PUBLICATIONS DE L'INSTITUT MATHÉMATIQUE
Nouvelle série tome 49 (63), 1991, 21–26

AN ALGORITHM TO RECOGNIZE A GENERALIZED LINE GRAPH AND OUTPUT ITS ROOT GRAPH

Slobodan K. Simić

Abstract. We present an efficient algorithm of comlexity $O(m)$ (m being the number of lines) to recognize a generalized line graph giving as output its root graph.

1. Introduction. It is well known in the spectral graph theory (see [3]) that the least eigenvalue of 0-1 adjacency matrix of a line graph is bounded from below by -2. Generalized line graphs (GLG for short) were introduced in [6] as an extension of the line graphs with respect to this property. As shown in [4] (see also [2]), it turned out that this generalization became more interesting, since it also preserves some other properties of line graphs. In this paper we give an algorithm capable to recognize a generalized line graph, and also to output its root graphs (if any). Similar problems for line graphs (line digraphs) were treated in [9] and [10] (resp. [11]).

To make the paper more self contained, we shall include here some necessary facts. Let $H = (V, E)$ be a graph. The line graph $L(H)$ of H has E as its point set and two elements x, y of E are adjacent in $L(H)$ whenever the lines x, y of H have a common end-point. The cocktail party graph $CP(n)$ is a regular graph on $2n$ points of degree $2n-2$. To construct the generalized line graph, suppose we are given (in addition to H) an n-tuple $a = (a_1, \ldots, a_n)$ of nonnegative integers with a_i, the i-th coordinate of a, corresponding to the point v_i of H; then, the generalized line graph $L(H; a)$ of H is obtained from disjoint copies of $L(H)$ and $CP(a_i)$, by joining a point in $L(H)$ with a point in $CP(a_i)$ whenever the point in $L(H)$ corresponds to a line in H that has v_i as an end-point. As it can be seen, a generalized line graph is a graph actually obtained from some labeled graph (with coordinates of an n-tuple being the point labels). For our further purposes, we will prefer to have some graph (or even a multigraph) instead of a labeled graph. As already done in [4], we can convert a labeled graph to a multigraph as follows: to each point of H, say v_i, we shall (instead of a label) attach $2a_i$ pendant lines, each

AMS Subject Classification (1985): Primary 05 C 99

parallel to exactly one line out of them. Call this multigraph H_a; its generalized line graph is actually its line graph (see the definition above) provided two points are adjacent if the corresponding lines have "exactly" one end-point in common. Other terminology, notation and facts, not to be mentioned hereafter, can be found in [5].

2. Theoretical background. Two basic problems for line graphs (or generally, for graphs obtained by some graph valued function) are:

— characterization problem: given a graph, is it a line graph of some graph?

— determination problem: given a graph which is a line graph, to what extent is its root graph determined?

For line graphs and generalized line graphs, there are several answers to both problems. The most elegant solution of the characterization problem for line graphs (in terms of forbidden induced subgraphs) appeared in [1]. Its natural generalization for generalized line graphs is given in [4] (or [2]), and also, independently in [8]. As these subgraphs (in both cases) have up to six points, this immediately guarantees the existence of polynomial time algorithms (of degree at most six) for recognition of line graphs (resp. generalized line graphs). Although these characterizations are local, for our further purposes the following (essentially global ones) are more preferable. In [7] line graphs were characterized by their clique structure.

THEOREM 1. *A graph is a line graph if and only if its lines can be partitioned into cliques such that:*

(i) *each point is in at most two cliques, and*

(ii) *any two cliques have at most one point in common.*

To prepare the generalization from [4] (or [2]), we need some more definitions. The generalized cocktail party graph $GCP(n, m)$, is a graph obtained from a clique on n points by deleting m independent lines. A point of degree $n-1$ ($n-2$) is called 1-type (resp. 2-type) point.

THEOREM 2. *A graph is a generalized line graph if and only if its lines can be partitioned into generalized cocktail party graphs (GCPs for short) such that:*

(i) *each point is in at most two GPCs,*

(ii) *any two GCPs have at most one point in common, and*

(iii) *if two GCPs have a point in common, then it is of 1-type in both of them.*

We shall discuss the determination problem for connected graphs only. Actually, this restriction makes sense since the only connected graph with a disconnected generalized line graph is a graph consisting of a point whose label is one; on the other hand, a disconnected graph has a connected generalized line graph only if all but one of its components are trivial (just a point with a zero label). Now the well known result for line graphs (see [12], or [5]) reads:

THEOREM 3. *If $G = L(H)$ is a connected line graph, then except for $G = K_3$, its root graph H (ignoring isolated points) is unique; if $G = K_3$ the H is either K_3 or $K_{1,3}$.*

The next theorem refers to generalized line graphs [4] (or [2]).

