Tree Decomposition for k-Chordal Graph and Its Applications

Adrian Kosowski[1 †] and Bi Li[23 ‡] and Nicolas Nisse[2 §] and Karol Suchan [4 ¶]

[1]*INRIA Bordeaux Sud-ouest & CEPAGE Projet, LaBRI, 351 cours de la libération, F-33400 Talence Cedex, France*
[2]*INRIA & MASCOTTE, 2004, route des Lucioles B.P. 93, F-06902 Sophia Antipolis Cedex, France*
[3]*CAS & AAMS, 55 Zhongguancun East Road, Haidian District, 100190 Beijing, China*
[4]*Universidad Adolfo Ibáñz, Facultad de Ingenieria y Ciencias, Av. Diagonal Las Torres 2640, Peñalolén 7941169, Santiago, Chile*

It is interesting to find the tree-decomposition with bounded treewidth since many NP-hard problems have been shown to be polynomial or even linear time solvable when restricted to the graph with bounded treewidth, given together with a tree-decomposition. In this paper, we prove that for any graph G and an integer $k \geq 3$, we can find, in $|V(G)|^2$ time, either an induced cycle larger than k or a tree-decomposition with treewidth bounded by $(k-1)(\Delta-1)+2$ such that every bag is a caterpillar with backbone chordless path of order at most $k-1$, where Δ is the maximum degree. Moreover, we get that the graph is k-hyperbolic if we find the tree-decomposition. Then we apply the tree-decomposition to design a compact routing scheme for k-chordal graph using address, routing table and message header of size $O(k \log n)$ and the additive stretch $4k-2+(4k-6)\log\Delta$.

Keywords: Tree-decomposition, Compact routing algorithm, k-chordal graph

1 Introduction

Tree-decomposition and treewidth, introduced by Robertson and Seymour [RS84], play an important role in the theory of graph minors. They are popular tools widely applied to solve various problems in many different fields of computer science. Many NP-hard problems have been shown to be linear time solvable for graph with bounded treewidth, given together with a proper tree decomposition. Thus it is interesting to find tree-decomposition and to determine the treewidth in the graph.

Simply speaking, a tree-decomposition is a mapping of a graph into a tree that can be used to speed up solving certain problems on the original graph. The treewidth measures the number of graph vertices mapped onto any tree node in an optimal tree decomposition. Typically, there are many dynamical programming algorithm, based on a given tree-decomposition with bounded treewidth, using time polynomial in the number of vertices, but exponential in the treewidth of the graph, e. g., the maximum independent set problem [CM93, Bod93b]. So it is important to find a tree-decomposition with small treewidth to design the algorithm for hard problems.

Routing problem is one of the practical problems, in which tree-decomposition is applied [Dou05]. With more and more complicated networks in our life, it is a basic activity to deliver a message from a sender to a receiver. To do this, we need to find a path between the pair. If every sender knows the shortest path to any receiver, then it is easy to deliver messages. However, in the context of large scale networks, it is impossible for each processor to store all the information to take the shortest path to go to any of the processors since it requires too big memory space. So the natural desirability is to find the path as short as possible with limited information. This is the compact routing problem and the solution is a compact routing scheme, including the routing information stored in processors' local memory (*routing table*) and the routing function deciding

[†] I don't know.
[‡] I don't know how to write here.
[§] But he knows.
[¶] I don't know.

the edge to take. The efficiency of a routing scheme is measured by its *multiplicative stretch factor* (resp., *additive stretch factor*, i.e., the maximum ratio (resp., difference) between the length of a route computed by the scheme and that of a shortest path connecting the same pair of nodes. Considering the space complexity of the routing scheme, it is important to have a routing table with small size, for which tree-decomposition can be apply.

