Integrated Intelligent Research (IIR)
International Journal of Computing Algorithm
Volume: 03 Issue: 03 December 2014 Pages: 236-239
ISSN: 2278-2397

Chordal Graphs and Their Clique Graphs

J.Arockia Aruldoss[1], P.Kalaivani[2]
[1,2]PG&Research Department Of Mathematics,
St. Joseph's College, Cuddalore, Tamilnadu, India
E-mail: aruligori@gmail.com, pkalaivanimsc90@gmail.com

Abstract - In this paper, we present a new structure for chordal graph. We have also given the algorithm for MCS(Maximal Cardinality Search) and lexicographic BFS(Breadth First Search) which is used in two linear time and space algorithm. Also we discuss how to build a clique tree of a chordal graph and the other is simple recognition procedure of chordal graphs.

Keywords - Chordal graph, MCS, Clique graph, BFS.

I. INTRODUCTION

Chordal graphs have been considered as the intersection graphs of subtrees of a tree. Chordal graphs are often represented by a clique tree. The structure of clique tree does not only appeared in the graph theory literature, but in context of ascyclic database schemes and in the context of spare matrix computations too.Chordal graphs can also be characterized using Perfect Elimination Orderings (PEO). A vertex is simplicial if and only if its neighbourhood is a complete subgraph. An elimination ordering $x_1, x_2, \ldots x_n$ is perfect if and only if each x_i is simplicial in the subgraph indued by $x_i, \ldots\ldots x_n$ A new structure namely the clique graph is introduced .In this paper some graph properties of this structure are studied with regard to clique trees. And the clique graph is justified as being the optimal structure containing all clique trees of a chordal graph.

II. THE CLIQUE GRAPH OF A CHORDAL GRAPH

We will introduce and study a new structure called the clique graph of a chordal graph. We will show the ties between clique graphs and the clique trees. In the clique is studied with regard to the clique intersection graph, which can be seen as the cliques hypergraph. The clique graph defined here is a subgraph of the clique intersection graph. We will prove that a clique graph can be seen as the minimal graph containing all clique trees. All graphs considered here are supposed to be connected, if not each connected component has to be considered separately.

A. Definition

Given an undirected graph G = (V,E), and two non-adjacent vertices a and b, a subset S⊂ V is an a, b – separator if the removal of S separators a and b in distinct connected components. If no proper subset of S is an a, b – separator then S is a minimal a, b – separator. A (minimal) separator is a set of vertices S for which there exist non adjacent vertices a and b such that S is a (minimal) a, b – separator.

B. Definition

Let $G = (V, E)$ be a chordal graph. The clique-graph of G, denoted by $C(G) = (V_c, E_c, \mu)$, with $\mu : E_c \to N$, is defined as follows:

1. The vertex set V_c, is the set of maximal cliques of G
2. The edge (C_1, C_2) belongs to E_c if and only if the intersection $C_1 \cap C_2$ is a minimal a, b – separator for each $a \in (C_1 \setminus C_2)$ and each $b \in (C_2 \setminus C_1)$;
3. The edges of $(C_i, C_j) \in V_c$ are weighted by the of the corresponding minimal separator $S_{ij} : \mu(C_i, C_j) = |s_{ij}|$.

Let C_i and C_j be two maximal cliques of a chordal graph. Hereafer, we will notes $S_{ij} = C_i \cap C_j$ if and only if S_{ij} is a minimal a, b – separator for each $a \in (C_i \setminus C_j)$ and each $b \in (C_i \setminus C_j)$. Let us now prove several structure properties of the clique graph.

C. Definition

Let G = (V,E) be a chordal graph. A clique tree of G is a tree $T_C = (C, F)$ such that C is the set of maximal cliques of G and for each vertex $x \in E$, the set of maximal cliques containing x induces a subtree of T_C .

Triangle Lemma-Let (C_1, C_2, C_3) be a 3-cycle in $C(G)$ and let S_{12}, S_{13}, S_{23} be the associated minimal separators of G. Then two of these three minimal separators are equal and included in the third.

