PUBLICATIONS DE L'INSTITUT MATHÉMATIQUE
Nouvelle série, tome 79(93) (2006), 1–12 DOI: 10.2298/PIM0693001M

MORE ABOUT SINGULAR LINE GRAPHS OF TREES

M. C. Marino, I. Sciriha, S. K. Simić, and D. V. Tošić

Communicated by Žarko Mijajlović

ABSTRACT. We study those trees whose line graphs are singular. Besides new proofs of some old results, we offer many new results including the computer search which covers the trees with at most twenty vertices.

1. Introduction

Line graphs, and more generally, generalized line graphs, represent the class of graphs with several remarkable spectral properties. Recall first that their least eigenvalue is greater than or equal to -2 (see, for example, [**3**] for more details). Next, it is worth mentioning that 0 (or -1), can be, in many instances, the eigenvalue of these graphs. One reason for these phenomena is the presence of the so called *duplicate* (or *co-duplicate*) vertices. (Recall, following [**4**], that two vertices with the same open (closed) neighbourhood are called duplicate (resp. co-duplicate) vertices.) It is also noteworthy that the numbers 0 and -1 (as the eigenvalues of graphs) have a special role in spectral graph theory (see, for example, [**2**, Chapter 7]). In addition, graphs having 0 as an eigenvalue, i.e., *singular graphs*, are significant in mathematical chemistry (see, for example, [**1**]).

Given a simple graph H, its *line graph* (denoted by $L(H)$) is a graph G whose vertex set is equal to the edge set of H, with two vertices in G being adjacent if the corresponding edges in H are adjacent (i.e., have a common vertex). If $G = L(H)$, then H is called a *root graph* of G.

A *generalized line graph* can be viewed as the line graph of some special type of multigraphs. A double hanging edge (at some vertex) is called a *petal.* A (simple) graph with petals attached at its vertices is, in fact, a *root graph* of a generalized line graph (denoted by $\hat{H}$). We now emphasize only that two edges in the root

2000 *Mathematics Subject Classification*: Primary 05C50.
Key words and phrases: trees, line graphs, generalized line graphs, singular graphs.
The second author was supported by the University of Malta and the Italian Cultural Center in Malta. The third and fourth authors were supported by the Serbian Ministry of Science.

graph are adjacent if they have exactly one common vertex (so that edges from the same petal are non-adjacent). We denote by $L(\hat{H})$ the generalized line graph of $\hat{H}$.

EXAMPLE 1.1. In Fig. 1 we show the construction of a generalized line graph from its root multigraph. (Note that only two vertices of the root graph have petals).

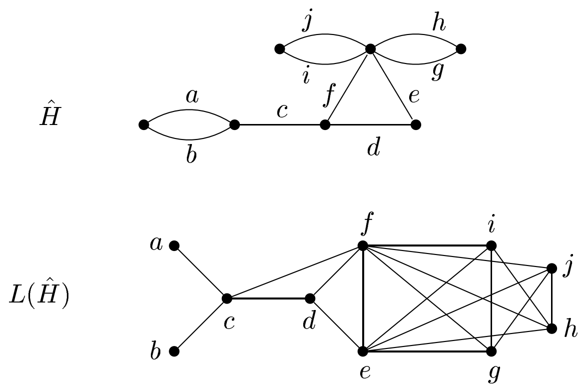

FIGURE 1. Construction of a generalized line graph from its root multi-graph.

We will consider generalized line graphs in Section 4.

For line graphs of trees, duplicate vertices can appear only if such a tree is equal to P_4. (More generally, if a connected graph is a line graph, then it can have duplicate vertices only if its root graph has P_4 as its spanning subgraph.) Therefore, there remains to consider line graphs of trees, which have 0 as an eigenvalue, i.e., which are singular.

Singular line graphs of trees were already studied in literature. I. Gutman and I. Sciriha (see [**5**]) were the first who proved that singular line graphs of trees have 0 as a simple eigenvalue, i.e., of multiplicity (or nullity) one. Other proofs of this theorem, and related results, are provided by I. Sciriha (see [**7, 8**]). Here, we will offer yet another proof of this interesting theorem and, in addition, some other results.

For any graph, say G, let $\mathcal{A}(G)$ be its adjacency matrix, while $\mathcal{B}(G)$ its (vertex to edge) incidence matrix. Consider now the graph H, in the role of the root graph of G (so that $L(H) = G$). Then

$$\mathcal{B}(H)\mathcal{B}(H)^T = \mathcal{A}(H) + \mathcal{D}(H), \quad \mathcal{B}(H)^T\mathcal{B}(H) = \mathcal{A}(G) + 2\mathcal{I}.$$

Here, $\mathcal{D}(H)$ is a diagonal matrix with (vertex) degrees along the diagonal.

The *eigenvalues* of a graph G are the eigenvalues of its adjacency matrix and are said to form the spectrum of G. So G is *singular* if its adjacency matrix is singular, or equivalently, if 0 is an eigenvalue of G. H is an *L-singular* graph (or, an LS graph for short) if $L(H)$ is singular.