Our Contributions. First, for any graph G and an integer $k \geq 3$, we can find, in $|V(G)|^2$ time, either an induced cycle larger than k or a tree-decomposition with treewidth bounded by $(k-1)(\Delta-1)+2$ such that every bag induces a caterpillar with backbone chordless path of order at most $k-1$, where Δ is the maximum degree. So for k-chordal graph, which contains no induced cycle larger than k, we can always find the tree-decomposition with nice structure and bounded treewidth. This improves the result of Bodlaender and Thilikos, who bounded the treewidth by $\Delta(\Delta-1)^{k-3}$ [BT97]. Moreover, we get that G is k-hyperbolic if we find the tree-decomposition. Secondly, we apply our tree-decomposition to design a compact routing scheme for k-chordal graph. For each node the routing table, the address and the header of the message are all of size $O(k \log n)$, which is a big improvement of $O(\log^2 n)$ given by Dourisboure [Dou05], and the additive stretch is $4k-2+(4k-6)\log\Delta$. When we consider the interval graph, a subclass of chordal graph, the additive stretch is 10.

Related Work. There are lots of papers working on determining the treewidth and finding the optimal tree-decompositions. Given a graph G and an integer k, it is NP-complete to decide whether the treewidth of G is at most k or not [ACP87]. Bodlaender [Bod93a] gave a linear time algorithm, that either outputs a tree-decomposition of G with treewidth at most k or determine that the treewidth of G is larger than k, for all constant k. For chordal graph, cographs [BM93], circular arc graphs [SSR94], chordal bipartite graphs [KK95] and etc., the treewidth problem is polynomial solvable. Bodlaender and Thilikos [BT97] proved that the treewidth problem is NP-complete for graphs with small maximum degree. Also for k-chordal graph, they bounded the treewidth with k and maximum degree Δ by $\Delta(\Delta-1)^{k-3}$. Chordality and hyperbolicity are both parameters measuring the tree-likeness of a graph. There have been some papers working on the relation between the chordality and hyperbolicity of graph [BC03, WZ11]. The problem of computing the chordality of a graph is NP-complete : Let G be an induced subgraph of a grid. Let H be obtained from G by subdividing all its edges once. There is a correspondence between any induced cycle in H and any cycle in G. Since computing the longest cycle in subgrid is NP-complete [IPS82], the result follows. Finding the longest induced path is $W[2]$-complete [CF07] and the problem is Fixed Parameter Tractable in planar graphs [KK09]. It is coNP-hard to decide wether the given graph G is k-chordal or not for $k = \Theta(|V(G)|)$ [Ueh99].

For routing problem, there is a series of papers investigated the tradeoff between the stretch and the size of the routing table. For any n node graph, [PU89] shows that every routing scheme with s stretch requires at least $\Omega(n^{1/(2s+4)})$ for local routing table. For trees, [FG01] gives the optimal labeled routing scheme with $O(\log n)$ local routing table. For k-chordal graph, [Dou05] designs a routing scheme with $O(\log^2 n)$ routing table and additive stretch $k+1$. And in [NSR08], there is a routing scheme with $O(\Delta \log n)$ routing table and additive stretch $k-1$.

2 Tree-decomposition

In this section, for a given a graph $G=(V,E)$ and an integer $k \geq 3$, we prove that we can find either an induced cycle larger than k in G or a tree-decomposition with nice structure and bounded treewidth for G. Moreover, we discuss about the hyperbolicity of the graph.

Definition 1 A tree-decomposition of a graph $G=(V,E)$ is a pair $(\{X_i | i \in I\}, T=(I,M))$, where T is a tree and $\{X_i | i \in I\}$ is a family of subsets, called bags, of vertices of G such that

- $V(G) = \cup_{i \in I} X_i$;
- for every edge $uv \in E(G)$ there exists (at least) one $i \in I$ such that $u, v \in X_i$;
- for every vertex $v \in V(G)$, $\{i \in I | v \in X_i\}$ induces a (connected) subtree of T.

The treewidth of a tree-decomposition $(\{X_i | i \in I\}, T=(I,M))$ is $\max_{i \in I} |X_i| - 1$. The treewidth of a graph G is the minimum treewidth over all possible tree-decomposition of G.

