Algorithms for Square Roots of Graphs (Extended Abstract)

Yaw-Ling Lin
Steven S. Skiena*

Department of Computer Science
State University of New York
Stony Brook, NY 11794-4400
e-mail: {yawlin,skiena}@sbcs.sunysb.edu

1 Introduction

Given a graph G, the *distance* between vertices u and v, denoted by $d_G(u,v)$, is the length of the shortest path from u to v in G. The *kth power* ($k \geq 1$) of G, written G^k, is defined to be the graph having $V(G)$ as its vertex set with two vertices u, v adjacent in G^k if and only if there exists a path of length at most k between them, i.e., $d_G(u,v) \leq k$. Similarly, graph H has an *kth root* G if $G^k = H$. For the case of $k = 2$, we say that G^2 is the *square* of G and G is the *square root* of G^2. The square of a graph can be computed by squaring its adjacency matrix.

Powers of graphs have many interesting properties. For example, Fleischner [4] proved that the square of a biconnected graph is always hamiltonian. Although the hamiltonian cycle problem is NP-complete for general graphs [5], the biconnectivity of a graph can be tested in linear time. Thus an efficient algorithm for finding the square root of a graph could be useful for finding hamiltonian cycles in square graphs.

Mukhopadhyay [15] showed that a connected undirected graph G with vertices $v_1, ..., v_n$ has a square root if and only if G contains a collection of n complete subgraphs $G_1, ..., G_n$ such that for all $1 \leq i, j \leq n$: (1) $\bigcup_{1 \leq i \leq n} G_i = G$, (2) $v_i \in G_i$, and (3) $v_i \in G_j$ if and only if $v_j \in G_i$. Characterizations of squares of digraphs were given by Geller [7] and of nth power of graphs and digraphs was given by Escalante, Montejano, and Rojano [3]. Unfortunately, these characterizations do not lead to efficient algorithms.

This paper studies two distinct classes of problems concerning powers of graphs. In Section 2, we concentrate on the problem of finding the square roots of graphs, presenting efficient algorithms for finding the tree square roots of G^2, finding square roots when

*The work of the second author was partially supported by NSF Research Initiation Award CCR-9109289

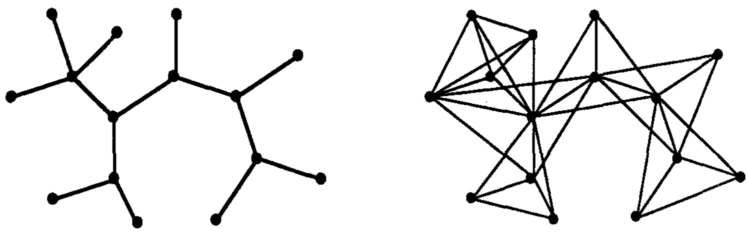

Figure 1: A tree and its square.

G^2 is a planar graph, and finding any square roots which are subdivision graphs. This last problem provides an efficient algorithm for inverting total graphs. In Section 3, we consider the complexity of finding maximum cliques and hamiltonian cycles in G^k given G. We conclude with a list of open problems.

2 Finding Square Roots of Graphs

In this section, we concentrate on the problem of finding a square root of a given graph. Not all graphs are squares, the smallest example being a simple path on three vertices. Further, a graph may have exponentially many distinct square roots. For example, any graph containing a vertex of degree $n-1$ is a square root of the complete graph K_n.

We give an $O(m)$ time algorithm for finding the tree square roots of a graph in Section 2.1, and an $O(n)$ algorithm for finding the square roots of a planar graph in Section 2.2. In Section 2.3, we provide an $O(m^2)$ time algorithm for the inversion of total graphs.

2.1 Tree Square Roots

In this section, we present an $O(|V|+|E|)$ algorithm for finding the tree square root T of a given graph $G = T^2 = (V, E)$. Tree square roots were first considered by Ross and Harary [16], who showed that they are unique up to isomorphism. Figure 1 presents a tree and its square. Later, in Section 3, we will consider algorithmic problems on tree squares and prove the graph theoretic result that all powers of trees are chordal.

