Sum labelling of trees

(Working draft)

Mirka Miller[1,4] Joe Ryan[2] Zdeněk Ryjáček[3,4]

July 31, 2002

Keywords:
AMS Subject Classification (2000):

1 Introduction

Graph $G = (V(G), E(G))$: finite, undirected, simple
$d_G(x)$ - the degree of a vertex; $d_G(x) = 0$ - isolated vertex, minimum degree $\delta(G)$;
tree, branch of a tree at a vertex;
for notation not defined here we refer to [1]
the Depth-First-Search (DFS) algorithm (or a reference ???)
introduction to the problem of sum graphs,
definition of $\sigma(G)$
known results
M.N. Ellingham [2] proved the following result.

Theorem A [2]. *If T is a tree of order at least 2, then*

$$\sigma(T) = 1.$$

However, the proof of this result is relatively complicated.

[1]School of Electrical Engineering and Computer Science, The University of Newcastle, NSW 2308, Australia, email mirka@cs.newcastle.edu.au
[2]Newcastle Graduate School of Business, The University of Newcastle, NSW 2308, Australia, email joe.ryan@newcastle.edu.au
[3]Department of Mathematics, University of West Bohemia, and Institute of Theoretical Computer Science (ITI), Charles University, P.O. Box 314, 306 14 Pilsen, Czech Republic, e-mail ryjacek@kma.zcu.cz
[4]Research supported by grant ...

2 Sum labelling of trees

In this section of the paper, we present a simple algorithm for sum labelling of trees with one isolate (note that by ... such algorithm is optimal). The algorithm follows the well-known Depth-First-Search (DFS) algorithm with one modification: whenever the algorithm finds a new vertex $x \in V(T)$ of a tree T having degree $d_T(x) \geq 3$, it first visits all its neighbors of degree 1 (if any), before proceeding to branches of T at x of higher order. This algorithm will be referred to as the Modified Depth-First-Search (MDFS) algorithm (in fact, the MDFS is just one of the possible implementations of the DFS, in which edges with vertices of degree 1 are given higher priority).

The algorithm for sum labelling of trees follows the ordering of vertices given by the MDFS, starts with labels 1 and 2 at a vertex of degree 1 and its neighbor, and at each step assigns the "new found vertex" a label that is equal to the sum of labels of the (only) edge that has at that point no working vertex. Finally, the sum of labels of the last edge is assigned to the only isolated vertex. The algorithm gives an alternative (and simpler) proof of Theorem A.

Example 1. An example of a tree labelled with this algorithm is shown in Figure 1.