Definition 2 A caterpillar is a graph consists of a chordless path, called backbone, and any set of leaves attached to this path.
Definition 3 A graph $G = (V,E)$ is k-hyperbolic if it satisfies : taken any three vertices $x,y,z \in V(G)$, for any vertex $v \in P_{xy}$, $d_G(v,u) \leq k$ for any $u \in P_{xz} \cup P_{yz}$ where P_{xy}, P_{xz}, P_{yz} are any shortest path between x,y, x,z and y,z respectively.
Notation : For graph $G = (V,E)$, $N_G(v)$ is the set of the neighbors of v. $N_G(S) = \cup_{v \in S} N_G(v) \setminus S$ and $N_G[S] = N_G(S) \cup S$ for subset $S \subseteq V$.

Theorem 1 *Given a graph $G = (V,E)$ and an integer $k \geq 3$, we have the following results :*
(i) we can find either an induced cycle with order larger than k or a tree-decomposition $(\{X_i | i \in I\}, T = (I,M))$ with treewidth bounded by $(k-1)(\Delta-1)+2$ such that X_i induces a caterpillar with backbone a chordless path P_i of order at most $k-1$, i. e. for $v \in X_i$, we have $v \in N_G[P_i]$, $i \in I$. Moreover, the time complexity is $O(|V(G)|^2)$;
(ii) we claim that either G is not k-chordal or G is k-hyperbolic.

Sketch of proof. (i) The proof is by induction on $|V(G)|$. We prove that either we find an induced cycle larger than k, or for any chordless path $P = \{v_1, \ldots, v_i\}$ with $i \leq k-1$, there is a tree-decomposition for G with one bag containing $N_G[P]$ and every bag inducing a caterpillar with backbone a chordless path of order at most $k-1$. (ii) It is easily obtained from (i).

Then it is easy to obtain the following corollary for k-chordal graph.

Corollary 1 *Given a k-chordal graph $G = (V,E)$, we can find a tree-decomposition $(\{X_i | i \in I\}, T = (I,M))$ such that X_i induces a caterpillar with backbone a chordless path P_i of order at most $k-1$, i. e. for $v \in X_i$, we have $v \in N_G[P_i]$, $i \in I$. Moreover, the time complexity is $O(|V(G)|^2)$.*

3 Routing Scheme for k-Chordal Graph

In this section, applying the special tree-decomposition in Corollary1, we give a routing scheme for k-chordal graph which uses message headers, addresses and local memories of size $O(k \log n)$ bits per node, and the additive stretch $4k-2+(4k-6)\log\Delta$, where Δ is the maximum degree.

Let G be a k-chordal graph. Let F be a BFS-tree of G rooted at r and $(\{X_i | i \in I\}, T = (I,M))$ be a tree-decomposition of G from Corollary1. We need some notations introduced in [FG01] to construct the addresses and routing tables. Since of limited space, we just describe the main idea of the routing algorithm.
Main Idea of the Routing Algorithm Let s,d be two vertices of G, s the sender and d the receiver. If s is the ancestor of d in the BFS-tree F, then we can use the algorithm in [FG01] and obtain the shortest path from s to d. In the following, we consider the case that s is not the ancestor of d in F. Firstly, We route along the path from s to r in the tree F until we arrive the first node u such that B_u is an ancestor of B_d in T, where B_u (B_d) is the bag containing u (d) with minimum identifier [FG01] in the tree-decomposition T. (The node u exists, since $u = r$ satisfies at least.) Then in the bag B_u, there is a node a, which is an ancestor of d in tree F, since B_u either is an separator of d and r or contains at least one of d and r. So we search a in the bag B_u after arriving u. Then we route from a to d in the tree F.
Remark For interval graph, subclass of chordal graph, we have a caterpillar BFS-tree F. Then we can prove that our routing algorithm obtain the additive stretch 2.