Our algorithm for inversion proceeds by identifying the leaves of the tree square root, and then trimming all leaves from G to obtain a graph which is the square of a smaller tree.

Now consider a graph $T^2 = (V, E)$ which is known to be the square of some tree $T = (V, E')$. The *degree* of a vertex v in G, written $deg_G(v)$, is the size of its neighborhood in G. A vertex v is a *leaf* (or *endpoint*) of a graph G if $deg_G(v) = 1$.

The *diameter* of a graph G is the longest distance defined between a pair of vertices. A tree with diameter less than or equal to 2 is called a *star*. Note that the square of a star is a complete graph. The *neighborhood* of a vertex v in G, denoted by $N_G(v)$, is the set containing all vertices that are adjacent to v. Since $v \notin N_G(v)$, we denote $\{v\} \cup N_G(v)$ by $N'_G(v)$. A vertex v is *simplicial* if the induced subgraph of its neighborhood forms a clique.

Lemma 2.1 *v is a simplicial point of T^2 if and only if v is a leaf of T or T is a star.*

Proof. Assume that v is a leaf of T. Since $N'_{T^2}(v) = \{u : d_T(u,v) \leq 2\}$, the induced subgraph $\langle N'_{T^2}(v)\rangle_T$ can be obtained by starting from a leaf V and walking at most two steps in T. The resulting graph has a diameter less than or equal to two, therefore $\langle N'_{T^2}(v)\rangle_T$ is a star. The square of a star is a complete graph, so $\langle N'_{T^2}(v)\rangle_{T^2}$ is complete, meaning v is simplicial. Now assume v is not a leaf and T is not a star. Then there must be a path (x, v, y, z) in T. Since $d_T(x, z) = 3$ implies $xz \notin E(T^2)$ and $x, z \in N_{T^2}(v)$, v is not simplicial. ▮

For each leaf v of T, we can partition $N_{T^2}(v)$ into two sets L_v and M_v, where L_v denotes all leaves of T in $N_{T^2}(v)$ and M_v the set $N_{T^2}(v) - L_v$. According to Lemma 2.1, a leaf node has the lowest degree of any vertex in its neighborhood of T^2. The following lemma characterizes when the degree of a leaf will be strictly less than the degree of its neighboring internal nodes.

Lemma 2.2 *Let T be a non-star tree and v a leaf of T, then*

1. *For all $u \in L_v, deg_{T^2}(u) = deg_{T^2}(v)$.*
2. *For all $w \in M_v, deg_{T^2}(w) > deg_{T^2}(v)$.*

Proof. For part 1, let $u \in L_v$. Since $\langle N'_{T^2}(v)\rangle_T$ is a star, we know that $N'_{T^2}(v) = N'_{T^2}(u)$. So $deg_{T^2}(u) = deg_{T^2}(v)$. For part 2, let $w \in M_v$. Since T is not a star, without loss of generality, there is either a path (v, w, x, y) or a path (v, x, w, y) in T. For either case, $y \in N_{T^2}(w)$ but $y \notin N_{T^2}(v)$. We conclude that $N'_{T^2}(w) \not\subset N'_{T^2}(v)$. Since, by Lemma 2.1, $N'_{T^2}(v) \subset N'_{T^2}(w)$, $deg_{T^2}(w) > deg_{T^2}(v)$. ▮

The *center* of star S is the central vertex v such that for all other vertices u in S, $d_S(u, v) = 1$. If $|S| \neq 2$, there is only one center v in S, which we denote by *center*(S). Let K_n denotes the complete graph of size n. The cardinality of M_v, $|M_v|$, provides valuable information about the structure of the graph.

