Available online at www.sciencedirect.com
SCIENCE DIRECT®
European Journal of Operational Research 162 (2005) 55–69
EUROPEAN JOURNAL OF OPERATIONAL RESEARCH
www.elsevier.com/locate/dsw

Minimal multicut and maximal integer multiflow: A survey

Marie-Christine Costa [a,*], Lucas Létocart [a], Frédéric Roupin [b]

[a] CEDRIC, CNAM, 292 rue St-Martin, 75141 Paris cedex 03, France
[b] CEDRIC, CNAM-IIE, 18 allée Jean Rostand, 91025 Evry cedex, France

Received 1 November 2001; accepted 14 October 2003
Available online 25 January 2004

Abstract

We present a survey about the maximum integral multiflow and minimum multicut problems and their subproblems, such as the multiterminal cut and the unsplittable flow problems. We consider neither continuous multiflow nor minimum cost multiflow. Most of the results are very recent and some are new. We recall the dual relationship between both problems, give complexity results and algorithms, firstly in unrestricted graphs and secondly in several special graphs: trees, bipartite or planar graphs. A table summarizes the most important results.

Keywords: Integer multicommodity flows; Minimum multicut; Multiway cut; Complexity; Approximation

1. Introduction

This paper deals, on the one hand, with the minimization of multicuts and some special cases as the well-known multiterminal (or multiway) cut, and on the other hand, with the maximization of integral multiflows and some special cases as unsplittable flows. Consider a n-vertex, m-edge connected graph $G = (V, E)$ with a positive value u_e on each edge e of E and a list of K pairs of terminal vertices $\{s_k, t_k\}$, $k \in \{1, \dots, K\}$. Then, consider the values u_e as capacities and associate a commodity with each terminal pair $\{s_k, t_k\}$. The integer multicommodity flow problem, IMFP, consists in maximizing the sum of the integral flows F_k of each commodity (from s_k to t_k) subject to capacity and flow conservation requirements. Now, consider u_e as the weight of the edge e, the $s_k - t_k$ multicut problem, IMCP, is to find a minimum weight set of edges whose removal separates each pair $\{s_k, t_k\}$ of the list. Such problems have got many applications as in telecommunication, routing and railroad transportation. See, for example, the telephone call congestion problem [6], and the parallel query optimization in databases [31].

For $K = 1$ the problems are the ordinary max flow-min cut problems solvable in polynomial time but both integer multiflow and multicut problems are known to be NP-hard and Max SNP-hard for $K \geqslant 3$ [18,27]. In spite of the difficulty of the problems, several parameters, as the type of the considered graph or the number of terminals, can make them easier. We distinguish directed graphs

* Corresponding author. Tel.: +33-140-272524; fax: +33-140-272296.
E-mail addresses: costa@cnam.fr (M.-C. Costa), letocart@cnam.fr (L. Létocart), roupin@iie.cnam.fr (F. Roupin).

0377-2217/$ - see front matter © 2003 Elsevier B.V. All rights reserved.
doi:10.1016/j.ejor.2003.10.037

from undirected graphs: it is not possible to transform an undirected problem in a directed one by replacing each edge by two opposite arcs, because the value of an edge would bundle both associated arcs. Note that, in a directed graph, if there is no directed path linking two terminals, they are considered as separated and no flow can be routed from one to the other.

We present numerous recent results: complexity results concerning the approximability and the NP-hardness of the basic problems in unrestricted graphs, and polynomial results for the basic problems in directed trees and for the multiterminal problems in trees. We also quote the few proposed methods to solve exactly the basic problems. In addition, we propose some new results. We show that the multiterminal cut problem is polynomial in acyclic graphs, and that IMFP is NP-hard in bipartite graphs. We also prove that IMFP with demands, when one seeks to maximize the number of satisfied demands, is as difficult to approximate as the maximum independent set problem. Finally, we propose a test to detect some polynomially solvable instances of IMFP and IMCP in undirected trees.