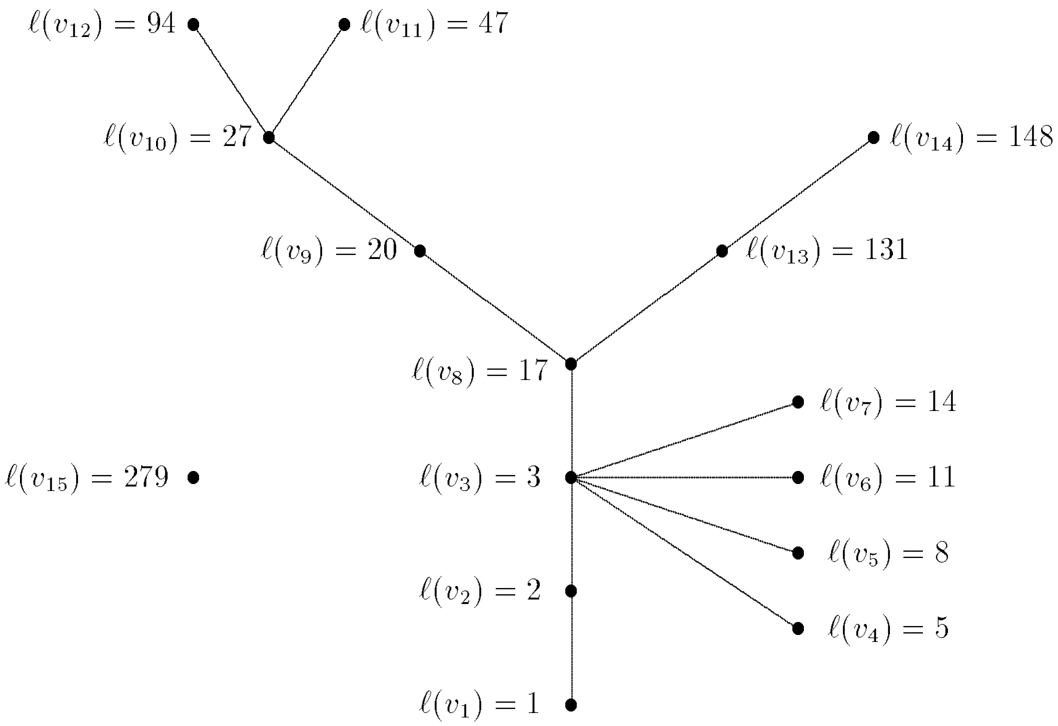

Figure 1

The labelling algorithm can be described in a more precise (and formal) way as follows.

Algorithm 2. (Sum labelling of trees)
Let T be a tree.

1. Let $v_1, \dots, v_n$ be vertices of T, numbered in the order of the MFDS algorithm, started at an (arbitrary) vertex $v_1 \in V(T)$ of degree $d_T(v_1) = 1$.

2. Set $T^* = (V(T) \cup \{v_{n+1}\}, E(T))$

   (i.e., T^* is obtained from T by adding an isolated vertex v_{n+1}).

3. Let A be the set of all vertices that are numbered in a forward-step that follows immediately after a back-step of the MFDS-algorithm. (In the tree of Figure 1, $A = \{v_5, v_6, v_7, v_8, v_{12}, v_{13}.\}$)

   Set $I_A = \{i \in \langle 1, n+1 \rangle \mid v_i \in A\}$.

4. For any $i \in \langle 2, n+1 \rangle \setminus I_A$ set $v_i^- = v_{i-1}$. For $i \in I_A$ let v_i^- denote the only neighbor v_j of v_i with $j < i$ (i.e., v_i^- is the only neighbor of v_i that, at the time v_i being numbered by the MDFS, had been already assigned a value v_j).

   For example, in Figure 1, $v_4^- = v_5^- = v_6^- = v_7^- = v_8^- = v_3$, $v_9^- = v_{13}^- = v_8$ and $v_{11}^- = v_{12}^- = v_{10}$.

5. Set $A^+ = \{x \in V(T) \mid x^- \in A\}$ (note that possibly $A \cap A^+ \neq \emptyset$).

   In Figure 1, $A^+ = \{v_6, v_7, v_8, v_9, v_{14}\}$.

6. For $i = 1, \dots, n+1$ define the labelling $\ell(v_i)$ as follows:

   (a) $\ell(v_1) = 1$, $\ell(v_2) = 2$,

   (b) for $i = 3, \dots, n+1$ set $\ell(v_i) = \ell(v_{i-1}) + \ell(v_{i-1}^-)$.

Remark 3. Note that if T is a path, then Algorithm 2 assigns the vertices of T the Fibonacci numbers.

Theorem 4. *Let T be a tree and let $\ell(v_i)$ be a labelling of its vertices obtained by Algorithm 2. Then, for any $x, y \in V(T^*)$, $xy \in E(T^*)$ if and only if there is a $z \in V(T^*)$ such that $\ell(x) + \ell(y) = \ell(z)$.*

Proof. Clearly, for any $xy \in E(T)$ there is a $z \in V(T^*)$ such that $\ell(x) + \ell(y) = \ell(z)$. It "remains" to prove the converse. It is sufficient to show that, for any $z = v_i \in V(T^*)$, $x = v_{i-1}$ and $y = v_{i-1}^-$ is the only pair of vertices of T^* with $\ell(x) + \ell(y) = \ell(z)$. Thus, let $z = v_i \in V(T^*)$ and let $x, y \in V(T^*)$ be such that $\ell(x) + \ell(y) = \ell(z)$.

1. The statement is obvious for $z \in \{v_1, v_2\}$.

2. Let $z = v_i$, $3 \leq i \leq n+1$, be such that $z \in V(T^*) \setminus A^+$. Then $v_{i-1}^- = v_{i-2}$, implying

$$\ell(v_i) = \ell(v_{i-1}) + \ell(v_{i-2}). \tag{1}$$

First note that, by the definition of the labelling ℓ, we have $\ell(v_{j-1}) < \ell(v_j)$ for any j, $2 \leq j \leq n+1$. Suppose that

$$\ell(v_i) = \ell(v_s) + \ell(v_t) \tag{2}$$

for some s, t, $1 \leq s < t \leq n+1$.