A λ-eigenvector of a (labelled) graph G is an eigenvector of G corresponding to the eigenvalue equal to λ; the λ-eigenspace of G is the corresponding eigenspace. A 0-eigenvector will be also called a *kernel eigenvector*.

If S is a finite set with $|S| = s$, then the s-dimensional vector space over $\mathbb{R}$ will be denoted by $\mathbb{R}^S$. The elements in $\mathbb{R}^S$ can be interpreted either as vectors (column matrices), or as functions from S to $\mathbb{R}$. We will make use of this fact interchangeably. To any (labelled) graph $G = (V(G), E(G))$ we associate the following two vector spaces:

$$\mathbb{R}^{V(G)} = \{\mathbf{x} \mid \mathbf{x} : V(G) \to \mathbb{R}\} \text{ (vertex space);}$$
$$\mathbb{R}^{E(G)} = \{\hat{\mathbf{x}} \mid \hat{\mathbf{x}} : E(G) \to \mathbb{R}\} \text{ (edge space).}$$

Given a vector $\mathbf{x} \in \mathbb{R}^{V(G)}$ ($\hat{\mathbf{x}} \in \mathbb{R}^{E(G)}$) then $\mathbf{x}(v)$ ($\hat{\mathbf{x}}(e)$) is interpreted as the weight of a vertex v (edge e) of G with respect to $\mathbf{x}$ (resp. $\hat{\mathbf{x}}$).

Assume now that $G = L(H)$. In the sequel, let $\mathcal{E}_0(G) \subseteq \mathbb{R}^{V(G)}$ be the 0-eigenspace, or the null-space, of G. So $\mathcal{E}_0(G) = \{\mathbf{x} \mid \mathcal{A}(G)\mathbf{x} = \mathbf{0}\}$. In view of how line graphs are defined, we can now consider the vector space $\hat{\mathcal{E}}(H) \subseteq \mathbb{R}^{E(H)}$, in which $\hat{\mathbf{x}} \in \hat{\mathcal{E}}(H)$ if and only if $\mathbf{x} \in \mathcal{E}_0(G)$, where $\hat{\mathbf{x}}(e) = \mathbf{x}(v)$ for any edge e of H and the corresponding vertex v of G. Clearly, $\mathcal{E}_0(G) \cong \hat{\mathcal{E}}(H)$.

Let $\mathbf{x}$ be a λ-eigenvector of G ($= L(H)$), and $\hat{\mathbf{x}}$ the vector related to $\mathbf{x}$ as above. Then we have

$$\lambda x(u) = \sum_{v \sim u} x(v) \quad u \in V(G),$$

where the summation is taken over all vertices v adjacent to u in G. Equivalently,

$$\lambda \hat{x}(e) = \sum_{f \sim e} \hat{x}(f) \quad e \in E(H),$$

where the summing is taken over all edges f adjacent to e in H. In addition, (u, e) (and (v, f)) are the pairs of vertices and edges (from G and H, respectively) which correspond to one another when the line graph is being constructed. In particular, if G is singular then

$$\sum_{v \sim u} x(v) = 0 \quad u \in V(G),$$

or equivalently, if H is an LS graph then

$$\sum_{f \sim e} \hat{x}(f) = 0 \quad e \in E(H).$$

Furthermore, since vector spaces $\mathcal{E}_0(G)$ and $\hat{\mathcal{E}}(H)$ are isomorphic we will sometimes drop the symbol $\hat{}$ (and the names of the graphs) from our notation. The basic terminology follows [**6**] (for graph spectra, the reader is referred to [**1**]).

The plan of the paper is as follows: in Section 2, we give some basic results including new (and shorter) proofs of some basic theorems; in Section 3, we discuss some ways of constructing LS trees, or of reducing them to simpler ones; in Section

4, we give some results obtained by a computer search; finally, in Section 5, we add some further considerations related to the topic in question.

2. Basic results

Let $G = L(H)$, where H is not necessarily a tree. We will now introduce yet another vector space (a subspace of $\mathbb{R}^{V(H)}$), denoted by $\mathcal{E}^+(H)$, and defined as follows:

$$\mathcal{E}^+(H) = \{\mathbf{y} \mid \mathbf{y} = \mathcal{B}(H)\mathbf{x},\ \mathbf{x} \in \hat{\mathcal{E}}(H)\}.$$

Note, $\mathbf{y}(u) = \sum_{e \sim u} \mathbf{x}(e)$, where $e \sim u$ here means that an edge e is incident to the fixed vertex u (of H). We will now prove the following result.

