Discrete Applied Mathematics 161 (2013) 2025–2037

Contents lists available at SciVerse ScienceDirect

Discrete Applied Mathematics

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

New bounds and constraint propagation techniques for the clique partitioning problem

Florian Jaehn [a], Erwin Pesch [b,*]

[a] Chair of Sustainable Operations and Logistics, University Augsburg, 86159 Augsburg, Germany

[b] Institute of Information Systems, University Siegen, 57068 Siegen, Germany

ARTICLE INFO

Article history:
Received 2 May 2012
Received in revised form 11 February 2013
Accepted 12 February 2013
Available online 16 March 2013

Keywords:
Clique partitioning
Constraint propagation
Branch and bound

ABSTRACT

This paper considers the problem of clustering the vertices of a complete edge-weighted graph. The objective is to maximize the sum of the edge weights within the clusters (also called cliques). This so-called Clique Partitioning Problem (CPP) is NP-complete, and has several real-life applications such as groupings in flexible manufacturing systems, in biology, in flight gate assignment, etc. Numerous heuristics and exact approaches as well as benchmark tests have been presented in the literature. Most exact methods use branch and bound with branching over edges. We present tighter upper bounds for each search tree node than those known from the literature, improve the constraint propagation techniques for fixing edges in each node, and present a new branching scheme. The theoretical improvements are reflected by computational tests with real-life data. Although a standard solver delivers best results on randomly generated data, the runtime of the proposed algorithm is very low when being applied to instances on object clustering.

1. Introduction

A task that frequently arises in qualitative data analysis is to uncover natural groupings, or types, of objects, each of which is characterized by several attributes. One can think of these objects as vertices of an edge-weighted graph, G; each positive or negative weight represents some measure of similarity or dissimilarity, respectively, of an edge-defining object pair. A clustering of the objects into groups is a partition of the graph, which means a partition of the vertex set of G into non-overlapping subsets. The set of edges connecting vertices of different subsets from some partition of G is called a multicut. In order to find groups that are as homogeneous as possible, positive edges should appear within groups and negative edges in the multicut. Hence, a best clustering is one with a minimal multicut weight.

If there are no restrictions on the number of vertices in each cluster, this problem is called Clique Partitioning Problem (CPP). The CPP is NP-complete unless all edge weights are positive or all weights are negative (see [6,21]). Theoretical aspects of this problem are discussed by Grötschel and Wakabayashi [9], and Grötschel and Wakabayashi [8] present a cutting plane algorithm as well as benchmark tests. Different publications refer to these tests: de Amorim et al. [3] apply simulated annealing, Dorndorf and Pesch [5] present an ejection chain heuristic as well as a branch and bound method, and Brusco and Köhn [1] describe two neighborhood search heuristics with an embedded relocation algorithm and an embedded tabu search method. Kochenberger et al. [14] introduce a new model representation for the CPP, which is especially suitable for large instances, which are then solved using tabu search. The cutting plane algorithm by Grötschel and Wakabayashi [8] is the basis for a cutting plane algorithm by Oosten et al. [17], who present further facet-defining inequalities.

Some papers present models with additional restrictions on the number of cliques or on the number of vertices in a clique. Ji and Mitchell [12] consider the problem in which each cluster must have a minimum number of vertices. They

* Corresponding author. Tel.: +49 271 7402420.
E-mail addresses: florian.jaehn@wiwi.uni-augsburg.de (F. Jaehn), erwin.pesch@uni-siegen.de (E. Pesch).

0166-218X/$ – see front matter © 2013 Elsevier B.V. All rights reserved.
doi:10.1016/j.dam.2013.02.011

present a branch and price scheme for solving the problem. The pricing problem is solved using an integer program. The clustering problem when considering either a maximal or a minimal number of clusters has been studied by Chopra and Rao [2]. The polytopes of both problems are analyzed and valid inequalities and facets are derived. If the number of vertices in each clique is restricted, or, more generally, if the sum of vertex weights in each clique is bounded, the problem is called clustering with knapsack constraint. This problem is analyzed and solved using branch and price algorithms by Johnson et al. [13] and Mehrotra and Trick [16].