The paper is organized as follows. In Section 2, we present the problems, we recall the dual relationship of their continuous relaxations by using very simple mathematical models and we classify the subproblems. We study the case of unrestricted graphs in Section 3, before considering several special graphs as trees or planar graphs, in Section 4. In each one of these two sections, firstly we deal with the basic minimum multicut and maximum integral multiflow problems, denoted by IMCP and IMFP, and secondly we deal with the subproblems. Finally we conclude with a table which summarizes the most important results.

The reader can find very complete surveys concerning the continuous multicommodity flow problem in [1,29] so we do not go back over that. Moreover, we do not deal with the minimum cost integral multiflow problem for which the reader is refered, for example, to [3,13,16], or [46]. We neither get on to the concurrent flow or sparsest cut problems which have also been widely studied [42,61]. Consider a multiflow in which one wish to send d_k units from the source vertex s_k to its sink vertex t_k, $k \in \{1, \dots, K\}$. The objective of the concurrent flow problem is to satisfy all demands by the same maximum proportional amount. The sparsest cut problem is to find a cut that minimizes the ratio capacity of the cut to the demand across the cut. However, if one is only interested to know if all demands can be satisfied or to maximize the flow bounded by these demands, then the problem appears to be a basic IMFP. Simply, add to the graph K vertices v'_k, and K edges (or arcs in digraphs) (t_k, v'_k) with capacities equal to d_k, $k \in \{1, \dots, K\}$, and then move the sink t_k from its initial vertex to the free endpoint of the new edge: all the demands are satisfied if and only if all the added edges are saturated by a maximum multiflow. Conversely, IMFP can be reduced to a maximum multiflow bounded by demands: just consider on each sink t_k, $k \in \{1, \dots, K\}$, a demand d_k equal to the sum of the capacities of the edges (or of the ingoing arcs in a digraph) adjacent to t_k.

2. The multicut and multiflow problems

In this section, we first give formulations of both problems by integer linear programs which allow to underscore their dual relationship. The duality helps to get many further results. Second, we pass in review the special cases.

There is no loss of generality to consider positive values on the edges. Clearly, this restriction is imposed by the definition of a capacity in IMFP. For IMCP, all the edges with negative weights belong to any minimum multicut, and an edge with a value equal to zero do not increase the cut value. Therefore, edges with nonpositive values can be added to the cut and removed from the graph. Moreover, IMFP and IMCP with nonintegral, but rational, values on the edges can be reduced to equivalent problems with integral values. For IMFP, replace the nonintegral values by their lower integer part. For IMCP, multiply the rational values by the product of their lowest common denominator. But, note that the transformation given for one problem cannot be applied to the other. All the results given in this paper are valid for integral values on the edges: in the following, we assume that $u_e \in \mathbb{N}^*$ for all $e \in E$.

2.1. Two continuous dual problems

Garg et al. [26] showed the duality of the continuous relaxed multicut and multiflow problems, by using an arc flow model of the multiflow problem in general graphs [1]. The variables represent the values of each flow on each edge, and the model involves two kinds of constraints: a capacity constraint on each edge and a flow conservation constraint at each vertex. However they use a path flow model when they study the problem in trees [27]. As in [61], we propose to use a path flow model that allows a simpler view of the duality. Let $G=(V,E)$ be some capacited graph and K pairs of terminal vertices $\{s_k,t_k\}$, $k\in\{1,\ldots,K\}$. For each edge e of E, denote by u_e the capacity of the edge e (u_e is assumed to be positive and integral). Let P^k be the set of all the elementary paths from s_k to t_k, $\Pi=\bigcup_{k\in\{1,\ldots,K\}}P^k$ and let M be the cardinality of Π.

Denoting by f_i the flow on the ith path p_i, $i\in\{1,\ldots,M\}$, the integer maximum multiflow problem, IMFP, is to maximize the sum of the f_i while satisfying the capacity constraints. It can be formulated as

$$(\text{IMFP})\left|\begin{array}{lll}\max & \sum_{i=1}^{M} f_i, & \\ \text{s.t.} & \sum_{i\,\text{s.t.}\,e\in p_i} f_i\leqslant u_e \quad \forall e\in E, & (1)\\ & f_i\in\mathbb{N}\quad \forall i\in\{1,\ldots,M\}. & \end{array}\right.$$

The multicut problem IMCP is to find a minimum weight set of edges whose removal separates each pair $\{s_k,t_k\}$, that implies to select at least one edge on each path p_i. It can be formulated as

$$(\text{IMCP})\left|\begin{array}{lll}\min & \sum_{e\in E} u_e c_e, & \\ \text{s.t.} & \sum_{e\in p_i} c_e\geqslant 1 \quad \forall i\in\{1,\ldots,M\}, & (2)\\ & c_e\in\{0,1\}\qquad \forall e\in E, & \end{array}\right.$$

where $c_e=1$ if and only if the edge e belongs to the cut.