THEOREM 4. *Except for the pairs of multigraphs from Fig. 1, if two connected generalized line graphs are equal, then their root multigraphs are also equal.*

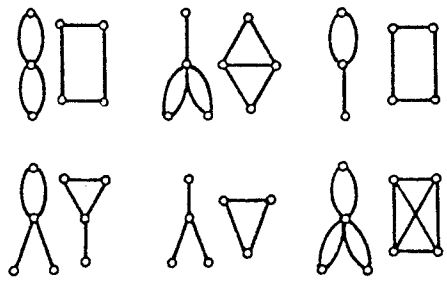

Fig. 1

A cover is a partition of lines of some graph into GCPs. It is called a proper cover if it satisfies the conditions (i)–(iii) of the Theorem 2. The next theorem is proved in [4].

THEOREM 5. *If G is a connected generalized line graph with more than 6 points, then there exists one and only one proper cover of G.*

This theorem is from now on of crucial importance. Its immediate consequence is that each point of a graph (satisfying the assumptions) could be assigned a type according to the rule: a point is of α-type if it is a common point of two GCPs; otherwise, it is of β-type (γ-type) if it is 1-type (resp. 2-type) point of the corresponding GCP.

In order to find a proper cover of some graph (if any), we first show how to single out an α-type point. From now on, $N(x)$ denotes the closed neighborhood of the point x, i.e. all points including x which are adjacent to x. In the following lemmas, we will assume that G is a connected generalizad line graph, other than a GCP, having at least 7 points.

LEMMA 1. *If x is an arbitrary point of G, then a point u of maximal degree among the points from $N(x)$ is an α-type point.*

Proof. By assumptions on G, its proper cover consists of at least two GCPs. Thus any point is adjacent to (or coincides with) some α-type point. On the other hand, it is obvious that an α-type point has more neighbors than each β-type or γ-type point adjacent to it. □

We next show how to resolve the closed neighborhood of some α-type point into two GCPs. Actually, it is enough to single out one of these two GCPs. The next lemma enables us to find in all but one particular case the points of this GCP with possibly one extra point (from the other GCP) included.

LEMMA 2. *Let u be an α-type point of G. We then have:*

1° *if u is adjacent to all other points of G, then G consists of two GCPs (u is their common point), and possibly, some independent lines which join the 1-type points from different GCPs.*

2° *if v ($\neq u$) is a point from $N(u)$ of maximal degree, then it is also an α-type point and these two points together with their common neighbors, all but possibly one, constitute a GCP.*

Proof. The first part is obvious. By Theorem 2, it follows at once that v is an α-type point. Let C be a GCP containing both u and v. More than one point outside C, adjacent to u and v, cannot exist without violationg (i) or (ii) from the Theorem 2. □

The discussion concerning the extra point from 2° in above, will be given in the next section. We now show how to complete the proper cover if we already know at least one of its members.

LEMMA 3. *If $C_1, \ldots, C_k$ are GCPs belonging to the proper cover of G, then, provided any of them is known, we can determine the rest of them by a simple procedure.*

Proof. Suppose, without loss of generality, that $C_1, \ldots, C_l$ ($l < k$) are already known. Observe the points of G which belong to these GCPs and are incident to lines not belonging to them. These points are α-type points, and since we know one GCP incident to them, the other is determined at once. □

3. The algorithm. The outline of the algorithm is as follows. We first single out an α-type point, the common point of the first two clusters (point sets which are intended to be GCPs). If these two clusters (basic ones) are indeed GCPs, we proceed by scanning all other α-type points not scanned so far. An α-type point is (fully) scanned if at least two clusters incident to it are detected. The process of scanning the α-type points has also strong refuting capabilities based on Theorem 2. If no contradiction is encountered, we first update the set of non scanned α-type points and then skip to the next α-type point. When all α-type points are scanned, the labelling of points is performed in order to get the root graph. The label of some point is intended to be a two element set consisting of the end points of the line in the root (multi-) graph to which the point observed corresponds. The process of labelling (at least partially) can also start in some earlier stages, as it will be described below.

From now on, only to gain clarity, we will assume that an instance graph G ($= (V, E)$) is a connected graph with at least 7 points, which is not a GCP.

Step 1: initial branching

Pick any point, say x. Find $u \in N(x)$ of the largest degree. If $\deg(u) = |V|-1$, then go to step 2; otherwise go to step 3.

Step 2: detection of basic clusters if $\mathrm{rad}(G) = 1$

Find in $N(u) \setminus \{u\}$ two points, say v and w, with the largest degres (assume $\deg(v) \geq \deg(w)$. Now we have:

a) If $\deg(v) = \deg(w) = |V| - 1$, then stop. ($G$ is not a GLG)

b) If $\deg(v) = |V| - 1$ and $\deg(w) < |V| - 1$, then set $S := \{s \mid s \in N(u) \cap N(v), \deg(s) = 2\}$. If $|S| \neq 1$ then stop; otherwise $C_1 := V \setminus \{s\}$, $C_2 := \{v, s\}$, and go to step 4.

c) If $\deg(v) < |V| - 1$, then, if $N(v) \cap N(w) = \{u\}$ set $C_1 := N(v)$, $C_2 := N(w)$; otherwise pick any point $s \in N(v) \cap N(w)$ not equal to u and set $C_1 := N(v) \cap (N(w) \cup N(s))$, $C_2 := (V \setminus C_1) \cup \{u\}$, and go to step 4.