Lemma 2.1. *Let $G = L(H)$. Then vector spaces $\mathcal{E}_0(G)$ ($\cong \hat{\mathcal{E}}(H)$) and $\mathcal{E}^+(H)$ are isomorphic.*

Proof. Let, for short, $\mathcal{B} = \mathcal{B}(H)$ and $\mathcal{A} = \mathcal{A}(G)$. We have already seen that $\mathcal{B}^T\mathcal{B} = \mathcal{A} + 2\mathcal{I}$. If $\mathbf{x} \in \hat{\mathcal{E}}(G)$, then $\mathcal{A}\mathbf{x} = 0$ and $\mathcal{B}^T\mathcal{B}\mathbf{x} = 2\mathbf{x}$. Thus (by pre-multiplying the latter by $\mathbf{y}^T$, where $\mathbf{y} \in \hat{\mathcal{E}}(G)$) we get $(\mathcal{B}\mathbf{y})^T(\mathcal{B}\mathbf{x}) = 2\mathbf{y}^T\mathbf{x}$. Furthermore, if $\mathbf{y} = \mathbf{x}$ then $\|\mathcal{B}\mathbf{x}\| = \sqrt{2}\,\|\mathbf{x}\|$. Therefore, $\mathcal{B}\mathbf{x} \neq \mathbf{0}$ if and only if $\mathbf{x} \neq \mathbf{0}$. Hence $\mathcal{B}$ is an injective linear transformation and thus an isomorphism as required. So the proof follows. $\square$

Remark 2.1. It is also interesting that $\mathcal{B}(H)$, as a mapping from $\hat{\mathcal{E}}(H)$ onto $\mathcal{E}^+(H)$, preserves the angles, but not the lengths. On the other hand, if we take $\mathcal{B}'(H) = \frac{1}{\sqrt{2}}\mathcal{B}(H)$, then the scalar product will be preserved (note that $(\mathcal{B}'(H)\mathbf{y})^T(\mathcal{B}'(H)\mathbf{x}) = \mathbf{y}^T\mathbf{x}$), and consequently the lengths as well. $\square$

Lemma 2.1 establishes that, via $\mathcal{E}_0(G)$ we are able to relate $\hat{\mathcal{E}}(H)$ in $\mathcal{E}^+(H)$ Thus we can focus on the root graph H only.

Assume henceforth that $\mathbf{x}$ and $\mathbf{y}$ are vectors in $\hat{\mathcal{E}}(H)$ and $\mathcal{E}^+(H)$, respectively, and that $\mathbf{y} = \mathcal{B}(H)\mathbf{x}$. We have already noted, how the components of $\mathbf{y}$ are related to the components of $\mathbf{x}$. We will now examine the converse relationship.

Lemma 2.2. *Given $\mathbf{x} \in \hat{\mathcal{E}}(H)$, then $\mathbf{y}(v) = \sum_{e \sim v} \mathbf{x}(e)$. Conversely, given $\mathbf{y} \in \mathcal{E}^+(H)$, and $e = uv$, we have*

$$\mathbf{x}(e) = \frac{\mathbf{y}(u) + \mathbf{y}(v)}{2}.$$

Proof. The first part follows from $\mathbf{y} = \mathcal{B}\mathbf{x}$, where $\mathcal{B} = \mathcal{B}(H)$. To prove the second part, consider the relation $\mathcal{B}^T\mathcal{B}\mathbf{x} = 2\mathbf{x}$ (taken from the proof of Lemma 2.1). Thus we get $\mathcal{B}^T\mathbf{y} = 2\mathbf{x}$, and the result immediately follows. $\square$

Remark 2.2. We can use Lemma 2.2 to get more general results for the components of $\mathbf{y}$.

Firstly, assume that $u_0, u_1, \ldots, u_k$ is a walk in H starting from $v = u_0$ and terminating in $w = u_k$. Then we have:

$$\mathbf{y}(w) - (-1)^k\mathbf{y}(v) = 2\sum_{i=0}^{k-1}(-1)^i\mathbf{x}(u_iu_{i+1}). \tag{2.1}$$

To see this, we observe that $\mathbf{y}(u_i) + \mathbf{y}(u_{i+1}) = 2\mathbf{x}(u_i u_{i+1})$, for each $i = 0, 1, \ldots, k$ (see Lemma 2.2). After multiplying each of these relations with $(-1)^i$, and summing in i over the above range, the required result follows.

Secondly, assume that u is any vertex of H and $v_1, v_2, \ldots, v_k$ its neighbours (here $k = \deg(u)$). Then by Lemma 2.2, we have:

$$(\deg(u) - 2)\mathbf{y}(u) + \sum_{i=1}^{k} \mathbf{y}(v_i) = 0. \tag{2.2}$$

To see this, we observe that (by Lemma 2.2) $\mathbf{y}(u) + \mathbf{y}(v_i) = 2\mathbf{x}(uv_i)$, for each $i = 1, 2, \ldots, k$. Summing these relations in i over the above range, we obtain (2.2). □

We will now assume that H is a tree with a singular line graph, i.e., an LS. Then we have:

LEMMA 2.3. *If T is an LS tree and $\mathbf{y} \in \mathcal{E}^+(T) \setminus \{\mathbf{0}\}$, then $\mathbf{y}(v) \neq 0$ for all $v \in V(T)$. In addition, all components of $\mathbf{y}$ can be chosen to be odd integers.*

