

A Linear Time Algorithm for the Line Subgraph Problem in Halin Graphs

Dingjun Lou, Jun Liang and Guanpu Han

School of Data and Computer Science, Sun Yat-sen University, Guangzhou 510006, People's Republic of China

Abstract—Given a graph $G=(V,E)$ and a positive integer $p \leq |E|$. The Line Subgraph Problem is: Is there a subset $E' \subseteq E$ such that $|E'| \geq p$ and $H=(V,E')$ is a line graph. In this paper, we design a linear time algorithm to solve the line subgraph problem for Halin graphs. The algorithm is optimal.

keywords-halin graph; line graph; linear time algorithm

I. INTRODUCTION AND TERMINOLOGY

A Halin graph H is defined as follows: First, we embed a tree T in the plane such that each inner vertex of T has degree at least 3, then we draw a cycle C through all leaves of T to form a planar graph. Then $H=T \cup C$ is called a Halin graph, where T is called the characteristic tree of H and C is called the accompanying cycle of H . An example of a Halin graph is shown in Figure 1.

The simplest Halin graphs are wheels. Suppose a Halin graph $H=T \cup C$ is not a wheel. If w is an inner vertex of T such that all neighbors v_1, v_2, \dots, v_k of w except one neighbor are leaves of T , then the induced subgraph $H[\{w \cup \{v_1, v_2, \dots, v_k\}]$ is called a fan of H and w is called the center of the fan (See Figure 1). We define a fan with k leaves of T to be a k -fan. If an inner vertex w' of T is not the center of a fan but w' is adjacent to k consecutive leaves v_1, v_2, \dots, v_k of T on C , then $H[\{w' \cup \{v_1, v_2, \dots, v_k\}]$ is called a pseudo k -fan, in particular, if $k=1$, then the pseudo k -fan is called a pseudo single fan.

For all terminology and notation not defined in this paper, the reader is referred to [1].

Halin graphs were introduced by German mathematician Halin [3] as minimally 3-connected planar graphs. It can be used as a model of a network with minimum cost and fault tolerance. Halin graphs have many good properties. For example, many NP-complete or NP-hard problems restricted on Halin graphs can be solved in polynomial time or even in linear time.

The TSP problem for general graphs is an NP-hard problem. However, Cornuejols, Naddef and Pulleyblank [2] give a linear time algorithm to solve the TSP for a weighted Halin graph. Lou [5] proves that each Halin graph is Hamiltonian connected, that is, for each pair of vertices u and v in a Halin graph H , there is a Hamiltonian path from u to v in H . Then Li, Lou and Lu [4] design a linear time algorithm

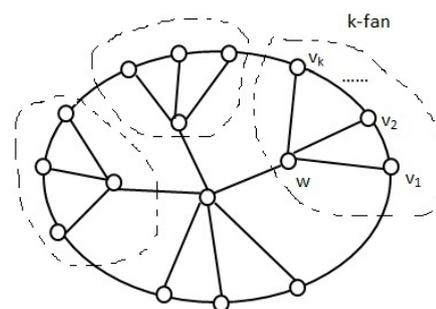


FIGURE 1. AN EXAMPLE FOR A HALIN GRAPH

to find a Hamiltonian path with minimum weight between each pair of vertices in a weighted Halin graph.

Phillips, Punnen and Kabadi [7] design a linear time algorithm to solve the Bottleneck TSP in a weighted Halin graph. Lou and Liu [6] give a linear time algorithm to solve the cubic subgraph problem for Halin graphs, which is to find a cubic subgraph with maximum number of edges in a Halin graph. The problems mentioned above for general graphs are all NP-complete or NP-hard.

In this paper, we give a linear time algorithm to solve the line subgraph problem for Halin graphs. The line subgraph problem is: Given a graph $G=(V,E)$ and a positive integer $p \leq |E|$. Is there a subset $E' \subseteq E$ such that $|E'| \geq p$ and $H=(V,E')$ is a line graph.

The line subgraph problem for general graphs is also NP-complete. We restrict the problem on Halin graphs and design a linear time algorithm to solve the problem, the algorithm is optimal.