Lemma 2.3 *Let v be a leaf of T, then*

1. *if $|M_v| \leq 1$, T is a star;*
2. *if $|M_v| = 2$, say $M_v = \{x, y\}$, then $xy \in E(T)$;*
3. *if $|M_v| \geq 3$, $(v, center(M_v)) \in E(T)$.*

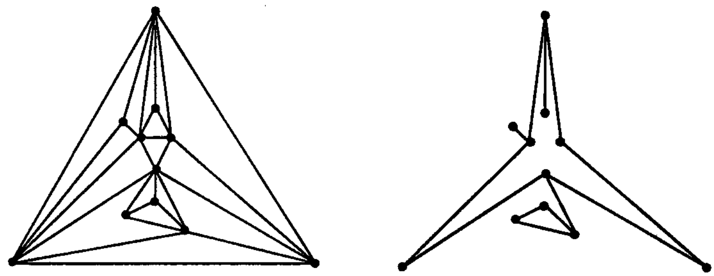

Figure 2: A planar graph and its square root.

Proof. For part 1, either $|M_v| = 0$, meaning $T = K_2$, or $|M_v| = 1$, where again, T is a star. For part 2, let $M_v = \{x, y\}$. Since v is a leaf, Lemma 2.1 implies that $\langle N'_{T^2}(v)\rangle_T$ is a star. Since $|N_{T^2}(v)| \geq |M_v| = 2$ meaning $|N'_{T^2}(v)| \geq 3$, $center(\langle N'_{T^2}(v)\rangle_T)$ is well-defined. Further, $center(\langle N'_{T^2}(v)\rangle_T) \notin L_v$. So, without loss of generality, we can let x be the center of $\langle N'_{T^2}(v)\rangle_T$. Again, since $\langle N'_{T^2}(v)\rangle_T$ is a star, y must be connected to x. So $xy \in E(T)$. For part 3, $|M_v| \geq 3$. Since $\langle N_{T^2}(v)\rangle_T$ is a star and v is the center, we know that $center(\langle N_{T^2}(v)\rangle_T) = center(\langle M_v\rangle_T)$. Clearly $(v, center(\langle N_{T^2}(v)\rangle_T)) \in E(T)$. ∎

Given a tree T, we can delete all the leaves of T resulting in a smaller tree T'. This trimming operation defines a function $trim(T) = T'$.

Theorem 2.4 *The tree square root of a graph G can be found in $O(m)$ time, where m denotes the number of edges of the given tree square graph.*

Proof. The proof appears in the complete paper. In summary, it takes $O(m)$ time to identify all leaves of T, and another $O(m)$ time to trim the tree and determine the structure of T. With a final $O(m)$ confirmation step, we conclude that the tree square root of a square graph can be identified in linear time. ∎

2.2 Square Roots of Planar Graphs

In this section, we present an $O(n)$ algorithm for finding the square root of a given planar graph, based on the characterization of planar squares found by Harary, Karp, and Tutte [10] and the linear time triconnected components algorithm given by Hopcroft and Tarjan [13]. Planar roots are not necessarily unique up to isomorphism. Figure 2 presents a planar graph and its square root.

Theorem 2.5 (Harary, Karp, and Tutte [10]) *A graph G has a planar square if and only if*

1. *every point of G has degree less than or equal to three,*

2. *every block of G with more than four points is a cycle of even length, and*

3. G does not have three mutually adjacent articulation vertices.

An *endline* of G is an edge uv of G such that either u or v is a leaf. A *burr* of G is a maximal connected (induced) subgraph of G in which every bridge is an end line. By removing all leaves of a burr B in G, the remaining subgraph B' is called the *central block* of B. Since B does not contain inner bridge, B' must be a biconnected component, a single vertex, or the null graph. As shown by Theorem 2.5, if G^2 is a planar graph, then for each burr B of G, we have three cases:

Case 1. The central block B' is an even length cycle with more than four vertices. Let C_n denote a chordless cycle of length n. That is, $B' = C_{2n}$, for some n greater than 2. Further, each vertex of B' is adjacent to at most one endline.