Proof: Assume that two minimal separators among are in S_{12}, S_{13}, S_{23} comparable for the inclusion order. Let S_{12} and S_{13} be these minimal separators. Then there exist two vertices x and y such that $x \in (S_{12} \setminus S_{13})$ and $y \in (S_{13} \setminus S_{12})$ Since C_1, C_2, C_3 are distinct maximal clique, $C_2 \setminus C_3$ and $C_3 \setminus C_2$ are not empty. The vertices x and y do not belong to $C_2 \cap C_3$ For each $a \in (C_2 \setminus C_3)$ and each $b \in (C_3 \setminus C_2)$, the path a, x, y, b exist and is not cut by

$C_2 \cap C_3$. A contradiction therefore $C_2 \cap C_3$ is not a,b – separator and the edge between C_2 and C_3 does not exist. Therefore if there exists a 3-cycle in, then the three minimal separators on the edges can be linearly ordered by inclusion. Without loss of generality, assume that $S_{12} \subset S_{13} \subseteq S_{23}$. Therefore $S_{13} \subset C_1$ and $S_{13} \subset C_2$. And so $S_{13} \subset (C_1 \cap C_2)$ This leads to a contradiction : $S_{13} \subset S_{12}$. We have proved that $S_{12} = S_{13} = S_{23}$.
Note that the converse is false. Let C_1, C_2, C_3 be three maximal cliques such that $(C_1, C_2) \in E_c$ and $(C_1, C_3) \in E_c$. Then $S_{12} = S_{13}$ does not imply that the edge $(C_2, C_3) \in E_c$. But the following property stands.
Lemma1.2
Let T be a clique tree of the chordal graph G and let C_1 and C_2 be two adjacent maximal cliques. Then $C_1 \cap C_2$ is a minimal separator for all $a \in (C_1 \setminus C_2)$ and $b \in (C_2 \setminus C_1)$

III. GREEDY ASPECTS OF RECOGNITION ALGORITHMS

Let G=(V,E) be a graph with n vertices. When an elimination ordering is computed by BFS or MCS, another procedure must verify if it is a perfect elimination ordering in order to prove that G is triangulated. BFS or MCS computes the elimination ordering in the reverse order. In this section, we will give an explanation of the greedy aspect of the two linear recognition algorithms of chordal graphs: MCS and BFS. The main result proves that both algorithms compute a maximum spanning-tree of the clique graph.Let us examine how MCS and BFS. visit a chordal graph. We just give the proof for MCS, but this proof can be easily transformed for BFS. When MCS chooses a new vertex x , then the mark level of this vertex, noted mark (x), is maximum over all unnumbered vertices. The set of vertices who has marked x will be denoted by M(x).

Lemma- Let G = (V,E) be a chordal graph. In a execution of MCS or BFS on G, maximal cliques of G are visited consecutively.
Proof- Let α be the PEO computed by MCS. Let N = { x_n ,....., x_i } be the set of numbered vertices at some step. Then x_i is simplicial in G [x_n ,....., x_i], and so x_i belongs to a unique maximal clique. Let us prove that x_{i-1} belong to a new maximal clique iff mark (x_{i-1}) ≤ mark(x_i).

- Assume that (x_{i-1}) and x_i belong to the same maximal clique. Since x_i is simplicial in G [x_n ,....., x_i], all the vertices of M(x_i) belong to this maximal clique. Hence mark (x_{i-1}) = *mark* (x_i)+ 1.
- Assume that x_{i-1} belong to new maximal clique. . Since x_i was a vertex with the biggest level mark over all unnumbered vertices when it was chosen, *mark* (x_{i-1})≥ $|M(x_{i-1}) \setminus \{x_i\}|$ But there exist at least one vertex of the maximal clique containing x_i in G[x_n ,....., x_i] which does not mark (x_{i-1})≤ mark (x_i).