The line graph H of a graph $G=(V,E)$ is such a graph that, for each edge $e \in E(G)$, H has a vertex v_e corresponding to e , two vertices v_e and v_f of H are adjacent if and only if their corresponding edges e and f are adjacent in G (i.e. e and f have a common end vertex in G). Such a graph G is called the base graph of H .

II. ANALYZING THE STRUCTURE OF A HALIN GRAPH

In the following, we present a property of the line graphs which is a necessary condition for a graph to be a line graph.

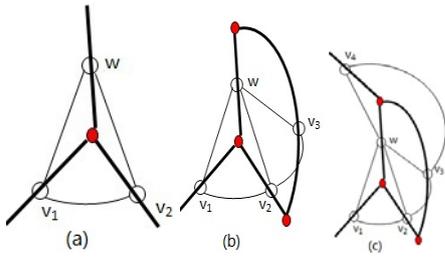


FIGURE II. THE STRUCTURE W IS A FAN

Lemma 2.1 A line graph is a claw-free graph.

Lemma 2.1 is a well-known property of the line graphs. A claw of a graph G is an induced subgraph isomorphic to $K_{1,3}$.

Now we analyze the structure of a Halin graph $H=T \cup C$ to find a base graph of a structure W of H , where the base graph is drawn with thick lines, and W is drawn with thin lines.

Case (1): The structure W is a fan.

Case (1.1): If W is a 2-fan $H[\{w\} \cup \{v_1, v_2\}]$, then its base graph exists (See Figure 2(a)). So 2-fan is a line subgraph.

Case (1.2): If W is a 3-fan $H[\{w\} \cup \{v_1, v_2, v_3\}]$, then its base graph exists (See Figure 2(b)). So 3-fan is a line subgraph.

Case (1.3): If W is a 4-fan $H[\{w\} \cup \{v_1, v_2, v_3, v_4\}]$, then its base graph exists (See Figure 2(c)). So 4-fan is a line subgraph.

Case (1.4): If W is a k -fan ($k \geq 5$) $H[\{w\} \cup \{v_1, v_2, \dots, v_k\}]$, then $H[\{w\} \cup \{v_1, v_3, v_5\}]$ is a claw, so W is not a line subgraph, and we shall delete the edges wv_5, wv_6, \dots, wv_k to find a line subgraph.

Case (2): The center of a fan is adjacent to an inner vertex of T .

Case (2.1): If the fan W is a 2-fan, then the center of W can be adjacent to an inner vertex of T , the base graph exists (See Figure 3).

Case (2.2): If the fan W is a k -fan ($k \geq 3$), $W=H[\{w\} \cup \{v_1, v_2, \dots, v_k\}]$, and if w is adjacent to an inner vertex x of T , then $H[\{w\} \cup \{x, v_1, v_3\}]$ is a claw. So, to find a line subgraph, the center of a k -fan ($k \geq 3$) cannot be adjacent to any inner vertex of T .

Case (3): The leaves of T in a fan W are adjacent to other vertices on C .

Case (3.1): If W is a 2-fan, and $W=H[\{w\} \cup \{v_1, v_2\}]$, then v_1 and v_2 can be adjacent to the neighbors on C , the base graph exists (See Figure 4(a)). In fact, v_1 (v_2) can be connected by a path on C to v_5 (v_3) in the line subgraph.

$v_4\}$, then v_1 and v_4 can be adjacent to the neighbors v_5 and v_7 respectively on C (See Figure 4(c)). In fact, v_1 (v_4) can be connected by a path on C to v_5 (v_7) in the line subgraph.

Case (4): An inner vertex w of T is contained in pseudo fans.