Case 2. The central block B' is a block of size at most four. Each vertex of B' can be linked with at most one leaf, providing it does not violate the conditions of Theorem 2.5. Such kind of burr has at most eight vertices. That is, a burr consists of a central block C_4 with each vertex adjacent to exactly one leaf.

Case 3. The central block B' consists of a single vertex v, which can be adjacent to at most three other leaves. That is, B is a star of size at most four.

Lemma 2.6 *The square of a burr is triconnected or complete; Further, each triconnected component of a planar square G^2 is the square of a burr in G.*

Proof. The proof appears in the complete paper. ∎

Hopcroft and Tarjan [13] shows that $O(m+n)$ time suffices to find all triconnected components of a given graph. For planar graphs, the number of edges $m \leq 3n-6$, so we can find all triconnected components in $O(n)$ time.

For G^2 to be planar, each large burr in G must be an even length cycle, perhaps including some number of endlines. Otherwise, the central block will just be a finite graph with size less than five. The following lemma characterizes the leaves of a large burr.

Lemma 2.7 *Given a planar square graph G^2 and a burr B of G such that $|B| > 5$, then v is a leaf of B if and only if $deg_{B^2}(v) = 3$.*

Proof. Since $|B| > 5$ and B^2 is planar, the central block B' must have at least four vertices. Otherwise the condition of Theorem 2.5 will be violated. First we show that for each non-leaf v of B, $deg_{B^2}(v) > 3$. If B' is an even length cycle C_{2n} such that $n \geq 3$, then $deg_{B^2}(v) \geq 4$. Otherwise, the central block is a block of size four. There are at least two leaves outside this size-four block. Again, $deg_{B^2}(v) \geq 4$.

Now we show the each leaf $v \in B$ has degree three in B^2. Let $uv \in B$ meaning $u \in B'$. Since B^2 is planar, $deg_B(u) \leq 3$. Since B' is a block, $deg_{B'}(u) \geq 2$, meaning $deg_B(u) \geq 3$. Since $deg_B(u) = 3$, $deg_{B^2}(v) = 3$. ∎

Now we show that we can efficiently find the original burr given its square.

Lemma 2.8 *Given a planar graph B^2, where B is a burr, the structure of B can be computed in $O(|B|)$ time.*

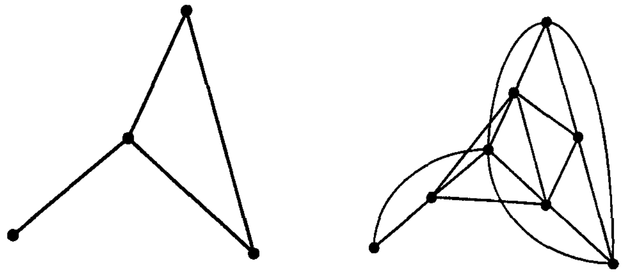

Figure 3: A graph and its total graph

Proof. The proof appears in the complete paper. ∎

Theorem 2.9 *The square root of a planar graph G^2 can be found in $O(n)$ time.*

Proof. By Lemma 2.6, the square burrs of G^2 can be found in $O(n)$ time. By Lemma 2.8, the structure of each burr B_i can be found in $O(|B_i|)$ time. The rest of the proof, which shows that to indentify the inner bridges between burrs can be done in $O(n)$ time also, appears in the complete paper. It then follows that finding the square root of a planar graph can be done in linear time. ∎

2.3 Inversion of Total Graphs

The *total graph* $T(G)$ of a graph $G = (V, E)$ has vertex set $V \cup E$ with two vertices of $T(G)$ adjacent whenever they are neighbors in G. If uv is an edge of G, and w not a vertex of G, then uv is *subdivided* when it is replaced by two edges uw and wv. If every edge of G is subdivided, the resulting graph is the *subdivision graph* $S(G)$. [9] Behzad [2] showed that, given a graph G, the total graph $T(G)$ is isomorphic to the square of its subdivision graph $S(G)$. In this section, we present an efficient algorithm for inverting total graphs by reducing the problem to finding square roots of the squares of subdivision graphs. Figure 3 presents a graph with its total graph.

