Discrete Applied Mathematics 164 (2014) 444–449

Contents lists available at ScienceDirect

Discrete Applied Mathematics

journal homepage: www.elsevier.com/locate/dam

Contracting chordal graphs and bipartite graphs to paths and trees☆

Pinar Heggernes[a], Pim van 't Hof[a,*], Benjamin Lévêque[b], Christophe Paul[b]

[a] Department of Informatics, University of Bergen, P.O. Box 7803, N-5020, Bergen, Norway

[b] CNRS, LIRMM, Université Montpellier 2, France

ARTICLE INFO

Article history:
Received 20 September 2011
Received in revised form 13 February 2013
Accepted 20 February 2013
Available online 15 March 2013

Keywords:
Edge contractions
Chordal graphs
Graph modification problems
Polynomial time algorithms

ABSTRACT

We study the following two graph modification problems: given a graph G and an integer k, decide whether G can be transformed into a tree or into a path, respectively, using at most k edge contractions. These problems, which we call Tree Contraction and Path Contraction, respectively, are known to be NP-complete in general. We show that on chordal graphs these problems can be solved in $O(n+m)$ and $O(nm)$ time, respectively. As a contrast, both problems remain NP-complete when restricted to bipartite input graphs.

1. Introduction

Graph modification problems play a central role in algorithmic graph theory, not in the least because they can be used to model many graph theoretical problems that appear in practical applications [15–17]. The input of a graph modification problem is an n-vertex graph G and an integer k, and the question is whether G can be modified in such a way that it satisfies some prescribed property, using at most k operations of a given type. Famous examples of graph modification problems where only vertex deletion is allowed include Feedback Vertex Set, Odd Cycle Transversal, and Chordal Deletion. In problems such as Minimum Fill-In and Interval Completion, the only allowed operation is edge addition, while in Cluster Editing both edge additions and edge deletions are allowed.

Many classical problems in graph theory, such as Clique, Independent Set and Longest Induced Path, take as input a graph G and an integer k, and ask whether G contains a vertex set of size at least k that satisfies a certain property. Many of these problems can be formulated as graph modification problems: for example, asking whether an n-vertex graph G contains an independent set of size at least k is equivalent to asking whether there exists a set of at most $n-k$ vertices in G whose deletion yields an edgeless graph. Some important and well studied graph modification problems ask whether a graph can be modified into an *acyclic graph* or into a *path*, using at most k operations. If the only allowed operation is vertex deletion, these problems are widely known as Feedback Vertex Set and Longest Induced Path, respectively. The problem Longest Path can be interpreted as the problem of deciding whether a graph G can be turned into a path by deleting edges and isolated vertices, performing at most k deletions in total. All three problems are known to be NP-complete on general graphs [7].

☆ This work is supported by the Research Council of Norway and by EPSRC UK grant EP/D053633/1.

* Corresponding author. Tel.: +47 90028811.

E-mail addresses: pinar.heggernes@ii.uib.no (P. Heggernes), pim.vanthof@ii.uib.no (P. van 't Hof), leveque@lirmm.fr (B. Lévêque), paul@lirmm.fr (C. Paul).

0166-218X/$ – see front matter © 2013 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.dam.2013.02.025

We study two graph modification problems in which the only allowed operation is *edge contraction*. The edge contraction operation plays a key role in graph minor theory, and it also has applications in Hamiltonian graph theory, computer graphics, and cluster analysis [13]. The problem of contracting an input graph G to a fixed target graph H has recently attracted a considerable amount of interest, and several results exist for this problem when G or H belong to special graph classes [2–4,11–14]. The two problems we study in this paper, which we call Tree Contraction and Path Contraction, take as input an n-vertex graph G and an integer k, and the question is whether G can be contracted to a tree or to a path, respectively, using at most k edge contractions. Since the number of connected components of a graph does not change when we contract edges, the answer to both problems is "no" when the input graph is disconnected. Note that contracting a connected graph to a tree is equivalent to contracting it to an acyclic graph. Previous results easily imply that both problems are NP-complete in general [1,4]. Very recently, it has been shown that Path Contraction and Tree Contraction can be solved in time $2^{k+o(k)} + n^{O(1)}$ and $4.98^k \cdot n^{O(1)}$, respectively [9].

We show that the problems Tree Contraction and Path Contraction can be solved on chordal graphs in $O(n + m)$ and $O(nm)$ time, respectively. It is known that Tree Contraction is NP-complete on bipartite graphs [9], and we show that the same holds for Path Contraction. To relate our results to previous work, we would like to mention that Feedback Vertex Set and Longest Induced Path can be solved in polynomial time on chordal graphs [5,19]. However, it is easy to find examples that show that the set of trees and paths that can be obtained from a chordal graph G by at most k edge contractions might be completely different from the set of trees and paths that can be obtained from G by at most k vertex deletions. As an interesting contrast, Longest path remains NP-complete on chordal graphs [8]. An extended abstract of this paper has appeared in the proceedings of the 6th Latin-American Algorithms, Graphs and Optimization Symposium (LAGOS 2011) [10].

