Journal of Discrete Algorithms 7 (2009) 102–111

Contents lists available at ScienceDirect

Journal of Discrete Algorithms

www.elsevier.com/locate/jda

Cardinality constrained and multicriteria (multi)cut problems

C. Bentz, M.-C. Costa, N. Derhy *, F. Roupin

CEDRIC-CNAM, 292 Rue Saint-Martin, 75141 Paris Cedex 03, France

ARTICLE INFO

Article history:
Received 26 March 2007
Accepted 15 April 2008
Available online 7 November 2008

Keywords:
Multicut
Cardinality constraint
Multicriteria optimization
Dynamic programming
$\mathcal{NP}$-hardness

ABSTRACT

In this paper, we consider multicriteria and cardinality constrained multicut problems. Let G be a graph where each edge is weighted by R positive costs corresponding to R criteria and consider k source–sink pairs of vertices of G and R integers $B_1, \dots, B_R$. The problem R-CriMultiCut consists in finding a set of edges whose removal leaves no path between the ith source and the ith sink for each i, and whose cost, with respect to the jth criterion, is at most B_j, for $1 \leqslant j \leqslant R$. We prove this problem to be $\mathcal{NP}$-complete in paths and cycles even if $R = 2$. When $R = 2$ and the edge costs of the second criterion are all 1, the problem can be seen as a monocriterion multicut problem subject to a cardinality constraint. In this case, we show that the problem is strongly $\mathcal{NP}$-complete if $k = 1$ and that, for arbitrary k, it remains strongly $\mathcal{NP}$-complete in directed stars but can be solved by (polynomial) dynamic programming algorithms in paths and cycles. For $k = 1$, we also prove that R-CriMultiCut is strongly $\mathcal{NP}$-complete in planar bipartite graphs and remains $\mathcal{NP}$-complete in $K_{2,d}$ even for $R = 2$.

1. Introduction

In [2], Bruglieri et al. study a generalization of the well-known minimum cut problem where an additional cardinality constraint is considered. They show that the problems of finding a minimum cut of cardinality either equal to or greater than a given value p are both strongly $\mathcal{NP}$-hard. However, they ask whether the problem MinCutCard, where we look for a minimum cut separating the source s and the sink t, and whose cardinality is at most p, can be solved in polynomial time.

In fact, the decision version of this problem can be seen as a particular case of a multicriteria simple cut problem. In the problem R-CriCut, we are given two vertices s and t, R edge-weight positive functions $w_1, \dots, w_R$, R bounds $B_1, \dots, B_R$ and we look for a cut C which separates s and t such that $w_i(C) \leqslant B_i$ $\forall i \in \{1, \dots, R\}$. For $R = 2$, if we set $w_2(e) = 1$ $\forall e \in E$ and $B_2 = p$, we obtain the decision version of MinCutCard. 2-CriCut has been shown strongly $\mathcal{NP}$-complete for general graphs in [10]. Besides, when we look for a global cut of the graph, i.e. a partition of the vertices into two connected components, the problem is polynomial when the number of criteria is bounded [1].

Let MinMultiCutCard and R-CriMultiCut be generalizations of MinCutCard and R-CriCut respectively, defined as the cardinality constrained and the multicriteria versions of the multicut problem. Given a (directed or not) graph $G = (V, E)$ and a set $T = \{(s_1, t_1), \dots, (s_k, t_k)\}$ of k distinct source–sink pairs of terminal vertices, a multicut C is a subset of E whose removal leaves no (directed) path between s_i and t_i for each $i \in \{1, \dots, k\}$. Then, MinMultiCutCard and R-CriMultiCut can be defined from MinCutCard and R-CriCut respectively, by replacing "cut" by "multicut". For fixed $k > 2$, the minimal multicut problem MinMultiCut (i.e. the optimization version of 1-CriMultiCut) is APX-hard both in undirected and in

* Corresponding author. Tel.: +33 (0) 1 58 80 85 50; fax: +33 (0) 1 40 27 22 96.
E-mail addresses: cedric.bentz@lri.fr (C. Bentz), costa@cnam.fr (M.-C. Costa), nicolas.derhy@cnam.fr (N. Derhy), roupin@cnam.fr (F. Roupin).

1570-8667/$ – see front matter
doi:10.1016/j.jda.2008.04.004

directed graphs [5]. For arbitrary values of k, it is APX-hard in undirected stars (i.e. trees of height 1) [8] but becomes polynomial in directed trees [4].

Obviously, the difficult cases of MinMultiCut are difficult for MinMultiCutCard and R-CriMultiCut. The question is then: do the polynomial cases of MinMultiCut remain polynomial when we add a cardinality constraint or when we consider the multicriteria version?

We study these problems and provide some answers in this paper, which is divided into three sections.

The first one deals with simple cut problems. We show that MinCutCard is strongly $\mathcal{NP}$-hard thus settling one of the open problems of Bruglieri et al. in [2]. Then, we prove that, in planar bipartite graphs, 2-CriCut is $\mathcal{NP}$-complete and R-CriCut is strongly $\mathcal{NP}$-complete.