PROOF. Recall, $\mathbf{y} = \mathcal{B}(T)\mathbf{x}$ for some $\mathbf{x} \in \hat{\mathcal{E}}(T) \setminus \{\mathbf{0}\}$. Since $\mathbf{x}$ is, in fact, an eigenvector of $L(T)$ for an integral eigenvalue (i.e., for $\lambda = 0$), we can assume that all components of $\mathbf{x}$ are integers. Next, we can also assume that they are relatively prime, i.e., that their greatest common divisor is one (otherwise, this follows by an appropriate scaling). On the other hand, for any edge, say uv, of T we have $\mathbf{y}(u) + \mathbf{y}(v) = 2\mathbf{x}(uv)$ (by Lemma 2.2). Thus, $\mathbf{y}(u)$ and $\mathbf{y}(v)$ are of the same parity, whenever u and v are adjacent. But since T is connected, the same follows at once for any two vertices of T (see also (2.1)). Now, if all entries of $\mathbf{y}$ are odd, we are done. Otherwise, assume that all entries of $\mathbf{y}$ are even. Consider next the entries of $\mathbf{x}$. Then, at least one entry is odd (otherwise, their greatest common divisor is not one), and let, say $e_1 = vw_1$, be an edge for which $\mathbf{x}(e_1)$ is odd. Since $\mathbf{y}(w_1)$ is even and $\mathcal{B}(T)\mathbf{x} = \mathbf{y}$, there exists an edge incident to w_1, say $e_2 = w_1 w_2$ $(\neq e_1)$, for which $\mathbf{x}(e_2)$ is also odd. Next, since $\mathbf{y}(w_2)$ is even, there exists an edge incident to w_2, say $e_3 = w_2 w_3$ $(\neq e_2)$, for which $\mathbf{x}(e_3)$ is also odd. Repeating this procedure, after some number of steps, we will end up, since T is finite, with a hanging edge, say $e_h = w_{h-1} w_h$ $(\neq e_{h-1})$, for which $\mathbf{x}(e_h)$ is also odd. But then $\mathbf{y}(w_h)$ $(= \mathbf{x}(e_h))$ is odd, a contradiction. This completes the proof. □

We will now give results which are direct consequences of the ones above. The first one is the main theorem of [**5**] (see also [**7**]).

THEOREM 2.1. *If T is a tree, then the nullity of $L(T)$ is at most one.*

PROOF. Assume for contradiction, that the nullity of $L(T)$, or equivalently the dimension of $\hat{\mathcal{E}}(T)$, is at least two. But then, by Lemma 2.1, the same applies for $\mathcal{E}^+(T)$. Consider next two linearly independent vectors in $\mathcal{E}^+(T)$. Now we can easily construct their linear combination in which at least one entry is zero (but, of course, not all). But this contradicts Lemma 2.3, and the proof follows. □

We give a new proof of the following theorem, which is also proved in [**7**, Theorem 4.4].

THEOREM 2.2. *If T is an LS tree, then its order (i.e., the number of vertices) is even.*

PROOF. For any graph G, since $\mathcal{B}\mathbf{x} = \mathbf{y}$, then in particular $\mathbf{y}(u) = \mathbf{x}(e)$ for an end-vertex u and $\mathbf{y}(v) = \sum_{v\sim e} \mathbf{x}(e)$.

Summing over all vertices of a tree T we get

$$\sum_{v\in V(T)} \mathbf{y}(v) = 2 \sum_{e\in E(T)} \mathbf{x}(e).$$

For an LS tree, if $\mathbf{x}$ is taken to have the minimal integral norm (as in the proof of Lemma 2.3), then all components of $\mathbf{y}$ are odd, and the proof immediately follows. □

REMARK 2.3. Suppose that T is an LS tree, and e is one of its edges. Let T_1 and T_2 be the components of $T - e$. Denote by n_1 and n_2 the orders of T_1 and T_2, respectively. Notice that n_1 and n_2 are of the same parity (by Theorem 2.2). We also assume (due to scaling) that the entries of $\mathbf{x}$ are relatively prime (so all the components of $\mathbf{y}$ are odd). By using the same arguments as in the proof of Theorem 2.2, we can show that $\mathbf{x}(e)$ is even (or odd) if and only if n_1 and n_2 are even (resp. odd). So, in particular, all hanging edges of T have odd weights if $\mathbf{x}$ is chosen as above. □

THEOREM 2.3. *If $\mathbf{x} \in \hat{\mathcal{E}}(T) \smallsetminus \{\mathbf{0}\}$ where T is an LS tree, then all hanging edges attached to the same vertex (of T) have the same weights with respect to $\mathbf{x}$. In addition, all these weights are non-zero.*

PROOF. Observe first that the collection of hanging edges at some vertex of T gives rise in $L(T)$ to a collection of co-duplicate vertices in $L(T)$. Then, in the corresponding λ-eigenvector (for $\lambda = -1$), all entries can be taken to be equal to zero, except for two which correspond to a pair of co-duplicate vertices, where these two entries are equal in modulus, but of opposite signs. Now the result easily follows by the orthogonality of eigenvectors (of $L(T)$) corresponding to distinct eigenvalues. So, in particular, it holds for $\lambda = 0$. The rest of the proof is based on Lemma 2.3. Namely, if e is a hanging edge, then $\mathbf{x}(e) = 0$ would imply $\mathbf{y}(u) = 0$ where u is a terminal vertex of e (here, $\mathbf{x}$ and $\mathbf{y}$ are interpreted as usual). So the proof follows. □

3. Constructions and reductions

In this section we will consider the possibility of constructing new LS trees from the smaller ones. These constructions will also open an inverse problem, i.e., of reducing the LS trees to some simpler, or basic forms.