- If $t \geq i$, then $\ell(v_t) \geq \ell(v_i)$, contradicting (2) (since $\ell(v_s) > 0$).
- If $t \leq i-2$, then $s < t \leq i-2$ implies $i \geq 4$ and $\ell(v_s) + \ell(v_t) \leq \ell(v_{i-3}) + \ell(v_{i-2}) < \ell(v_{i-2}) + \ell(v_{i-1}) = \ell(v_i)$, contradicting (2).
- Hence $t = i-1$ and from (1) we get $s = i-2$.

This shows that v_{i-2}, v_{i-1} is the only pair of vertices satisfying (1) in this subcase.

3. Finally, let $z = v_i$, $3 \leq i \leq n+1$, be such that $z = v_i \in A^+$. Then

$$\ell(v_i) = \ell(v_{i-1}) + \ell(v_{i-1}^-), \tag{3}$$

where $\ell(v_{i-1}^-) = \ell(v_j)$ for some j, $2 \leq j \leq i-3$. Suppose again that

$$\ell(v_i) = \ell(v_s) + \ell(v_t) \tag{4}$$

for some s, t, $1 \leq s < t \leq n+1$. Similarly as in the previous subcase, we can suppose that $t \leq i-2$ (since $t \geq i$ yields a contradiction and for $t = i-1$ we are done).

On the other hand, $s \leq j$ implies $\ell(v_s) + \ell(v_t) \leq \ell(v_j) + \ell(v_{i-2}) < \ell(v_{i-1}^-) + \ell(v_{i-1}) = \ell(v_i)$, contradicting (3). Hence we have $j < s < t < i-2$.

Suppose that v_j has no neighbors of degree 1. Then $\ell(v_i) = \ell(v_{i-1}) + \ell(v_j)$, where $\ell(v_{i-1}) = \ell(v_{i-2}) + \ell(v_{i-3})$. But then $s < t \leq i-2$ implies $\ell(v_i) = \ell(v_s) + \ell(v_t) \leq \ell(v_{i-3}) + \ell(v_{i-2}) = \ell(v_{i-1}) < \ell(v_i)$, contradicting (4).

Hence v_i has neighbors of degree 1. By the MDFS algorithm, the neighbors of degree 1 of v_j are numbered first. Thus, let k, $j \leq k \leq i-2$, be the largest integer for which $v_j v_k \in E(T)$ and $d_T(v_k) = 1$.

Suppose first that $i = k+2$. Then, by the rules of Algorithm 2, we have

$$\ell(v_i) \equiv \ell(v_{j-1}) \equiv \ell(v_s) \equiv \ell(v_t) \pmod{\ell(v_j)}.$$

Hence there are integers k_s, k_t, k_i such that

$$\begin{aligned} \ell(v_s) &= \ell(v_{j-1}) + k_s \ell(v_j), \\ \ell(v_t) &= \ell(v_{j-1}) + k_t \ell(v_j), \\ \ell(v_i) &= \ell(v_{j-1}) + k_i \ell(v_j). \end{aligned}$$

¿From this and from (4) we then get

$$\ell(v_{j-1}) + k_i \ell(v_j) = \ell(v_i) = 2\ell(v_{j-1}) + (k_s + k_t)\ell(v_j),$$

from which

$$(k_i - k_s - k_t)\ell(v_j) = \ell(v_{j-1}),$$

contradicting the fact that $0 < \ell(v_{j-1}) < \ell(v_j)$.

Hence we have $i \geq k+3$, but then, similarly as in the previous subcase, $\ell(v_i) = \ell(v_{i-1}) + \ell(v_j)$ and $\ell(v_{i-1}) = \ell(v_{i-2}) + \ell(v_{i-3})$, implying

$$\ell(v_i) = \ell(v_s) + \ell(v_t) \leq \ell(v_{i-3}) + \ell(v_{i-2}) = \ell(v_{i-1}) < \ell(v_i),$$

a contradiction. ■

References

[1] Bondy, J.A.; Murty, U.S.R.: Graph theory with applications. Macmillan, London and Elsevier, New York, 1976.

[2] Ellingham, M.N.: Sum graphs from trees. Ars Combin. 35 (1993), 335-349.