtain for the polynomials (which by the conditions are equal):

$$\begin{aligned} s(T_1) &= \prod_{i=1}^{k}(1 + x + x^2 + \cdots + x^{a_i}) + \sum_{i=1}^{k}\left(a_i + (a_i - 1)x + \cdots + x^{a_i - 1}\right) \\ s(T_2) &= \prod_{i=1}^{k}(1 + x + x^2 + \cdots + x^{b_i}) + \sum_{i=1}^{k}\left(b_i + (b_i - 1)x + \cdots + x^{b_i - 1}\right) \end{aligned}$$

Assume by contradiction that for some i, $a_i \neq b_i$ holds and i is the smallest such index. Denote by c the constant term in the polynomials and note that $c = a_1 + a_2 + \cdots + a_k + 1 = b_1 + b_2 + \cdots + b_k + 1$, thus $i < k$. Wlog., we may assume $a_i < b_i$. Compare the coefficients of degree $c - a_i - 2$ in the expansion of the two polynomials. Note that $c - 1$ is the degree of the polynomials. The sums $a_i + (a_i - 1)x + \cdots + x^{a_i - 1}$ do not contribute to this term (because $k \geq 3$ and $i < k$), so we only have to compare the expansion of the products. The expansion of the product gives a reciprocal polynomial, thus it is enough to show that the degree $a_i + 1$ term differs in the products. This coefficient can be calculated if we consider the products modulo $x^{a_i + 2}$. Then the first $i - 1$ factors coming from $s(T_1)$ and $s(T_2)$ are identical, but in the ith factor, $s(T_2)$ has the additional term $x^{a_i + 1}$, which contributes to the product. For the remaining factors, $s(T_2)$ has always at least as many terms as $s(T_1)$. □

A similar statement holds for rooted STF-vectors which, however, is easier to prove.

Proposition 4.5. *Let $k, l \geq 3$, and $1 \leq a_1 \leq a_2 \leq \ldots \leq a_k$, $1 \leq b_1 \leq b_2 \leq \ldots \leq b_l$. Let $T_1 = SL(a_1, a_2, \ldots, a_k)$ and $T_2 = SL(b_1, b_2, \ldots, b_l)$, with the vertices of degree larger than 2: v_1 and v_2 as roots. If $r(T_1, v_1) = r(T_2, v_2)$, then $k = l$ and $a_i = b_i$ for $i = 1, 2, \ldots, k$.*

Proof. We have

$$\begin{aligned} f = r(T_1, v_1) &= \prod_{i=1}^{k}(1 + x + x^2 + \cdots + x^{a_i}) \\ g = r(T_2, v_2) &= \prod_{i=1}^{l}(1 + x + x^2 + \cdots + x^{b_i}) \end{aligned}$$

Assume by contradiction that $a_k \neq b_l$, say $a_k > b_l$. If we look at the polynomials as complex polynomials, then a primitive (a_k)th root of unity is a root of f but not a

root of g. So $a_k = b_l$, and a primitive (a_k)th root of unity is a root of both f and g. We deduce that the factor $(1 + x + \cdots + x^{a_k})$ is present in both products. After simplifying, we proceed by induction on $\max(k, l)$ and the claim follows. □

5. STF-equivalent trees

Definition 5.1. *We say that trees T_1 and T_2 are STF-equivalent if* $\mathrm{stf}(T_1) = \mathrm{stf}(T_2)$.

In this section we consider non-isomorphic STF-equivalent trees. We performed computer experiments in order to determine STF-equivalent pairs of trees for $n \leq 21$ (n is the number of vertices). We found that for $n \leq 9$ no such pairs exist and for $10 \leq n \leq 21$, there always exist non-isomorphic STF-equivalent trees. This means that in general, unique reconstruction from STF-vectors is impossible. We show the computational results in Table 1.

n	#trees	#classes	largest class	#dog's bone
0	1	0	0	0
⋮	⋮	⋮	⋮	⋮
9	47	0	0	0
10	106	1	2	1
11	235	4	2	0
12	551	5	2	1
13	1301	12	2	1
14	3159	32	2	0
15	7741	62	2	0
16	19320	139	3	3
17	48629	298	3	0
18	123867	649	3	0
19	317955	1441	4	2
20	823065	3330	3	2
21	2144505	7932	4	0
22	5623756	?	?	3
24	39299897	?	?	2
25	104636890	?	?	3
28	2023443032	?	?	7
31	40330829030	?	?	4

Table 1. The number of STF-equivalence classes containing at least two trees and the maximal size of a class for $n \leq 21$. The last column shows the number of classes that contain a special kind of graph which we call dog's bone graphs – all such examples are shown for $n \leq 31$.

Based on computational investigation, we tried to construct general examples of non-isomorphis STF-equivalent pairs. We present two infinite families of non-isomorphic STF-equivalent pairs, showing that for sizes $n = 3k + 1$, there always

exist such pairs. We introduce a notation for a special kind of graph, which – based on its shape – we call dog's bone graphs.