Note that the number of variables implemented in (IMFP) is potentially exponential, whereas the arc-flow model involves only mK variables. Nevertheless, the path flow model is the one which is the most often used to obtain the results presented in this paper.

Proposition 1. (Garg et al. [26]) *The continuous relaxation of the minimum multicut program is the linear dual of a continuous maximum multicommodity flow program.*

Proof. (*sketch*) Consider the continuous relaxation of (IMFP) and associate a dual variable c_e to each constraint (1); the obtained dual program is the continuous relaxation of (IMCP); just note that the constraints ($c_e\leqslant 1\forall e\in E$) can be omitted in the continuous program. □

Let f^* and c^* be optimal solutions of (IMFP) and (IMCP). The complementary slackness conditions of optimality in linear programming are given by:

$$\forall i\in\{1,\ldots,M\}\quad f_i^*>0\Rightarrow\sum_{e\in p_i}c_e^*=1, \tag{3}$$

$$\forall e\in E\quad c_e^*>0\Rightarrow\sum_{i\,\text{s.t.}\,e\in p_i}f_i^*=u_e. \tag{4}$$

The conditions (3) mean that, in any optimal solution, either the flow on a path p_i is equal to 0 or the associated constraint (2) is saturated. If the variables c_e^* are integer then there is exactly one edge of p_i in the multicut for all i such that $f_i^*>0$. The conditions (4) mean that, in any optimal solution, if the edge e is not saturated by the flow then $c_e^*=0$. If the variables are integer then all the edges in the cut are saturated edges, i.e. edges with residual capacities equal to zero. These conditions are useful to study several special cases (see Section 4.2).

Note that the duality results presented above are valid in directed or undirected graphs. In the case of a single commodity ($K=1$) the vertices of the primal and dual polyhedrons are integral and the max flow-min cut theorem is a direct consequence of this integrality. But this property is not true in general and that explains the difficulty of the problems [26]. The general cut condition originally given in [47] is also a direct consequence of the Proposition 1. It must be verified by any multiflow F:

(Cut condition) for all $X\subseteq V$,

$$\sum_{e\in\rho_1(X)}u_e\geqslant\sum_{k\in\rho_2(X)}F_k,$$

where $\rho_1(X)$ is the set of edges with exactly one endpoint in X and $\rho_2(X)$ is the set of those $k \in \{1, \dots, K\}$ for which exactly one of s_k and t_k belongs to X. For a given instance, the value of any feasible multiflow is at most the value of any multicut.

2.2. The subproblems

2.2.1. The multiterminal cut and flow problems

One well known subproblem is the MULTITERMINAL (or MULTIWAY) CUT problem where the cut must separate each pair of vertices belonging to a given set of terminals, $X \subseteq V$. Denote by $|X|$ the number of terminals. The MULTITERMINAL CUT problem is the special case of IMCP where the K pairs to separate are the $\frac{1}{2}|X|(|X|-1)$ pairs of terminals. We also consider the MULTITERMINAL FLOW problem which is the maximum integral multiflow associated by duality with the MULTITERMINAL CUT problem: the objective is to maximize the total amount of flow routed between any pair of terminals in X. Both problems can be stated as (IMFP) and (IMCP).

2.2.2. The K-cut problem

A more particular case of the multicut problem is the K-CUT problem. Recall that in this problem, one seeks to partition the n vertices of a graph into K nonempty sets. For a fixed K, it is a particular case of the MULTITERMINAL CUT problem: one can consider all subsets of K vertices, then solve all the corresponding MULTITERMINAL CUT instances, and finally keep the best cut found. By this way, we obtain an optimal solution of K-CUT since at least K vertices have to be in different sets.