Our first construction is based on introducing an edge (or a bridge) between two copies of LS trees.

THEOREM 3.1. *Let T_1 and T_2 be two LS trees, and let T be a tree obtained from the (disjoint) union of T_1 and T_2 by connecting them by an edge e. Then T is an LS tree for any choice of e.*

PROOF. Let $e = u_1u_2$, where u_1 and u_2 are vertices belonging to T_1 and T_2, respectively. Let $\mathbf{x}_1$ and $\mathbf{x}_2$ be 0-eigenvectors of $L(T_1)$ and $L(T_2)$, respectively. Next, assume that $\mathbf{y}_1 = \mathcal{B}(T_1)\mathbf{x}_1$ and $\mathbf{y}_2 = \mathcal{B}(T_2)\mathbf{x}_1$. With an appropriate choice of $\mathbf{x}_1$ and $\mathbf{x}_2$ (and with some scaling if necessary – see also Lemma 2.3), we can assume that $\mathbf{y}_1(u_1) + \mathbf{y}_2(u_2) = 0$. Define next $\mathbf{x}$ as follows:

$$\mathbf{x}(f) = \begin{cases} 0 & f = e, \\ \mathbf{x}_1(f) & f \in E(T_1), \\ \mathbf{x}_2(f) & f \in E(T_2). \end{cases} \tag{3.1}$$

It is now a matter of routine to check that $\mathbf{x}$ is a 0-eigenvector of $L(T)$.

This completes the proof. □

REMARK 3.1. From this theorem, we can easily deduce that any tree T with a perfect matching is an LS tree. To see this, observe first that K_2 is an LS tree; observe next that any tree with a perfect matching can be constructed starting from K_2 by adding in turns (like in Theorem 3.1) a copy of K_2 to the previously constructed trees. □

From the proof of Theorem 3.1 (see (3.1)) we have that $\mathbf{x}(e) = 0$ whenever $\mathbf{x} \in \mathcal{E}_0(T)$ (see also Theorem 2.1). In view of this situation, we can say that some edge of an LS tree is *light* (*heavy*) if the corresponding entry in $\mathbf{x}$ is zero (resp. non-zero). In addition, any LS tree T without light edges will be called a *heavy* tree (note, then $L(T)$ is a nut graph, according to [**7**]).

We will now establish a criterion for distinguishing light from heavy edges in LS trees. Then we can split some non-heavy LS tree to smaller ones (which can be all heavy) – this can be viewed as a reduction of LS trees.

THEOREM 3.2. *Let T be an LS tree, and e any of its edges. Then e is a light edge (of T) if and only if both components of $T - e$ are LS trees.*

PROOF. In view of Remark 3.1, we have only to prove one half of the claim. For this purpose, assume that e is a light edge. If so, then the angle[1] (in $L(T)$) corresponding to the eigenvalue 0 for the vertex originating from the edge e (of T) is 0. Therefore, by interlacing, the multiplicity of 0 in $L(T - e)$ is at least one, but at most two (note, in $L(T)$ it was one, and in general it can change by one at most in any vertex deleted subgraph; in addition, it cannot be 0 since the angle in question is 0; see [**3**, Chapter 7]). So, at least one of the components of $T - e$ is an LS tree. Assume next that both components are not LS trees. In other words, if $T - e = T_1 \cup T_2$, assume that, say T_1, is an LS tree, but not T_2. Let $\mathbf{x}_1$ and $\mathbf{x}_2$ be the vectors obtained as restrictions of $\mathbf{x}$, the 0-eigenvector of $L(T)$, to the edges of T_1 and T_2, respectively. If so, $\mathbf{x}_2 = \mathbf{0}$. Next, let u_2 be the vertex of T_2 incident to e. But then, since $\mathbf{y} = \mathcal{B}(T)\mathbf{x})$, $\mathbf{y}(u_2) = 0$, a contradiction (by Lemma 2.3).

This completes the proof. □

---

[1] See [**2**, Chapter 4].

REMARK 3.2. Note that in line graphs of LS trees, the multiplicity of 0 in any vertex-deleted subgraph is either zero or two. □

In what follows we will focus our attention only on heavy LS trees. For these trees, we have the following structural restriction.