2. Definitions and notation

All the graphs considered in this paper are undirected, finite and simple. We use n and m to denote the number of vertices and edges of the *input* graph of the problem or the algorithm under consideration. Given a graph G, we denote its vertex set by $V(G)$ and its edge set by $E(G)$. The (open) *neighborhood* of a vertex v in G is the set $N_G(v) = \{w \in V(G) \mid vw \in E(G)\}$ of neighbors of v in G. The *degree* of a vertex v in G, denoted by $d_G(v)$, is $|N_G(v)|$. The *closed neighborhood* of v is the set $N_G[v] = N_G(v) \cup \{v\}$. For any set $S \subseteq V(G)$, we write $N_G(S) = \cup_{v\in S} N_G(v) \setminus S$ and $N_G[S] = N_G(S) \cup S$. A subset $S \subseteq V(G)$ is called a *clique* of G if all the vertices in S are pairwise adjacent. A vertex v is called *simplicial* if the set $N_G[v]$ is a clique. For any set of vertices $S \subseteq V(G)$, we write $G[S]$ to denote the subgraph of G *induced* by S. If the graph $G[S]$ is connected, then the set S is said to be *connected*. We say that two disjoint sets $S, S' \subseteq V(G)$ are *adjacent* if there exist vertices $s \in S$ and $s' \in S'$ such that $ss' \in E(G)$. For any set $S \subseteq V(G)$, we write $G - S$ to denote the graph obtained from G by removing all the vertices in S and their incident edges. If $S = \{s\}$, we simply write $G - s$ instead of $G - \{s\}$.

The *contraction* of edge $e = uv$ in G removes u and v from G, and replaces them by a new vertex, which is made adjacent to precisely those vertices that were adjacent to at least one of the vertices u and v. Instead of speaking of the contraction of edge uv, we sometimes say that a vertex u is *contracted on* v, in which case we use v to denote the new vertex resulting from the contraction. Let $S \subseteq V(G)$ be a connected set. If we repeatedly contract a vertex of $G[S]$ on one of its neighbors in $G[S]$ until only one vertex of $G[S]$ remains, we say that we *contract S into a single vertex*. We say that a graph G can be *k-contracted* to a graph H, with $k \le n - 1$, if H can be obtained from G by a sequence of k edge contractions. Note that if G can be k-contracted to H, then H has exactly k fewer vertices than G has. We simply say that a graph G can be *contracted* to H if it can be k-contracted to H for some $k \ge 0$. Let H be a graph with vertex set $\{h_1, \ldots, h_{|V(H)|}\}$. Saying that a graph G can be contracted to H is equivalent to saying that G has a so-called *H-witness structure* $\mathcal{W}$, which is a partition of $V(G)$ into *witness sets* $W(h_1), \ldots, W(h_{|V(H)|})$ such that each witness set is connected, and such that for every two $h_i, h_j \in V(H)$, witness sets $W(h_i)$ and $W(h_j)$ are adjacent in G if and only if h_i and h_j are adjacent in H. By contracting each of the witness sets into a single vertex, which can be done due to the connectivity of the witness sets, we obtain the graph H. An H-witness structure of G is, in general, not uniquely defined (see Fig. 1).

If H is a subgraph of G and $v \in N_G(V(H))$, then we refer to the vertices in $N_G(v) \cap V(H)$ as the *H-neighbors* of v. The *distance* $d_G(u, v)$ between two vertices u and v in G is the number of edges in a shortest path between u and v, and $diam(G) = \max_{u,v\in V(G)} d_G(u, v)$. For any two vertices u and v of a path P in G, we write uPv to denote the subpath of P from u to v in G. We use P_ℓ to denote the graph isomorphic to a path on ℓ vertices, i.e., P_ℓ is the graph with ordered vertex set $\{p_1, p_2, p_3, \ldots, p_\ell\}$ and edge set $\{p_1p_2, p_2p_3, \ldots, p_{\ell-1}p_\ell\}$. Similarly, C_ℓ denotes the graph that is isomorphic to a cycle on ℓ vertices, i.e., C_ℓ is the graph with ordered vertex set $\{c_1, c_2, c_3, \ldots, c_\ell\}$ and edge set $\{c_1c_2, c_2c_3, \ldots, c_{\ell-1}c_\ell, c_\ell c_1\}$. A graph is *chordal* if it does not contain a chordless cycle on at least four vertices as an induced subgraph.

3. Contracting chordal graphs

In this section, we show that Tree Contraction and Path Contraction can be solved in polynomial time on chordal graphs. It is easy to see that the class of chordal graphs is closed under edge contractions, and we use this observation throughout this section.