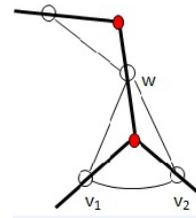


FIGURE III. THE CENTER OF A 2-FAN IS ADJACENT TO AN INNER VERTEX OF T

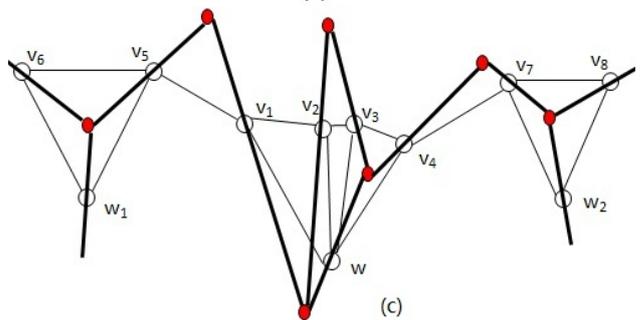
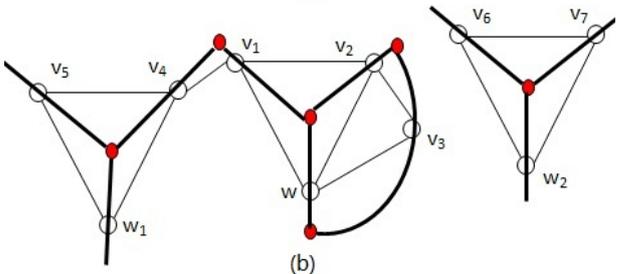
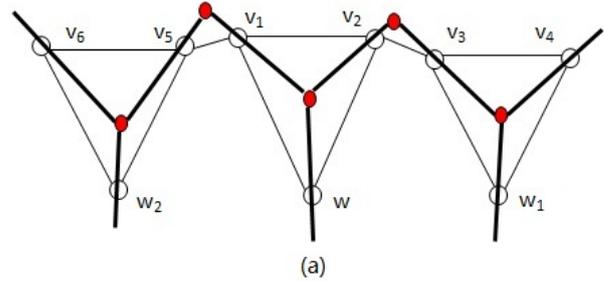


FIGURE IV. THE LEAVES OF T IN A FAN W ARE ADJACENT TO OTHER VERTICES ON C

Case (3.2): If W is a 3-fan, and $W=H[\{w\} \cup \{v_1, v_2, v_3\}]$, then v_1 can be adjacent to a neighbor v_4 on C , but v_3 cannot (See Figure 4(b)). So, to find a line subgraph with maximum number of edges, we must delete one edge wv_3 , then treat the fan $W-wv_3$ as a 2-fan. Then w can be adjacent to an inner vertex of T and v_3 can be adjacent to the neighbor v_6 on C .

Case (3.3): If W is a 4-fan, and $W=H[\{w\} \cup \{v_1, v_2, v_3,$

Case (4.1): If the inner vertex w of T is contained in a pseudo single fan $W=H[\{w\} \cup \{v_1\}]$, then $H[\{v_1\} \cup NH(v_1)]$ is a claw. So the edge wv_1 will be deleted to find a line subgraph.

Case (4.2): If the inner vertex w of T is contained in one pseudo 2-fan $W=H[\{w\} \cup \{v_1, v_2\}]$, similar to Cases(1.1),

(2.1) and (3.1), W can be kept and w can be adjacent to an inner vertex of T , and $v_1(v_2)$ can be adjacent to its neighbors on C .

Case (4.3): If the inner vertex w of T is contained in one pseudo 3-fan $W=H[\{w\} \cup \{v_1, v_2, v_3\}]$, then the edge wv_3 will be deleted, w can be adjacent to an inner vertex of T and $v_1(v_3)$ can be adjacent to its neighbors on C .

Case (4.4): If the inner vertex w of T is contained in one pseudo 4-fan $W=H[\{w\} \cup \{v_1, v_2, v_3, v_4\}]$, similar to Cases(1.3), (2.2) and (3.3), W can be kept, w cannot be adjacent to any inner vertex of T , and $v_1(v_4)$ can be adjacent to its neighbors on C .

Case (4.5): If the inner vertex w of T is contained in one pseudo k -fan $W=H[\{w\} \cup \{v_1, v_2, \dots, v_k\}]$ for $k \geq 5$, similar to Cases(1.4), (2.2) and (3.3), W is not a line subgraph, we shall delete the edges wv_5, wv_6, \dots, wv_k , w cannot be adjacent to any inner vertex of T and $v_1(v_k)$ can be adjacent to its neighbors on C .