COROLLARY 3.1. *If T is a heavy LS tree, then it has no hanging paths of length greater than one.*

PROOF. Assume for contradiction that P is a hanging path of T whose length is at least two. Let ee be the edge of P which is adjacent to some hanging edge. It is now easy to see (by Theorem 3.2) that e is a light edge, and consequently, T is not a heavy tree. So the proof follows. □

We will now consider another possibility for getting new LS trees from some known ones (which are not necessarily heavy ones). The basic idea consists in a *two-phase modification* of some LS tree. The first phase is a vertex *splitting*. (Recall, by splitting a vertex, say u (in any graph G), we understand the following two steps: (i) deletion of u from G along with all incident edges; (ii) adding (to $G-u$) two vertices, say u_1 and u_2, and, for each edge uv of G, adding one edge which is equal to either u_1v or u_2v (but not both). Note that if $E(u)$ is a set of edges incident to u, then a vertex splitting is determined by a bipartition $E_1(u)\dot{\cup}E_2(u)$; then if $uv \in E_i(u)$ ($i=1$ or 2), this means that an edge u_iv is added in step (ii)). The second phase is related to an one-vertex extension. Namely, we then add a vertex, say w, adjacent only to vertices u_1 and u_2 (from (i)).

LEMMA 3.1. *Let T be an LS tree, and T' a tree obtained from T by a two-step modification described above. Then T' is an LS tree.*

PROOF. Under the above notation, let $e_1 = u_1w$ and $e_2 = u_2w$ (w is the vertex as in (ii) above). For the sake of simplicity, we will assume that the edges of T' incident with u_1 or u_2 are viewed as the edges of T incident to u. Then, the edges of T' are $E(T) \cup \{e_1, e_2\}$. Assume next that $\mathbf{x}$ is a 0-eigenvector of $L(T)$, and that $E(u)$, the set of edges of T incident to u, is divided into two disjoint subsets $E_1(u)$ and $E_2(u)$ as above). Define then a vector $\mathbf{x}'$ (on the edge set of T') as follows:

$$\mathbf{x}'(e) = \begin{cases} \mathbf{x}(e) & e \in E(T), \\ -\sum_{f \in E_2(u)} \mathbf{x}(f) & e = e_1, \\ -\sum_{f \in E_1(u)} \mathbf{x}(f) & e = e_2. \end{cases}$$

Now it is a matter of routine to check that the vector $\mathbf{x}'$ is a 0-eigenvector for $L(T')$. So the proof follows. □

REMARK 3.3. Notice first that the two-phase modification of an LS tree T in which $E_1(u)$ or $E_2(u)$ is a singleton, is equivalent to the subdivision (by inserting two vertices) of one of the edges (of T) incident to u. Notice also that there is no need (see Corollary 3.1) to subdivide hanging edges of LS trees – for otherwise, we get trees which are not heavy. Therefore, we will consider, further on, only the subdivisions of edges which lie in *internal paths*. (Recall that an internal path of any graph, not necessarily a tree, is a path between two fixed vertices which are both of degree at least three, while all other vertices between them (if any) are of degree two.) □

We will now ask when the resulting LS trees (obtained by the two-phase modification of LS trees) are heavy. By inspecting the proof of Lemma 3.1 get:

COROLLARY 3.2. *Let T be an LS tree, and let T' be a tree obtained by a two-phase modification of T with respect to u. Then T' is a heavy LS tree if and only if T is heavy and* $\sum_{e\in E_i(u)} \mathbf{x}(e) \neq 0$ $(i = 1$ *and* $2)$.

We will now turn to reductions. We will focus on deleting vertices of degree two from LS trees. In the sequel, $\cdot$ stands for coalescence (or dot product) $G \cdot H$ of two rooted graphs G and H (obtained by identifying a vertex of G with a vertex of H).

THEOREM 3.3. *Let T be an LS tree, and let u be a vertex (of T) of degree two having u_1 and u_2 as its neighbours. Let T_{u_1} and T_{u_2} be the components of $T - u$ with u_1 and u_2 as its roots. Then $T_{u_1} \cdot T_{u_2}$ is an LS tree. Moreover, it is a heavy LS tree if and only if T is a heavy LS tree.*

PROOF. Let $\mathbf{x}$ be a 0-eigenvector of $L(T)$. Set $a = \sum_{v\sim u_1, v\neq u} \mathbf{x}(u_1 v)$ and $b = \sum_{v\sim u_2, v\neq u} \mathbf{x}(u_2 v)$. Now we have (by the eigenvalue equations) that $\mathbf{x}(uu_2) = -a$ and $\mathbf{x}(uu_1) = -b$. Consider now the vector $\mathbf{x}'$ constructed as follows:

$$\mathbf{x}'(e) = \begin{cases} \mathbf{x}(e) & e \in E(T_{u_1}), \\ -\mathbf{x}(e) & e \in E(T_{u_2}). \end{cases}$$

It is now a matter of routine to check that the vector $\mathbf{x}'$ is a 0-eigenvector of $L(T')$. The rest of the proof immediately follows. □