If we define the trace of α as the sequence of mark level of the vertices when they are number, each maximal clique is represented by an increasing sequence.We can conclude that the maximal cliques are visited consecutively. A similar argument holds for BFS.

Algorithm - Maximum weighted spanning-tree of clique graph

Data: A clique graph C(G)

Result: A maximum spanning-tree of C(G)
Choose a maximal clique C_1
For i=2 to n do
Choose a maximally (under inclusion) labeled edge adjacent to C_1 ,....., C_{i-1} to connect the new clique C_i

Theorem 2.1
Let G be a chordal graph ,then algorithm 1 computes a maximum weighted spanning tree (i.e a clique tree)of the clique graph $C(G)$.

Proof:
Let T_I be the tree built by algorithm 1 at step i .We now prove by induction the following

invariant: can be completed as a maximum spanning tree of $C(G)$.

Clearly the property holds for T_1. Let us suppose by induction, it hold for T_{i-1} and that (C_i, C_j), with $j < i$ is the edge chosen by the algorithm at step i. Therefore it exists a maximum spanning tree T containing T_{i-1}.

If (C_j, C_i) belongs to T, we have finished. Else it exists a unique path μ =[$C_j = D_1$,...., $D_k = C_i$]from C_j to C_i in T.

Let D_h be the last vertex of C_1,.... C_{i-1} belonging to μ.algorithm 1 is insures that A=*label*[(D_h, D_{h+1})] does not contain B= *label*[(C_j, C_i)].If ∃ b∈ $B - A$,then b∈ C_j and therefore by the definition of clique tree,b must also belong to all cliques in μ,a contradiction.

Therefore A=B, and a maximum spanning tree T' can be obtained from T by exchanging the edges (D_h, D_{h+1}) and (C_j, C_i), which finishes the proof.

Volume: 03 Issue: 03 December 2014 Pages: 236-239
ISSN: 2278-2397

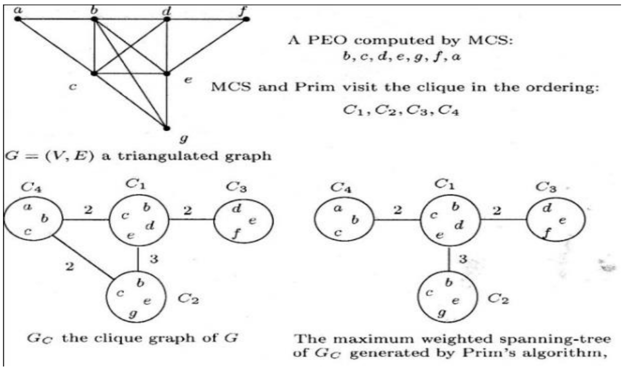

Figure 1 MCS and BFS compute maximum spanning trees of CC(G).

IV. LINEAR ALGORITHMS ON THE CLIQUE TREES

In this section, we will first present a linear time space algorithm which computes a clique tree of G will be presented. This algorithm based on MCS. A similar algorithm was presented in we defined the trace of an PEO α computed by MCS as the sequence of mark level of the vertices when they are number. algorithm for the recognition of PEO is presented.

Lemma -Let G be a chordal graph. Let T be a tree whose nodes are the maximal cliques of G, and such that an edge between the maximal cliques C_j and $C_k (k < j)$ exists if and only if $mark_j \in L_k$. Then T is a clique-tree of G.

Theorem -The algorithm 2 computer a PEO and its associated clique-tree if and only if the input graph G=(V,E) is chordal. The complexity of this algorithm is O(n+m) where n= $|V|$ and m= $|E|$.

Proof - If the algorithm 2 computes a PEO and a clique-tree, then the input graph is trivially chordal. The algorithm 2 is an extended version of MCS, hence if G is chordal, the computed elimination ordering is perfect. By lemma, the step1 build a clique-tree if G is chordal.Let us have a look at the size of a clique tree. First of all, when a vertex is marked by MCS, this mark corresponds to an edge. Hence the size of all mark sets is O(m), the number of edges in G. Since there is at most n increasing sequence,

T contains at most n nodes and so O(n) edges.