Case (4.6): If the inner vertex w of T is contained in at least two pseudo fans $W_1=H[\{w\} \cup \{v_1, v_2, \dots, v_r\}]$, $W_2=H[\{w\} \cup \{u_1, u_2, \dots, u_s\}]$ ($r \geq 2$ and $s \geq 2$) and W_3, W_4, \dots, W_t , then delete the edges $wv_3, wv_4, \dots, wv_r, wu_3, wu_4, \dots, wu_s$ and all edges from w to the leaves of T in W_3, W_4, \dots, W_t , and w cannot be adjacent to any inner vertex of T and the edges on C incident with (or contained in) W_1, W_2, \dots, W_t remain. The remaining graph is a line subgraph (See Figure 5).

Case (5): The structure of inner vertices of T .

If w is an inner vertex of T such that w is not the center of a fan or a pseudo fan, $d_T(w) \geq 3$, and the vertices in $NH(w)$ are inner vertices of T , then $H[\{w\} \cup NH(w)]$ forms a claw. So the inner vertices of T form a line subgraph if and only if they form a forest with each tree to be a path by deleting some edges of T .

To obtain a line subgraph with maximum number of edges in $T-S$ (S is the set of all leaves of T), the inner vertices of T form a spanning subgraph T' of $T-S$, which is a set of paths. T' satisfies the following properties: The vertices x_1, x_2, \dots, x_r of degree less than 2 in T' are adjacent (in T) to vertices y_1, y_2, \dots, y_s of degree 2 (in T') but x_1, x_2, \dots, x_r are not adjacent to y_1, y_2, \dots, y_s in T' ; or x_1, x_2, \dots, x_r are adjacent (in T) to vertices w_1, w_2, \dots, w_t , where w_1, w_2, \dots, w_t are centers of k -fans or pseudo k -fans, but by Cases (2.2), (3.2) and (4), w_1, w_2, \dots, w_t are not adjacent to x_1, x_2, \dots, x_r in T' ; or x_i belongs to $\{w_1, w_2, \dots, w_t\}$ for some i 's, and by Cases (2.2), (3.2) and (4), x_i has degree less than 2 in T' .

To see that T' has maximum number of edges, suppose another spanning line subgraph T'' has maximum number of edges, but x_i ($1 \leq i \leq r$) has degree $d_{T'}(x_i) + 1$ in T'' , then $x_i y_j$ belongs to $E(T'')$, but then y_j has degree at least 3, so at least one edge incident with y_j (in T') must be deleted in T'' ; or $x_i w_j$

belongs to $E(T'')$, but by Cases (2.2), (3.2) and (4), w_j cannot have degree more than 1, so the edge incident with w_j in T' must be deleted in T'' ; or x_i belongs to $\{w_1, w_2, \dots, w_t\}$, by Cases (2.2), (3.2) and (4), x_i cannot have degree more

than its original degree in T' , otherwise the above two cases happen. We can obtain T'' from T' by a series of operations as above three cases. We can see that T'' will not have more edges than T' .

III. ALGORITHM TO FIND A MAXIMUM LINE SUBGRAPH

For any given Halin graph $H=T \cup C$, we use the following algorithm to find a maximum line subgraph for H .

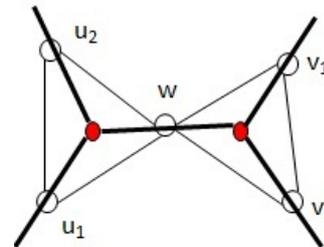


FIGURE 5. THE INNER VERTEX w OF T IS CONTAINED IN AT LEAST TWO PSEUDO k -FANS

Algorithm:

1. We do postorder traversal of T and choose an inner vertex r of T as the root such that r is not the center of any fan if there is such a vertex; otherwise r is any inner vertex of T .

2. When we visit an inner vertex w of T in the postorder traversal,

(2.1) If w is the center of a wheel $W=H[\{w\} \cup \{v_1, v_2, \dots, v_k\}] = H$, and if $k \geq 4$, then delete wv_5, wv_6, \dots, wv_k from H ; if $k=3$, then delete wv_3 . The resulting graph is a maximum line subgraph.