Références

[ACP87] Stefan Arnborg, Derek G. Corneil, and Andrzej Proskurowski. Complexity of finding embeddings in a k-tree. *SIAM J. Algebraic Discrete Methods*, 8 :277–284, April 1987.

[BC03] Hans-Jürgen Bandelt and Victor Chepoi. 1-hyperbolic graphs. *SIAM J. Discrete Math.*, 16(2) :323–334, 2003.

[BM93] Hans L. Bodlaender and Rolf H. Möhring. The pathwidth and treewidth of cographs. *SIAM J. Discrete Math.*, 6(2) :181–188, 1993.

[Bod93a] Hans L. Bodlaender. A linear time algorithm for finding tree-decompositions of small treewidth. In *STOC*, pages 226–234, 1993.

[Bod93b] Hans L. Bodlaender. A tourist guide through treewidth. *Acta Cybern.*, 11(1-2) :1–22, 1993.

[BT97] Hans L. Bodlaender and Dimitrios M. Thilikos. Treewidth for graphs with small chordality. *Discrete Applied Mathematics*, 79(1-3) :45–61, 1997.

[CF07] Yijia Chen and Jörg Flum. On parameterized path and chordless path problems. In *22nd Annual IEEE Conference on Computational Complexity (CCC 2007), 13-16 June 2007, San Diego, California, USA*, pages 250–263. IEEE Computer Society, 2007.

[CM93] Bruno Courcelle and Mohamed Mosbah. Monadic second-order evaluations on tree-decomposable graphs. *Theor. Comput. Sci.*, 109(1&2) :49–82, 1993.

[Dou05] Y. Dourisboure. Compact routing schemes for generalised chordal graphs. *Journal of Graph Algorithms and Applications*, 9(2) :277–297, 2005.

[FG01] P. Fraigniaud and C. Gavoille. Routing in trees. In *Proceedings of the 28th International Colloquium on Automata, Languages and Programming (ICALP)*, pages 757–772, 2001.

[IPS82] Alon Itai, Christos H. Papadimitriou, and Jayme Luiz Szwarcfiter. Hamilton paths in grid graphs. *SIAM J. Comput.*, 11(4) :676–686, 1982.

[KK95] Ton Kloks and Dieter Kratsch. Treewidth of chordal bipartite graphs. *J. Algorithms*, 19(2) :266–281, 1995.

[KK09] Yusuke Kobayashi and Ken-ichi Kawarabayashi. Algorithms for finding an induced cycle in planar graphs and bounded genus graphs. In *roceedings of the Twentieth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2009, New York, NY, USA, January 4-6, 2009*, pages 1146–1155. SIAM, 2009.

[NSR08] N. Nisse, K. Suchan, and I. Rapaport. Distributed computing of efficient routing schemes in generalized chordal graphs. Technical Report CMM-B-08/10-220, CMM, oct 2008. http ://www-sop.inria.fr/members/Nicolas.Nisse/publications/distribRouting.pdf.

[PU89] D. Peleg and E. Upfal. A trade-off between space and efficiency for routing tables. *Journal of the ACM*, 36(3) :510–530, 1989.

[RS84] Neil Robertson and Paul D. Seymour. Graph minors. iii. planar tree-width. *J. Comb. Theory, Ser. B*, 36(1) :49–64, 1984.

[SSR94] Ravi Sundaram, Karan Sher Singh, and C. Pandu Rangan. Treewidth of circular-arc graphs. *SIAM J. Discrete Math.*, 7(4) :647–655, 1994.

[Ueh99] Ryuhei Uehara. Tractable and intractable problems on generalized chordal graphs. Technical Report COMP98-83, IEICE, 1999. http ://www.jaist.ac.jp/ uehara/pub/tech.html.

[WZ11] Yaokun Wu and Chengpeng Zhang. Hyperbolicity and chordality of a graph. *Electr. J. Comb.*, 18(1), 2011.