Recall that a *maximal clique* of a graph is an induced complete subgraph such that no other vertex can be added to form a larger clique. We can identify whether an edge uv is in $S(G)$ by examining these maximal cliques in the square containing both vertices u and v.

Lemma 2.10 *Given a graph $G = (V, E)$ and an edge uv of G, there are at most two maximal cliques of size 3 containing both u and v in G^2.*

Proof. Let $K = \{u, v, w\}$ be a maximal clique in G^2. Note that it is not possible that $d_G(u, w) = d_G(v, w) = 2$ in G since that will imply that K is not maximal. So, without loss of generality, we can assume that uw is an edge of G. Note that v and w are the only vertices allowed to be adjacent to u since, otherwise, K will not be maximal. It is possible that v can be adjacent to a vertex $w' \neq w$ such that $N_G(v) = \{u, w'\}$, implying $\{u, v, w'\}$ is also a maximal clique, but that covers all possible maximal cliques of size 3 containing both u and v. ∎

Given a graph $G = (V, E)$ and its subdivision graph $S(G)$, we call the set of newly added vertices W, such that $V(S(G)) = V \cup W$. By Behzad's result, we know that $T(G) = [S(G)]^2$. Given $v \in V$, we call $Inner(v) = N_{S(G)}(v)$, the *inner neighborhoods* of vertex v, and $Outer(v) = N_G(v)$, the *outer neighborhoods.* Note that $N_{T(G)}(v) = N_{[S(G)]^2}(v) = Inner(v) \cup Outer(v)$. All inner nodes are elements of W, and all outer nodes are elements of V. We observe that

Lemma 2.11 *Let uv be an edge of G and w the subdivided vertex of uv, implying $uw, wv \in S(G)$. Then*

> *(i) $Inner(v) = N_{T(G)}(v) \cap N_{T(G)}(w) - \{u\}$*
>
> *(ii) For each pair $(w', u') \in Inner(v) \times Outer(v), w'u' \in S(G)$ if and only if $N_{T(G)}(w') \cap N_{T(G)}(v) - Inner(v) = \{u'\}$*

Proof. (i) Since, for each $x \in Outer(v) - \{u\}, d_{S(G)}(w, x) = 3$, it follows that $Inner(v) = N_{T(G)}(v) \cap N_{T(G)}(w) - \{u\}$.

(ii) The only if part is trivial by (i). Now assume that $\{u'\} = N_{T(G)}(w') \cap N_{T(G)}(v) - Inner(v)$. Note that $d_{S(G)}(u', w') \leq 2$ and $d_{S(G)}(u', v) = 2$. It follows that $u'v' \in S(G)$. ∎

With these results, now we are ready to present a polynomial time algorithm for the inversion of total graphs.

Theorem 2.12 *The inversion of a total graph H can be performed in $O(m^2)$ time where m denotes the number of edges of H.*

Proof. The proof appears in the complete paper. Denote the lowest degree of all vertices in H by δ and the number of degree δ vertices in $T(G)$ by α. In summary, it will spend at most $O(m)$ time for each possible triple $\{u, v, w\}$, while the number of candidate triples is bounded by $O(\alpha\delta)$. Since we need the time to initialize the adjacency matrix of H, the total time bound will be $O(n^2 + \alpha\delta m)$. In the worst case, this will take $O(m^2)$ time, although typically $\alpha\delta$ will be much smaller than m, which collapes the time complexity to $O(n^2)$. ∎

3 Algorithms on Powers of Graphs

In this section, we present several results concerning the complexity of two optimization problems for squares and higher powered graphs.

Hamiltonian cycles in powers of graphs have received considerable attention in the literature. In particular, Fleischner [4] proved that the square of a biconnected graph is always hamiltonian. Harary and Schwenk [11] proved that the square of a tree T is hamiltonian if and only if T does not contain $S(K_{1,3})$ as its induced subgraph. Here $S(K_{1,3})$ denotes the subdivision graph of the complete bipartite graph $K_{1,3}$. Harary and Schwenk's result leads to an optimal algorithm.