REMARK 3.4. In Corollary 3.2, we have seen that the two-phase modification of a heavy LS tree T with respect to vertex u need not be a heavy LS tree. In fact, if $\sum_{e\in E_1(u)} \mathbf{x}(e) = 0$, while $\sum_{e\in E_2(u)} \mathbf{x}(e) \neq 0$ (note, by Lemma 2.3, the sums cannot be both equal to zero) then the edge $e = wu_2$ (in T') is light. Consequently, $T' - e = T_1 \cup T_2$ consists (by Theorem 3.2) of two LS trees. The first tree has at the vertex u_1 a hanging edge $f = u_1 w$; if we look at T then we see that it is equal to $(T_1 - w)_{u_1} \cdot (T_2)_{u_2}$. So, it follows that a hanging edge of some LS tree is replaced by an arbitrary LS tree (in forming T), or that T is (after some local modification) splitted into two smaller LS trees. Furthermore, we will call LS trees which do not allow splitting into smaller LS trees by a two-phase modification, *v-irreducible* trees. In this context, heavy LS trees (without light edges) will be called *e- irreducible* trees. □

In view of Theorem 3.3 and Remark 3.4, LS trees which do not have vertices of degree two, and which are both v-irreducible and e-irreducible, will be called *basic* LS trees.

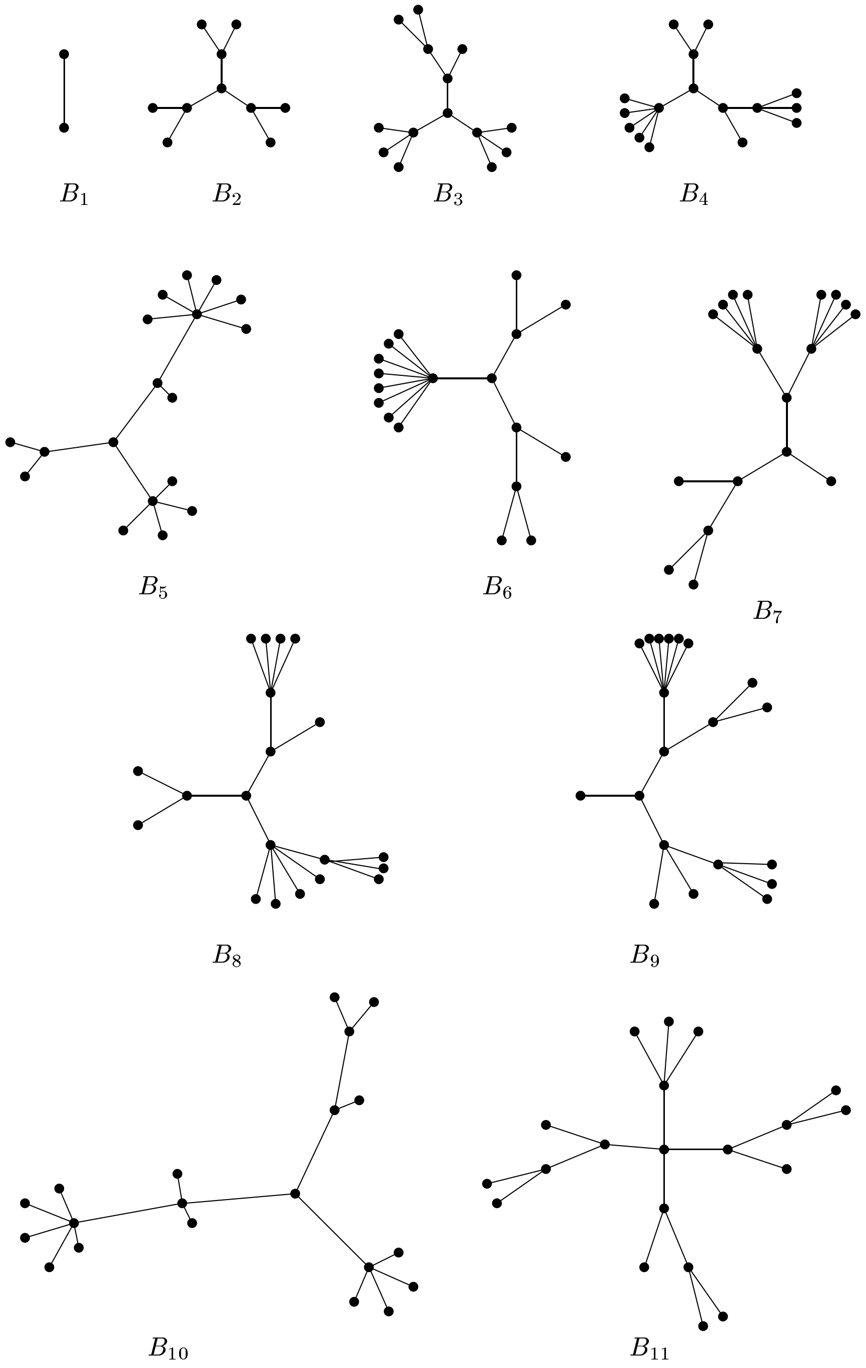

FIGURE 2. Basic LS trees up to 20 vertices

4. Basic LS trees with up to 20 vertices

In this section we will give some results obtained by a computer search. More precisely, we will list all basic LS trees with up to 20 vertices. These results are depicted in Fig. 2.