(2.2) If w is the center of a k -fan $W=H[\{w\} \cup \{v_1, v_2, \dots, v_k\}]$,

(2.2.1) If $k=2$, keep the edge joining w to another inner vertex in T ;

(2.2.2) If $k=3$, delete the edge wv_3 , and keep the edge joining w to another inner vertex in T ;

(2.2.3) If $k=4$, delete the edge joining w to another inner vertex of T ;

(2.2.4) If $k \geq 5$, delete the edges wv_5, wv_6, \dots, wv_k and the edge joining w to another inner vertex of T ;

(2.3) If w is contained in a pseudo k -fan $W_1=H[\{w\} \cup \{v_1, v_2, \dots, v_k\}]$,

(2.3.1) While w is contained in a pseudo single fan $W_1=H[\{w\} \cup \{v_1\}]$, do deleting the edge wv_1 ;

(2.3.2) If w is contained in one pseudo k -fan W_1 , then

(2.3.2.1) If $k=2$, keep only one edge which joins w to another inner vertex as its child in T ; if there is not such a child, then keep the edge to its father; Then delete the other edges joining w to other inner vertices of T ;

(2.3.2.2) If $k=3$, delete wv_3 , then do the same work as (2.3.2.1);

(2.3.2.3) If $k=4$, delete the edges joining w to other inner vertices of T ;

(2.3.2.4) If $k \geq 5$, delete the edges wv_5, wv_6, \dots, wv_k and do the same work as (2.3.2.3).

(2.3.3) If w is contained in at least two pseudo fans $W_1=H[\{w\} \cup \{v_1, v_2, \dots, v_r\}]$, $W_2=H[\{w\} \cup \{u_1, u_2, \dots, u_s\}]$ and W_3, W_4, \dots, W_t , and $r, s \geq 2$, then delete wv_3, wv_4, \dots, wv_r ; wu_3, wu_4, \dots, wu_s ; and all edges joining w to the leaves of T in W_3, W_4, \dots, W_t . Then delete the edges joining w to other inner vertices of T .

(2.4) If w is only adjacent to inner vertices of T , then two edges to its children are kept; if w is adjacent to only one child, then we keep one edge to its child and one edge to its father. Then we delete the other edges incident with w . If there is only one edge incident with w , we keep it.

3. When we have visited the root r , all vertices have been processed as in Step 2, and then the resulting graph is a maximum line subgraph.

By the analysis of the structure of a Halin graph H in Section 2, we can see that our algorithm is correct. Since our algorithm only do once postorder traversal of T , the worst time complexity of the algorithm is $O(n)$, where n is the number of vertices in H (we shall also notice that $|E(H)|=|V(H)|=O(n)$).

REFERENCES

- [1] J.A. Bondy and U.S.R. Murty, Graph theory with applications, Macmillan Press, London (1976).
- [2] G.Cornuejols, D.Naddef and W.Pulleyblank, Halin graphs and the traveling salesman problem, Mathematical Programming 26(1983), 287-294.
- [3] R.Halin, Studies on minimally n -connected graphs, Combinatorial Mathematics and its Applications, Academic Press, London(1971), 129-136.
- [4] Y.Li, Dingjun Lou, Y.Lu, Algorithms for the optimal Hamiltonian path in Halin graphs, Ars Combinatoria 87(2008), 235-255.
- [5] Dingjun Lou, Hamiltonian paths in Halin graphs, Mathematica Applicata(Chinese) 8:2(1995), 158-160.
- [6] Dingjun Lou and Junfu Liu, A linear time algorithm for cubic subgraph of Halin graphs, Proceedings of the 2nd International Conference on Computer Science and Electronics Engineering (ICCSEE 2013) (2013), 386-388.
- [7] J.M.Phillips, A.P.Punnen and S.N.Kabadi, A linear time algorithm for the bottleneck traveling salesman problem on a Halin graph, Information Processing Letters, 67(1998), 105-110.