2.2.3. The unsplittable flow problem

Generally the total flow of the commodity k, F_k, is split up between several paths of P^k, i.e. several paths linking s_k to t_k in the graph. Sometimes it occurs that each flow F_k is unsplittable: F_k must be routed on only one path of P^k. The UNSPLITFLOW problem is to select K paths, one in each set P^k, to route K flows verifying the capacity constraints, so as to maximize their sum. To get the associated program, we can add the following constraints to (IMFP):

$$f_i f_j = 0, \quad \forall p_i \neq p_j \in P^k, \quad \forall k \in \{1, \dots, K\}. \tag{5}$$

If the graph is a tree, there is at most one path between two vertices, the constraints (5) therefore no longer apply and the MAX UNSPLITFLOW problem is equivalent to IMFP.

2.2.4. The maximum multipath problem

The MAX CAPPATH problem is to maximize the total number of paths linking two paired terminal vertices, such as the number of paths an edge belongs to is not greater than the capacity of this edge. To get the associated program we have just to replace the constraints $f_i \in \mathbb{N}$ of (IMFP) by the following constraints:

$$f_i \in \{0, 1\} \quad \forall i \in \{1, \dots, M\}. \tag{6}$$

2.2.5. The maximum edge disjoint paths problem

Let us consider IMFP with all the capacities on the edges equal to 1: $u_e = 1$, for all $e \in E$. We get a MAX EDGEDISJPATH problem which is to maximize the total number of paths linking K paired terminal vertices such as an edge belongs to at most one path. Several paths are allowed between one terminal pair. All the flows f_i are then equal to 0 or 1 and the problem is also a special case of MAX CAPPATH.

3. Multiflow and multicut in unrestricted graphs

In this section, we give the main complexity results for the integer multiflow and multicut problems and their subproblems in unrestricted graphs. We also present some exact and approximation algorithms.

3.1. Solving IMFP

Contrary to the minimum cost integer multiflow problem for which several practical results have been published [3,16,46] there are few attempts solving IMFP and IMCP. Nevertheless Brunetta et al. proposed in [6] a branch-and-cut algorithm based on a polyhedral approach. They describe several classes of inequalities, and lifting procedures. In particular, they present a new class of valid constraints: the *multihandle comb inequalities*.

They prove that some of these inequalities define facets.

They solve instances of IMFP with unit capacities on the edges, i.e. MAX EDGEDISJPATH, up to 100 vertices, 495 edges and 5 commodities. They also apply their algorithm to a real-world problem having 8 vertices and 28 edges, integer capacity on each edge, and up to 13 commodities. In fact, it seems that one can only hope to solve exactly small instances in unrestricted graphs. The main reason can be found in the study of the complexity of these problems.

3.2. Complexity of IMCP and IMFP

First, recall that both problems are Max SNP-hard even in several particular cases [18,27,52]. This result implies that no polynomial time approximation scheme can exist for these problems unless P = NP. Karp [35] proved that IMFP is strongly NP-hard in directed and undirected graphs. The simple cases of IMFP where $K = 2$ remains NP-hard in undirected graphs [21], and in directed graphs even when all edge capacities are set to 1 [22] (one seeks to solve an instance of the MAX EDGEDISJPATH problem with $K = 2$).

Nevertheless, Hu [32] proved that IMFP in an undirected graph with $K = 2$ is polynomial if the capacities are even. This result was extended by Rajagopalan [54] to the case where the sum of the capacities of all the edges incident on each vertex is even. The proposed algorithm calculates a biflow by combination of two simple flows. These results can be achieved because the continuous solutions of (IMCP) and (IMFP) when $K = 2$ are semi-integral, i.e. variables are multiple of $\frac{1}{2}$ [24,33,60].

In fact, as we are going to see in the next section, IMFP is not only strongly NP-hard but finding an approximate solution within a fixed performance ratio for it is still an NP-hard problem. For IMCP, complexity results are also negative although some particular subproblems can be well approximated (see Section 3.4 for details).