Definition 5.2. *Let $a, b, c, d, e \geq 1$. The dog's bone tree $DB(a, b, c, d, e)$ is a tree that contains two vertices v, v' of degree 3 connected by a path of length c, and two paths of length a and b starting at v, and two other paths of length d and e starting at v'. See Figure 5 for an illustration.*

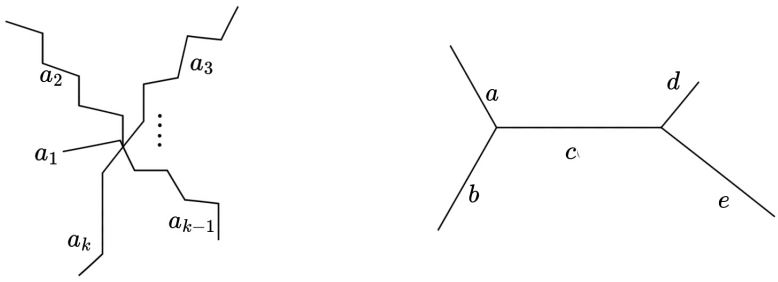

Figure 1. On the left: the star-like graph $SL(a_1, a_2, \ldots, a_k)$.
On the right: the dog's bone $DB(a, b, c, d, e)$.

Theorem 5.3. *Let $k \geq 1$. The trees $T_1 = DB(k, 2k+1, 1, k, 2k+1)$ and $T_2 = DB(k, k, 1, 2k, 2k+2)$ are STF-equivalent.*

Proof. Using Lemma 3.5, and applying summation for geometric series, after some calculation we have the following polynomials $f = s(T_1), g = s(T_2)$.

$$f = \frac{x\left(x^{k+1}-1\right)^2\left(x^{2k+2}-1\right)^2}{(x-1)^4} + 2\left(\frac{x\left(x^{3k+2}-1\right)}{x-1} - 3k - 2\right)(x-1)^{-1}$$

$$g = \frac{x\left(x^{k+1}-1\right)^2\left(x^{2k+1}-1\right)\left(x^{2k+3}-1\right)}{(x-1)^4}$$
$$+ \left(\frac{x\left(x^{2k+1}-1\right)}{x-1} + \frac{x\left(x^{4k+3}-1\right)}{x-1} - 6k - 4\right)(x-1)^{-1}$$

Their equality would be tedious to check by hand, but can readily be verified on a computer algebra system: if we replace all occurrences of x^k by a new variable y, then the difference of the resulting bivariate polynomials simplifies to 0. □

The following theorem can be proved similarly.

Theorem 5.4. *Let $k \geq 1$. The trees $T_1 = DB(k, 2k+2, 1, k+1, 2k+2)$ and $T_2 = DB(k, k+1, 1, 2k+1, 2k+3)$ are STF-equivalent.*

Corollary 5.5. *There exist non-isomorphic pairs of STF-equivalent trees for $6k+1$ and $6k+4$ vertices, for any $k \geq 1$.*

6. Summary and further work

In this paper we introduced the concept of STF-vectors and investigated the problem of reconstructibility of trees from subtree frequencies. We pose some open questions.

- Find more families of non-isomorphic STF-equivalent pairs and prove that such pairs exist for all $n \geq 10$.
- Find more types of graphs which are reconstructible from their STF-vectors.
- Are STF-equivalence class sizes unbounded as n grows?
- Calculate the STF-vector of a tree together with all RSTF-vectors. Are these $n+1$ vectors already sufficient for reconstruction up to isomorphism?
- Investigate the relationship of STF-vectors with other graph invariants, e.g. spectrum.

Besides these, our ongoing research will mainly focus on the labeled version of the problem, where each vertex or edge of the tree has a label from a finite set of colors.

Acknowledgement. The research of the second author was partially supported by a special contract No. 18370-9/2013/TUDPOL with the Ministry of Human Recources.

References

[1] Acharya, J., Das, H., Milenkovic, O., Orlitsky, A., Pan, S., *String reconstruction from substring compositions*, arXiv:1403.2439v1.

[2] Dudik, J., Schulman, L.J., *Reconstruction from subsequences*, J. Combin. Theory Ser. A, **103**(2003), no. 2, 337–348.

[3] Kós, G., Ligeti, P., Sziklai, P., *Reconstruction of matrices from submatrices*, Math. Comput., **78**(2009), 1733–1747.

[4] Krasikov, I., Roddity, Y., *On a reconstruction problem for sequences*, J. Combin. Theory Ser. A, **77**(1997), no. 2, 344–348.

[5] Manvel, B., Stockmeyer, P.K., *On reconstruction of matrices*, Math. Mag., **44**(1971), no. 4, 218–221.

[6] Manvel, B., *Reconstruction of trees*, Canad. J. Math., **22**(1970), 55–60.

[7] Manvel, B., *On reconstructing graphs from their sets of subgraphs*, J. Combin. Theory Ser. B, **21**(1976), 156–165.

Dénes Bartha
Eötvös Loránd University
Department of Computer Algebra
Budapest, Hungary
e-mail: denesb@gmail.com

Péter Burcsi
Eötvös Loránd University
Department of Computer Algebra
Budapest, Hungary
e-mail: bupe@compalg.inf.elte.hu