In Section 2, we show that MinMultiCutCard is strongly $\mathcal{NP}$-hard in directed stars but remains polynomial in paths (and directed paths) and cycles (and circuits).

In Section 3, we study R-CriMultiCut. We show that, in paths, this problem is strongly $\mathcal{NP}$-complete and remains $\mathcal{NP}$-complete for $R = 2$.

2. Simple cut problems

As already mentioned, Bruglieri et al. study in [2] the problem of finding a minimal cut subject to a cardinality constraint. However, MinCutCard (i.e. the case where we have an upper bound on the cardinality of the cut) was an open problem. We show the following theorem:

Theorem 1. MinCutCard *is strongly* $\mathcal{NP}$*-hard.*

Proof. We use a reduction from Bisection [7]. Let $G = (V, E)$ be an undirected graph with $2n$ vertices and m edges, and let B be a given value. The problem is to decide if there exists a partition of V into two disjoint sets V_1 and V_2 such that $|V_1| = |V_2| = n$ and such that the number of edges with one endpoint in V_1 and one endpoint in V_2 is less than or equal to B. Let I_{bi} be an instance of Bisection. We assume that $B < m$, otherwise I_{bi} would obviously have a solution.

We construct an instance I_{cut} of the decision version of MinCutCard as follows (see Fig. 1): first, let us assign weight 1 to the edges of G. Then, we add a vertex t and $2n$ edges of weight $nm + n^2 + m$ connecting t to each vertex of G. For each vertex v_i of G, we add a path q_i of $m + n$ vertices and we add $m + n$ edges connecting each vertex of q_i to v_i. The edges of q_i and the edges connecting the vertices of q_i to v_i have a weight equal to $(nm + n^2 + m)n + (m + n)n + m$. Finally we add a vertex s and $2(m + n)n$ edges of weight 1 connecting s to all the vertices of the paths q_i ($i \in \{1, \ldots, 2n\}$).

We claim that there exists a solution for I_{bi} if and only if there exists a cut separating s from t such that $w(C) \leqslant (nm + n^2 + m)n + (m + n)n + B$ and $|C| \leqslant n + (m + n)n + B$.

If we have a solution of I_{bi}, we construct a solution for I_{cut} in the following way. For each vertex v_i of V_1, we cut the edge connecting v_i to t. For each vertex v_i of V_2, we cut the edges connecting the vertices of q_i to s. Moreover, we cut the edges of G with one endpoint in V_1 and one endpoint in V_2. So, the cut separates $\{s\} \cup V_1 \cup \{v \in q_i \mid v_i \in V_1\}$ from $\{t\} \cup V_2 \cup \{v \in q_i \mid v_i \in V_2\}$. We have $|C| \leqslant |V_1| + (m + n)|V_2| + B = n + (m + n)n + B$ and $w(C) \leqslant (nm + n^2 + m)|V_1| + (m + n)|V_2| + B = (nm + n^2 + m)n + (m + n)n + B$.

Conversely, if we have a solution C of I_{cut}, we construct a solution of I_{bi} in the following way: V_1 is composed by the vertices of G connected to s and V_2 by the vertices of G connected to t. Note that no edge of weight $(nm + n^2 + m)n + (m + n)n + m$ can be in C since $B < m$ and $w(C) \leqslant (nm + n^2 + m)n + (m + n)n + B$.

Let us begin by showing that $|V_1| = |V_2| = n$.

$|V_2| \geqslant n$, because otherwise, we have to cut at least $n + 1$ edges connecting vertices of G to t, so: $w(C) \geqslant (n + 1)(nm + n^2 + m) = (nm + n^2 + m)n + (m + n)n + m > (nm + n^2 + m)n + (m + n)n + B$, which is not possible.

$|V_1| \geqslant n$ otherwise, we would have to cut at least $(n + 1)(m + n)$ edges connecting vertices of q_i to s ($i/v_i \in V_2$) so: $|C| \geqslant (n + 1)(m + n) = (m + n)n + m + n > (m + n)n + B + n$, and the cardinality constraint would be violated.

Thus, since $|V_1| + |V_2| = 2n$, we necessarily have $|V_1| = |V_2| = n$.

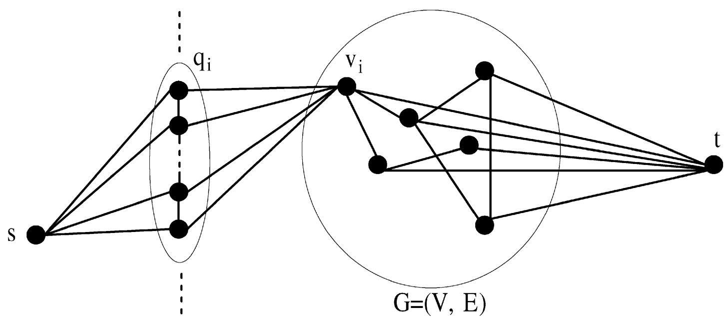

Fig. 1. The graph obtained for MinCutCard ($|q_i| = m + n$).