3.3. Approximation of IMFP and IMCP

Since IMFP and IMCP are NP-hard, one can only hope to obtain polynomial-time approximation algorithms to solve them. On the positive side, in undirected graphs, Garg et al. proposed in [26] an O(log(K))-approximate algorithm where K is the number of commodities. Their algorithm provides solutions to both IMFP and IMCP. To achieve this remarkable result, the authors use a linear programming relaxation based on the arc-flow model of IMFP. By solving the dual of this linear program, they define a new graph with distance labels on the edges. Then, using the complementary slackness conditions and starting from each terminal (source or sink) they build several cuts separating the initial vertex from its mate. Finally, they obtain a feasible multicut considering the union of all these cuts. Moreover, it is proved in the same paper that the analysis of the worst case is tight (an example achieving the bound is given).

In the directed case, both problems seem more difficult. For IMCP, Cheriyan et al. [9] proposed a polynomial-time algorithm which finds a multicut whose value C satisfies $C \leqslant 108F^{*3}$, where F^* is the value of a maximum multiflow. They also proved that one can find in polynomial-time a multicut whose value C satisfies $C \leqslant 39\ln(K+1)F^{*2}$. This result must be compared with the one obtained in undirected graphs [26]: $C = O(F^* \log(F^*))$.

The best negative result about the approximability of IMFP in directed graphs says that, unless P = NP, no polynomial time algorithm can provide a better performance ratio than $m^{\frac{1}{2}-\varepsilon}$ for any $\varepsilon > 0$ [30]. In the same paper, authors propose a greedy $O(\sqrt{md_{\max}}\log^2 m)$-approximate algorithm for the integer maxflow problem where one seeks to maximize the number of satisfied demands (here $d_{\max}$ is the maximal demand value). Now, we give another negative result about the approximability of this particular problem. We built the proof from an idea suggested in [30].

Proposition 2. *Unless P = NP, there is no polynomial-time approximation algorithm with a fixed performance guarantee for the integer maximum multiflow problem with the aim of satisfying a maximal number of demands in an undirected graph.*

Proof. We use a polynomial reduction from the MAXIMUM INDEPENDENT SET problem that preserves the strong negative approximation results known for this problem. Consider an instance $G = (V, E)$ of the MAXIMUM INDEPENDENT SET problem, with $V = \{v_1, \dots, v_n\}$. We build $G' = (V', E')$ in the following way: we add n new vertices v'_i ($i \in \{1, \dots, n\}$) to V and we add an edge between v_i and v'_j if and only if $(v_i, v_j) \in E$. Now, we consider n commodities and assign a source s_i to each vertex v'_i and a sink t_i to each vertex v_i ($i \in \{1, \dots, n\}$). At each sink t_i we set a commodity demand equal to the degree of v_i. Finally, we set the capacity of each edge of G' to one.

We claim that the maximal number of demands which can be satisfied is equal to the cardinality of a maximum independent set in G, i.e. to $\alpha(G)$: let v_i be a vertex of G such that the demand at t_i is satisfied; all edges incident to v_i are saturated by the flow F_i and thus, the demands at t_j such that $(v_i, v_j) \in E$ cannot be satisfied. Finally, the set of the vertices whose demands are totally satisfied is an independent set of G. Hence, the maximum number of satisfied demands is equal to $\alpha(G)$ (Fig. 1). $\square$

3.4. Complexity and approximation of the subproblems

Here, we give several complexity results about particular cases of IMFP and IMCP in unrestricted graphs.

3.4.1. Multiterminal cut and flow problems

As said in Section 2.2.1, the MULTITERMINAL CUT problem is a particular case of IMCP. In this section, K denotes $|X|$. Dalhaus et al. proved in

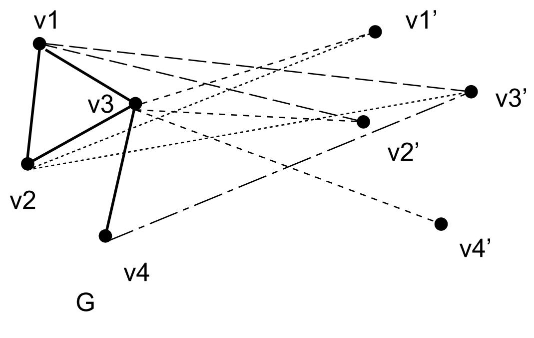


Fig. 1. IMFP with demands and INDEPENDENT SET problems.