Step 3: detection of basic clusters if $\mathrm{rad}(G) > 1$.

Pick $v \in N(u)$ of the largest degree, and $w \in N(v) \setminus N(u)$. Set $C_1 := (N(u) \cap N(v)) \setminus (N(u) \cap N(w))$, $C_2 := (N(u) \setminus C_1) \cup \{u\}$, and go to step 4.

Step 4: preparation of the main loop

Assign type to each point of C_1 and C_2. Then check if these clusters are indeed GCPs. If not, then stop; otherwise, check if β-type and γ-type points of these GCPs are adjacent only to the points within GCP to which they belong. If not, then stop; otherwise label u by setting $\mathrm{lab}(u) := \{1, 2\}$ and half label any other point, say x, of C_1 (or C_2) by setting $\mathrm{lab}(x) := \{1\}$ (resp. $\mathrm{lab}(x) := \{2\}$). Next, set $S_\alpha := \{s \mid s \in N(u) \setminus \{u\}, \mathrm{typ}(s) = \alpha\}$ (the α-type points to be scanned), $k := 2$ (the GCP counter), and go to step 5.

Step 5: main loop

Until $S_\alpha \neq \emptyset$ repeat. Pick $s \in S_\alpha$ for scanning. If $|\mathrm{lab}(s)| = 2$, then stop if there is a point in $N(s)$ not already visited (without type); otherwise, mark s as scanned and reenter the loop. If on the other hand $|\mathrm{lab}(s)| = 1$, then set $k := k + 1$ and $C_k := \{t \mid t \in N(s), \mathrm{lab}(s) \cap \mathrm{lab}(t) = \emptyset\} \cup \{s\}$. Next, assign type to each point of C_k, and check if it is a GCP. If not, then stop; otherwise, check if β-type and γ-type points of C_k are adjacent only to the points within it. If not, then stop; otherwise half label each point $t \in C_k$ by setting $\mathrm{lab}(t) := \mathrm{lab}(t) \cup \{k\}$. If $|\mathrm{lab}(t)| = 3$ for some t, then stop; otherwise, if t is an α-type point not already marked as scanned, then set $S_\alpha = S_\alpha \cup \{t\}$ and reenter the loop.

Finally, if we exit the loop with $S_\alpha = \emptyset$ then G is a GLG. To obtain its root graph, it is ehough to full label the points which are so far only half labeled (β and γ type points). To end this we have to keep track of the fact that two γ-type points from the same GCP have the same labels.

The corectness of the algorithm easily follows from the considerations given in the previous section. The time complexity is of $O(|E|)$, since for each point v of an instance graph, the amount of work done is of the order $O(\deg(v))$.

REFERENCES

[1] L. W. Beineke, *On derived graphe and digraphs*, in: *Beiträge zur Graphentheorie* (ed. H. Sachs et al.), Teubner-Verlag, Leipzig, 1968, 17–23.

[2] D. Cvetković, M. Doob, S. Simić, *Some results on generalized line graphs*, C.R. Math. Rep. Acad. Sci. Canada 2 (1980), 147–150.

[3] D. Cvetković, M. Doob, H. Sachs, *Spectra of graphs — Theory and Applications*, Deutscher Verlag der Wissenshaften – Academic Press, Berlin–New York, 1980.

[4] D. Cvetković, M. Doob, S. Simić, *Generalized line graphs*, J. Graph Theory 5 (1981), 385–399.

[5] F. Harary, *Graph Theory*, Addison-Wesley, Reading, Massachusetts, 1969.

[6] A. J. Hoffman, *Some recent results on spectral properties of graphs*, in: *Beiträge zur Graphentheorie* (ed. H. Sachs, H. J. Voss, H. Walther) Leipzig, 1968, 75–80.

[7] J. Krausz, *Démonstration nouvelle d'une théorème de Whitney sur les réseaux*, Math. Fiz. Lapok 50 (1943), 75–89.

[8] V. Kummar, S. B. Rao, N. M. Singhi, *Graphs with eigenvalues at least* -2, Linear Algebra Appl. 46 (1982), 27–42.

[9] P. G. H. Lehot, *An optimal algorithm to detect a line graph and output its root graph*, J. Assoc. Comput. Machinery 21 (1974), 569–575.

[10] N. D. Roussopoulos, *A* $\max\{m,n\}$ *algorithm for determining the graph H from its line graph*, Information Processing Letters 2 (1973), 108–112.

[11] M. M. Syslo, *A labelling algorithm to recognize line a line graph and output its root graph*, Information Processing Letters 15 (1982), 28–30.

[12] H. Whitney, *Congruent graphs and the connectivity of graphs*, Amer. J. Math. 54 (1932), 150–160.

Katedra za matematiku
Elektrotehnički fakultet
11000 Beograd, Jugoslavija

(Received 19 01 1989)