Theorem 3.1 *For any tree T, the problem of determining whether T^2 is hamiltonian can be answered in $O(n)$ time.*

Proof. The problem reduces to testing whether $S(K_{1,3})$ is a subgraph of T. Starting from a leaf v of T, we perform a depth-first search, for each node x computing the distance from v as well as the diameter of the subtree rooted at x. T^2 is hamiltonian unless there exists a vertex x which has three decendants of diameter at least two, or is a distance of at least two from v and has two decendants of diameter two. ∎

A graph G is *hamiltonian connected* if every two distinct vertices are connected by a hamiltonian path. Sekanina [17] proved G^3 is hamiltonian connected if G is connected, which implies that if the size of G is greater or equal to 3, then G^3 is hamiltonian. Here we present a linear time algorithm (in terms of the number of edges of input G) for finding a hamiltonian cycle in G^k, where $k \geq 3$.

Theorem 3.2 *Given a connected graph $G = (V, E)$ with size at least 3 and an integer $k \geq 3$, we can find a hamiltonian cycle in G^k within $O(|V| + |E|)$ time.*

Proof. The proof appears in the complete paper. ∎

Recall that a clique in a graph G is *maximum* if it is the largest induced complete subgraph of G. Here we prove that finding maximum cliques in powered graphs remains NP-complete by a reduction from the general problem of finding the maximum cliques in arbitrary graphs. In the subsequent section, we provide a linear algorithm for finding the maximum clique of a tree square.

Theorem 3.3 *Let $G = (V, E)$ be a graph. Then, for any fixed integer $k \geq 1$, finding the maximum clique of G^k is NP-complete.*

Proof. The proof appears in the complete paper. ∎

Graphs whose every simple cycle of length strictly greater 3 possesses a chord are called *chordal graphs.* In the literature, chordal graphs have also been called *triangulated, rigid-circuit, monotone transitive,* and *perfect elimination* graphs. Chordal graphs, which can be reconginized in linear time, are perfect graphs, and the maximum clique in chordal graphs can be determined in linear time [8].

Gavril [6] proved that a graph G is chordal if and only if G is the intersection graph of a family of subtrees of a tree. Figure 4 shows that the square of a chordal graph is not necessarily chordal. However, we will show that arbitrary powers of trees are always chordal.

Theorem 3.4 *All powers of trees are chordal.*

Proof. Let T be a tree. We want to prove that T^k is chordal for any positive integer k. Let $S(G)$ be the subdivision graph of a graph G as defined in Section 2.3, and $N^k_G(v)$ denotes the set $\{u \in G : d(u,v) \leq k\}$ where $d(u,v)$ is the distance between vertices u and v in G. Note that $uv \in G^k$ if and only if $N^k_{S(G)}(u) \cap N^k_{S(G)}(v) \neq \emptyset$, implying that G^k is the intersection graph of the family $S = \{N^k_{S(G)}(v) :$ for all $v \in G\}$. It follows that

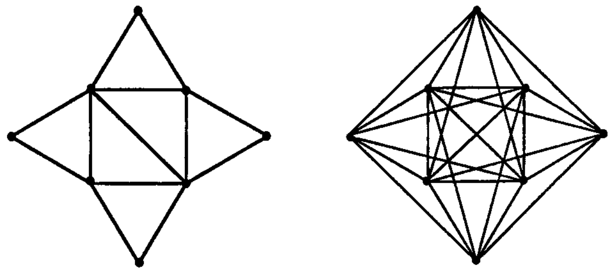

Figure 4: A chordal graph and its nonchordal square.