[18] that the MULTITERMINAL CUT problem in undirected graphs is NP-hard for $K \geqslant 3$ even in planar graphs, and is Max SNP-hard for $K \geqslant 3$ in unrestricted graphs. This last result has been used in [25] to prove that the MULTITERMINAL CUT problem is MAX SNP-hard in directed graphs even for $K = 2$. This also implies that no polynomial approximation scheme can exists unless P = NP [2]. Nevertheless, in [18], there is a positive result about the approximation of this problem in undirected graphs: there exists a polynomial time $(2 - \frac{2}{K})$-approximation algorithm for the MULTITERMINAL CUT problem in unrestricted graphs. The main idea of this algorithm is to build a feasible solution by doing K "isolating" cuts for each terminal vertex. This last result has been improved: first, Calinescu et al. [7] used a new geometric relaxation and obtained a $(\frac{3}{2} - \frac{1}{K})$-approximation algorithm. Their relaxation uses the K-simplex S_K which has K vertices; the ith vertex is the point x in S_K with $x_i = 1$ (and all other coordinates equal 0). The relaxation is as follows: map the vertices of the graph to points in S_K such that terminal i is mapped to the ith vertex of S_K. Each edge is mapped to the straight line between its endpoints. Then, in [34], Karger et al. improved the ratio of this approximate algorithm by studying the previous geometric relaxation and have obtained a 1.3438-approximate algorithm for any K, and a $\frac{12}{11}$-approximate algorithm for $K = 3$. Polyhedral results are given in [5,10,17].

Recall that a more particular case of IMCP is the K-CUT problem (see Section 2.2.2 for definition). This problem is polynomial for fixed K but NP-hard for a nonfixed K [28]: one can actually prove that the CLIQUE problem polynomially reduces to it. Levine [45] has recently improved Goldschmidt and Hochbaum's results [28] by proposing a polynomial time algorithm running in $O(mn^{K-2} log^3 n)$ when $K \leqslant 6$. Note that all these results only hold in undirected graphs.

In directed graphs, all the multicut problems seem harder to approximate. Nevertheless, the MULTITERMINAL CUT problem can be approximated within a ratio equal to $2 \log K$ [25]. The algorithm contains $\log K$ phases, and at each step it removes edges having capacity at most twice the value of the maximum multiflow on the considered edge. Moreover, Naor and Zosin have improved the O(log(K))-approximate algorithm of Garg et al. [26] by proposing a 2-approximation algorithm for the symmetric multicut problem [50]. This approximate algorithm uses a particular linear program where the integrality gap is at most 2. Recall that a symmetric multicut means a set of arcs whose removal disconnects either s_k from t_k or t_k from s_k, for every symmetric pair of commodities. Unfortunately, there is no relation between this problem and IMCP in directed graphs.

To our knowledge, there are no results about the MULTITERMINAL FLOW problem in unrestricted graphs. We can just make the following remark: expect an optimum solution f^* containing a subflow f, routed from a terminal t_1 to a terminal t_3 along a path containing a terminal t_2. Contradicting the optimality of f^*, we could improve the value of f^* by f. Simply, replace f by two independent subflows f_1 and f_2 equal to f, one from t_1 to t_2 and the other from t_2 to t_3. Therefore, no optimal solution admits a subflow "crossing" a terminal vertex.

3.4.2. Edge disjoint paths and unsplittable flows problems

On the positive side, some special cases of IMFP and IMCP are less difficult problems: the MAX UNSPLITFLOW and the MAX EDGEDISJPATH problems admits an $O(\sqrt{m})$-approximate algorithm (see [4,36,39]). This is the best guarantee that can be achieved by a polynomial algorithm for these two last problems in directed graphs, since these problems are NP-Hard to approximate in directed graphs with a factor $m^{\frac{1}{2}-\varepsilon}$ for any $\varepsilon > 0$ [30]. Furthermore, the MAX EDGEDISJPATH problem is NP-hard for $K = 2$ in directed graphs [22]. This is in contrast with the undirected case, where Robertson and Seymour [55] showed that, for any fixed K, the MAX EDGEDISJPATH problem is solvable in polynomial time. As the authors remark their algorithm is out of the range of practical usability when $K \geqslant 3$. For $K = 2$, if the degree of each nonterminal vertex is even, the optimum values of the MAX EDGEDISJPATH problem and of the associated multicut are equal [56].