Real-life applications of the CPP come from, but are not restricted to, biology, flexible manufacturing systems, airport logistics, and social sciences. In biology, the classification of animals and plants is based on qualitative and/or quantitative descriptions. As there is no limit on the number of classes, the CPP is suited for the determination of a classification. The test set provided by Grötschel and Wakabayashi [8] contains some instances from this application. In manufacturing systems, an approach called Group Technology (GP) (see, for example, the surveys by Potts and Van Wassenhove [18] and Liaee and Emmons [15]) has become popular; it uses similarities of different products (and/or activities) in their production (execution). The groups can be determined using the CPP, as shown by Wang et al. [22] and Oosten et al. [17]. Dorndorf et al. [4] present an application of the CPP to airport logistics. During ground handling, aircraft have to be grouped such that all aircraft of a group are assigned to the same gate. There are quite a few restrictions and objectives, which can be considered by choosing appropriate edge weights, e.g., aircraft being at the airport at the same time should not be in the same group, which is enforced by a large negative value. A variety of applications of the CPP in social sciences, especially in psychology, are given by Brusco and Köhn [1].

In this paper, we describe a new branch and bound algorithm for solving the CPP. First, we present upper bounds that are based on the triangular restrictions, i.e., the fact that, if vertices i and j are in the same cluster, and i and k are in the same cluster, then j and k have to be in the same cluster. In all instances of a test set to be found in literature, we were able to reduce the initial upper bound at the root node by at least 60%. Second, we use constraint propagation techniques. Especially when the difference between upper and lower bound becomes small, these techniques lead to numerous fixations of edges (i.e., the fact that two vertices must or must not be in the same cluster). In many instances tested, the upper bounds and the constraint propagation techniques (although easy to apply) lead to the optimal solution in the root node. Third, we present a new branching scheme using dichotomy. However, in general, the two branches do not halve the search space. Thus, our branching scheme guarantees that the child node containing the bigger part of the search space has a tight upper bound.

2. The clique partitioning problem

Consider a complete edge-weighted graph $G = (V, E, W)$ consisting of a set of vertices $V = \{1, 2, \ldots, n\}$, a set of edges $E = \left\{e_{ij} | e_{ij} = \{i, j\}, i, j \in V, i \neq j\right\}$, and a set of edge weights $W = (w_{ij}),\ i, j \in V,\ w_{ij} \in \mathbb{R} \cup \{-\infty\}$ with $w_{ij} = w_{ji}$. The clique partitioning problem is to find an equivalence relation on V, so that the sum of the edge weights of all vertex pairs in relation is maximized. This is equivalent to finding a partition of V into cliques, i.e., vertex subsets, so that the sum of the edge weights within the cliques is maximized. With binary variables

$$x_{ij} = \begin{cases} 1 & \text{if vertices } i \text{ and } j \text{ are in relation (i.e. } i \text{ and } j \text{ belong to the same clique),} \\ 0 & \text{otherwise} \end{cases}$$

for all edges $\{i, j\}$, the CPP can be described by the following model (see [8]):

$$\begin{aligned} \max \quad & \sum_{1 \le i < j \le n} w_{ij} \cdot x_{ij} \\ \text{s.t.} \quad & x_{ij} + x_{jk} - x_{ik} \le 1 \quad \text{for } 1 \le i < j < k \le n \\ & x_{ij} - x_{jk} + x_{ik} \le 1 \quad \text{for } 1 \le i < j < k \le n \\ & -x_{ij} + x_{jk} + x_{ik} \le 1 \quad \text{for } 1 \le i < j < k \le n \\ & x_{ij} \in \{0, 1\} \quad \text{for } 1 \le i < j \le n. \end{aligned} \tag{2.1}$$

The constraints guarantee the transitivity of the relation: if vertices i and j belong to the same clique and vertices j and k belong to the same clique, then vertices i, j, k belong to the same clique.

We present a branch and bound algorithm for solving this problem. The binary branching procedure decides, for every edge of the graph, whether it is selected and included in a potential solution or not. After each branching, consistency tests are applied (e.g., restrictions are checked) in order to find additional edges that now must or must not be selected. Afterwards, an upper bound is determined for each search tree node.

3. The search tree

First, a lower bound $\underline{g}$ with a corresponding feasible solution for the clique partitioning problem will be determined, using an arbitrary heuristic algorithm. We have used the ejection chain algorithm presented by Dorndorf and Pesch [5].

The search tree structure is as follows. At the root, there is the initial graph $G(V, E, W)$. Then branching with two child nodes and a subsequent constraint propagation follows. In the first node, a specific variable x_{ij} is explicitly set to 1. In other