Let us examine the size of the set of nodes in T. Since the vertices of minimal separation belong to several maximal cliques, the size of the nodes set is bigger than n. Let M(x) be the set of marks of x, where x is the first vertex of an

Algorithm -2: Maximum Cardinality Search and Clique-Tree
Data: A graph G=(V,E)
Result: If the input graph is chordal: a PEO and an associated clique-tree
T=(I,F) where I is the set of maximal cliques

```
Begin
    each vertex of X is initialized with the
    empty set
    Previousmark =-1
    j=0
  for i=n to 1 do
      choose a vertex x not yet number
      such that |mark (x)| is maximum
      if mark (x) ≤ previousmark then
            j=j+1
            create a maximal clique
            Cj=M(x) ∪ {x}
            create the tie between Cj
            and C(last(x))
      else
            Cj=Cj∪{x}
      For each y neighbor of x do
        M(y)=M(y) ∪ {x}
            Mark (y)=mark(y)+1
            last(y)=x
      previousmark = mark(x)
      x is numbered by i
      C(x)=j
end
```

Increasing sequence L_j. The node associated to x by the algorithm 2 is $M(x) \cup_j L_j$ .But M(x) corresponds to the edges between vertices of $M(x)$ and x . since every duplicated vertex belongs to the mark set of the first vertex of an increasing sequence, each of them can be associated an edge. Therefore the sum of the cardinalitry of the nodes of T is smaller than n+m, and the space complexity is O(n+m).All the operations of step 1, 2 and 3 can be done in constant time. Hence this algorithm has the same time complexity as MCS: O(n+m)

V. CONCLUSION

The clique graph of a chordal graph G introduced and studied in the first part,is shown to be the minimal structure containing all clique trees of G. The properties of the clique graph imply a simple algorithm for maximum spanning –trees which cannot be applied for general graphs. As in MCS is compared to Prim's algorithm.This paper presents a new and unified regard on as the MCS and BFS algorithms.. this approach ,studying the trace of algorithm like MCS and the underlying structure, can be very helpful for various generalizations of elimination orderings or generalizations of chordal graphs.

REFERENCES

[1] Beeri, R.Fagin, D.Maier, and M.Yannakakis.On the desireability of acyclic database and schemes.J.Assoc.Comput.,30:479-513,1983.
[2] C.Berge.Hypergraphs.North Hollands,1989. J.R.S. Blair and B.Peyton. An introduction to chordal graphs and clique trees.preprint.

[3] Brandstadt, F.F.Dragan,V.D.Chepoi, and V.I.Voloshin. Dually chordal graphs. In proceedings of the 19 thInter.Workshop on Graph-TheoreticConcept in Computer Science,1993.WG93.
[4] PBuneman.A characterization of rigid circuit graphs. Discrete Math.,9:205-212,1974.
[5] Dahlhaus,P.L.Hammer,F.Maffray,andS.Olariu.On domination elimination orderings and domination graphs.Technical Report 27-94,Rutgers University Centre of Operations Research,P.O.Box 5062,New Brunswick,NewJersey,USA,August 1994.
[6] A.Dirac. On rigid circuit graphs.Abh.Math.Sem.Uni.Hamburg 25,1961.
[7] Gavril.The intersection graphs of a path in a tree are exactly the chordal graphs.Journ.Comb.Theory,16:47-56,1974.
[8] yan B.Hayward. Weakly triangulated graphs. Journal of Combinatorial theory,39:200-209,1985.Serie B
[9] Krote, L.Lovasz, and R.Schrader. Greedoids. Number 4 in the algorithms and Combinatories.Springer Verlag,1991.
[10] G.Lewis,B.W.Peyton, and A.Pothen.A fast algorithm for reordering sparse matrices for parallel factorization. SIAM J.Sci.Stat.Comput.,10(6):1146-1173,November 1989.
[11] Olariu. Some aspects of the semi-perfect elimination.Discrete Applied Mathematics,31:291-298,1991.