REMARK 4.1. There are 970 100 trees with an even number of vertices on up to 20 vertices. We have extracted from among these trees only 11 basic LS trees. Thus the property of being a *basic LS tree* seems to be a very rare phenomenon. □

In the next section we will give further basic LS trees obtained by constructions (which are inspired by the considerations from this and former sections).

5. Some further considerations

In this section, we consider problems related to considerations from the previous sections.

Firstly, we can ask whether the collection of basic LS trees is finite or not. With respect to our definition of basic LS trees the answer is negative. The following two LS trees (see Fig. 3) are basic for every $k \geqslant 1$ (this can be easily checked by constructing the corresponding eigenvector). They are all extensions of the LS tree on 10 vertices depicted in Fig. 2. Since k is unbounded, we have constructed an infinite family of basic LS trees.

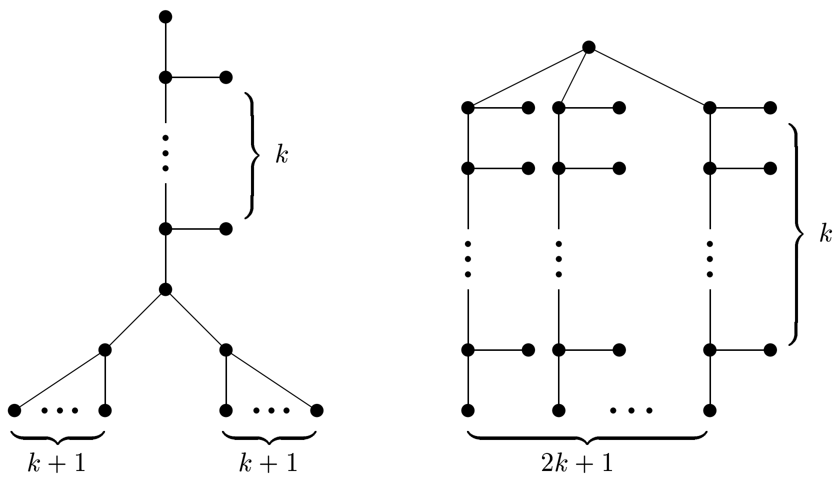

FIGURE 3. Two series of basic LS trees.

PROBLEM 5.1. *Are there other reductions (and constructions) which can be used to extend the definition of basic LS trees so that we get a finite set of "basic" LS trees?* □

Secondly, we can consider generalized line graphs. We first note that any petal gives rise to a pair of duplicate vertices. In particular, if $\hat{H}$ is a tree with petals, then G $(= L(\hat{H}))$ has nullity equal at most to the number of petals (in the root graph). This bound is increased by one if $\hat{H}$ with the petals replaced by an edge is an LS tree. However, if the root graph gives rise to a non-singular line graph upon the deletion of one edge from each petal, then the nullity is equal to the number of petals in $\hat{H}$. For example, if the number of vertices of the tree in question is odd, we get, for any number of petals, that the corresponding generalized line graph has a *rank property*, that is its adjacency matrix gets a full rank if all duplicate vertices are deleted. So we have here obtained (at least partially) one natural class of graphs for which the rank property holds (cf. Question 4.1, from [**9**]).

References

[1] D. Cvetković, M. Doob, H. Sachs, *Spectra of Graphs – Theory and Applications*, III revised and enlarged edition, Johan Ambrosius Bart. Verlag, Heidelberg–Leipzig, 1995.
[2] D. Cvetković, P. Rowlinson, S. Simić, *Eigenspaces of Graphs*, Cambridge University Press, 1997.
[3] D. Cvetković, P. Rowlinson, S. Simić, *Spectral Generalizations of Line Graphs – On graphs with least eigenvalue* -2, Cambridge University Press, 2004.
[4] M. Ellingham, *B*asic subgraphs and graph spectra, Australasian J. Comb. **8** (1993), 247–265.
[5] I. Gutman, I. Sciriha, *On the nullity of line graphs of trees*, Discrete Math. **232** (2001), 35–45.
[6] F. Harary, *Graph Theory*, Addison-Wesley, Reading, Mass., 1969.
[7] I. Sciriha, *On singular line graphs of trees*, Congr. Numer. **135** (1998), 73–91.
[8] I. Sciriha, *The two classes of singular line graphs of trees*, Rend. Sem. Mat. Messina, Ser II,**5** (1999), 167–180.
[9] G.F. Royle, *The rank of a cograph*, Electron. J. Comb. **10** (2003), #N11.

Department of Mathematics
Faculty of Electrical Engineering
University of Messina
Messina, Italy
marino@ingegneria.unime.it

(Received 09 12 2005)
(Revised 22 08 2006)

Department of Mathematics
Faculty of Science
University of Malta
Msida MSD 06, Malta
irene.sciriha-aquilina@um.edu.mt

Department of Mathematics
Faculty of Computer Science
11000 Belgrade, Serbia
ssimic@raf.edu.yu

School of Electrical Engineering
University of Belgrade
11000 Belgrade, Serbia
tosic@etf.bg.ac.yu