T^k is the intersection graph of the family $\{N^k_{S(T)}(v) : \text{ for all } v \in T\}$. Since $S(T)$ is a tree and $\langle N^k_{S(T)}(v)\rangle_{S(T)}$ is a subtree of $S(T)$ with center vertex v and radius k, T^k is the intersection graph of a family of subtrees of the tree $S(T)$. Thus T^k is chordal. ∎

Corollary 3.5 *The maximum clique in an arbitrary power of a tree can be determined in linear time.*

4 Conclusions

We have presented efficient algorithms for finding the square roots of graphs in three interesting special cases: tree squares, planar graphs and subdivided graphs. Further, we have studied the complexity of finding hamiltonian cycles and maximum cliques in powers of graphs. Several interesting open problems remain:

- What is the complexity of recognizing square graphs? We conjecture that the general problem is NP-complete.

- Let A be the adjacency matrix of a graph, made reflexive by adding a self-loop to each vertex. Given A^2, can we determine one of its $(0,1)$-matrix square roots in polynomial time? This problem is potentially easier than the finding the square root of a graph, since we are also given the number of paths of length at most two between each pair of vertices.

- What is the complexity of determining whether G^2 is hamiltonian, given G? Relevant results appear in [12, 14].

- What is the complexity of recognizing the squares of directed-acyclic graphs? Clearly, the square of DAG is a DAG. All square roots of a DAG G^2 contain the transitive reduction of G^2 as a subgraph. The time complexity of finding both the transitive closure and reduction of a digraph is equivalent to boolean matrix multiplication [1].

- Give an $o(m^2)$ algorithm for inverting total graphs.

5 Acknowledgments

We thank Gene Stark for providing a simpler proof of Theorem 3.4, and Alan Tucker for valuable discussions.

References

[1] A. Aho, M. Garey, and J. Ullman. The transitive reduction of a directed graph. *SIAM J. Computing*, 1:131–137, 1972.

[2] M. Behzad. A criterion for the planarity of a total graph. *Proc. Cambridge Philos. Soc.*, 63:679–681, 1967.

[3] F. Escalante, L. Montejano, and T. Rojano. Characterization of n-path graphs and of graphs having nth root. *J. Combin. Theory B*, 16:282–289, 1974.

[4] H. Fleischner. The square of every two-connected graph is Hamiltonian. *J. Combin. Theory B*, 16:29–34, 1974.

[5] M. R. Garey and D. S. Johnson. *Computers and Intractability – A guide to the Theory of NP-Completeness.* Freeman, New York, 1979.

[6] F. Gavril. The intersection graphs of subtrees in trees are exactly the chordal graphs. *J. Combin. Theory B*, 16:47–56, 1974.

[7] D. P. Geller. The square root of a digraph. *J. Combin. Theory*, 5:320–321, 1968.

[8] M. C. Golumbic. *Algorithmic Graph Theory and Perfect Graphs.* Academic Press, New York, 1980.

[9] F. Harary. *Graph Theory.* Addison-Wesley, Massachusetts, 1972.

[10] F. Harary, R. M. Karp, and W. T. Tutte. A criterion for planarity of the square of a graph. *J. Combin. Theory*, 2:395–405, 1967.

[11] F. Harary and A. Schwenk. Trees with hamiltonian square. *Mathematika*, 18:138–140, 1971.

[12] G. Hendry and W. Vogler. The square of a connected $S(K_{1,3})$-free graph is vertex pancyclic. *J. Graph Theory*, 9:535–537, 1985.

[13] J. E. Hopcroft and R. E. Tarjan. Dividing a graph into triconnected components. *SIAM J. Computing*, 2:135–158, 1973.

[14] M. Matthews and D. Summer. Hamiltonina results in $S(K_{1,3})$-free graphs. *J. Graph Theory*, 8:139–146, 1984.

[15] A. Mukhopadhyay. The square root of a graph. *J. Combin. Theory*, 2:290–295, 1967.

[16] I. C. Ross and F. Harary. The square of a tree. *Bell System Tech. J.*, 39:641–647, 1960.

[17] M. Sekanina. *On an ordering of the set of vertices of a connected graph.* Technical Report No. 412, Publ. Fac. Sci. Univ. Brno, 1960.