Discrete Optimization 18 (2015) 168–188

Contents lists available at ScienceDirect

Discrete Optimization

www.elsevier.com/locate/disopt

A branch-and-price-and-cut method for computing an optimal bramble

Sibel B. Sonuc[a,1], J. Cole Smith[b,*], Illya V. Hicks[c,2]

[a] Özyeğin University, Department of Industrial Engineering, Nişantepe Mah. Orman Sk. Çekmeköy, İstanbul, Turkey

[b] Clemson University, Department of Industrial Engineering, 100-B Freeman Hall, Clemson, SC 29634-0920, United States

[c] Rice University, Computational and Applied Mathematics Department, 6100 Main St. - MS 134, Houston, TX 77005-1892, United States

A R T I C L E I N F O

Article history:
Received 1 April 2015
Received in revised form 17 September 2015
Accepted 21 September 2015
Available online 6 November 2015

Keywords:
Bramble
Branch-and-price
Treewidth
Integer programming

A B S T R A C T

Given an undirected graph, a bramble is a set of connected subgraphs (called bramble elements) such that every pair of subgraphs either contains a common node, or such that an edge (i, j) exists with node i belonging to one subgraph and node j belonging to the other. In this paper we examine the problem of finding the bramble number of a graph, along with a set of bramble elements that yields this number. The bramble number is the largest cardinality of a minimum hitting set over all bramble elements on this graph. A graph with bramble number k has a treewidth of $k-1$. We provide a branch-and-price-and-cut method that generates columns corresponding to bramble elements, and rows corresponding to hitting sets. We then examine the computational efficacy of our algorithm on a randomly generated data set.

1. Introduction

The concept of brambles is first introduced by Seymour and Thomas [1], who also show that a graph having bramble number k has a *treewidth* of $k - 1$. Following this result, determining a graph's bramble number received attention from many researchers. Bodlaender et al. [2] provide a heuristic for computing brambles on general graphs (and on planar graphs with slight modification of the algorithm), where brambles are constructed such that they potentially give the best bramble order while keeping the computation and construction times reasonable. Many other studies focus on efficient algorithms for specially structured graphs. On graphs that can be formed as Cartesian products of complete graphs, Lucena [3] finds lower

* Corresponding author. Tel.: +11 864 656 4716.
E-mail addresses: sibel.sonuc@ozyegin.edu.tr (S.B. Sonuc), jcsmith@clemson.edu (J.C. Smith), ivhicks@rice.edu (I.V. Hicks).
1 Tel.: +90 216 564 9322.
2 Tel.: +11 713 348 5667.

http://dx.doi.org/10.1016/j.disopt.2015.09.005
1572-5286/

bounds on treewidths by using brambles. This approach employs a method to generate brambles for such graphs so that computing the orders of these brambles is simplified. Birmelé et al. [4] use brambles to show that a k-prism has a treewidth of $O(k^2)$. By the Graph Minor Theorem [5], there exists a function f such that for every graph with treewidth at least k, there is a bramble of order $k+1$ and size $f(k)$. Grohe and Marx [6] show that there is an exponential lower bound for the function f, and opens the question of the existence of an exponential upper bound on f. Grigoriev et al. [7] compute brambles in grid minors to provide lower bounds on planar graphs.

In this paper we provide an optimal integer programming algorithm to compute a general graph's bramble number. Finding optimal brambles may be beneficial for clustering or vertex covering problem types where the subsets must have some type of pairwise "connectivity" or hierarchical structure. Several different clustering methods and connectivity measures have been studied in the operations research literature. Mokken [8] introduces the concepts of *cliques*, *clubs*, and *clans*, each of which provides a different definition of connectivity and a different structure of clustered subgraphs. Bourjolly et al. [9] present an exact branch-and-bound algorithm to solve the maximum k-club problem. Veremyev and Boginski [10] extend the definition of clubs with *R-robust k-clubs*, and provide a compact formulation. Another class of clique relaxations are *k-plexes* [11]. Balasundaram et al. [12] introduce the maximum k-plex problem, and their work is further extended by McClosky and Hicks [13]. The maximum vertex packing, or independent set, problem is the same as the clique problem on the complement of a graph. Relaxations of the vertex packing problem have been studied in [14,15].

A survey of clique-defining properties exploited in clique relaxation models is compiled by Pattillo et al. [16], where the authors also propose a classification scheme for the resulting clusters. Mehrotra and Trick [17] use a branch-and-price algorithm for the maximum weighted clique problem, and Ji and Mitchell [18] extend this work by introducing minimum clique size constraints. Fellows et al. [19] study clusters that overlap by either sharing an edge or a node, and present their identifying characteristics along with the computational complexity of determining clusters with overlaps. In this paper, we refer to such pair of clusters as *touching* instead of overlapping. Feremans et al. [20] review different clustering problems including the generalized minimum clique problem, in which the optimal solution consists of a subgraph with vertices defined by a hitting set over a set of clusters.

Vertex covering is similar to vertex partitioning with the relaxation that vertices may be shared by subsets; this version of vertex covering is not to be confused with the more known vertex cover problem. For example, in wireless networks, vertex partitioning algorithms have been utilized to cluster "interfering nodes" together for better scheduling of the network to achieve higher data rates [21]. Gerber and Kobler [22] solve the vertex partitioning problem on classes of graphs having bounded clique-width, and in [23] the authors introduce the *satisfactory partitioning* problem. A survey of results and complexities of the variants of the satisfactory partitioning problem is prepared by Bazgan et al. [24]. In addition, finding optimal bramble sets may yield insights in understanding the structure of optimal tree decompositions.

A more well known and active area of research addresses the problem of finding a (simple, undirected) graph's bramble number, given Seymour and Thomas' result that brambles provide a lower bound on the treewidth of graphs [1]. In graph-theory, identifying a graph's treewidth is essential because of the difficulty of problems in this area and the efficiency of algorithms utilizing tree decomposition. For fixed treewidths, tree decomposition has been shown to produce polynomial-time algorithms for a number of NP-complete problems modeled on graphs [25–28]. In addition to its theoretical importance with respect to graph structure, tree decomposition is an important tool in computational and theoretical complexity. Practical algorithms utilizing tree decompositions have also been developed by Lauritzen and Spiegelhalter [29] and Koster et al. [30] (see also Röhrig [31]). The notions of treewidth and tree decomposition were introduced by Robertson and Seymour [32] as part of a series of papers proving Wagner's conjecture (also known as the Graph Minors Theorem). However, the problem of finding the treewidth itself is shown to be NP-