For directed and undirected graphs, the MAX EDGEDISJPATH problem is NP-hard if we do not fix K ([22,49]). Furthermore, many results about disjoint paths problems were presented by Frank [23] and Schrijver [58].

4. Special graphs

This section deals with the study of the different problems in special graphs as trees, bipartite graphs, planar graphs and rings. Before presenting the problems in trees, we show how to solve the MULTITERMINAL CUT and FLOW problems in directed acyclic graphs.

4.1. Acyclic graphs

We have seen that the MULTITERMINAL CUT problem is NP-hard in directed graphs, nevertheless if the graph does not admit directed cycle, we prove the following result:

Proposition 3. *The multiterminal cut and integer flow problems are solvable in polynomial time in an acyclic directed graph by using a simple flow algorithm.*

Proof. Let $G = (V, E)$ be an acyclic digraph. Denote by $d^+(v)$ (resp. $d^-(v)$) the outgoing (resp. ingoing) degree of $v \in V$ and recall that, in any optimal solution, no flow is routed "through" a terminal vertex (see the end of Section 3.4.1). Without making any assumption on the maximum multiterminal flow, we can split up each terminal vertex t_k such that $d^+(t_k) \neq 0$ and $d^-(t_k) \neq 0$ into two terminal vertices, t'_k and t''_k. t'_k (resp t''_k) is the final (resp. initial) endpoint of each arc having t_k as final (resp. initial) endpoint (see Fig. 2). Let us add, to the graph so obtained, an ingoing vertex v_0 and an outgoing vertex v_{n+1}, and add an arc from v_0 to each terminal vertex t_k such that $d^-(t_k) = 0$, and an arc from each terminal t_k such that $d^+(t_k) = 0$ to v_{n+1}. All these arcs are valued with a sufficiently large number Δ. We denote this new graph by $\widehat{G}$. Finding an optimal multiterminal flow in G is equivalent to finding a simple maximum flow from v_0 to v_{n+1} in $\widehat{G}$ since the added arcs do not limit the flow. Moreover, the associated

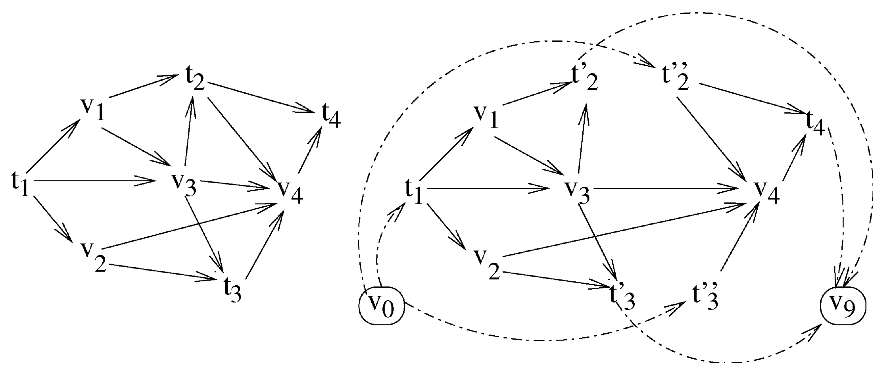


Fig. 2. A graph G and its associated graph $\widehat{G}$.

minimum cut in $\widehat{G}$ is the minimum multiterminal cut in G, because the added arcs have got values Δ and cannot belong to a minimum cut in $\widehat{G}$. □

Note that, if the graph G is not acyclic, a solution in $\widehat{G}$ can contain a flow routed from t''_k to t'_k, that does not define a flow in G.

Furthermore, the decision problem "are there K pairwise disjoint paths?" is polynomial for fixed K in acyclic digraphs [22]. The total number of disjoint paths is bounded by m in any graph. Therefore, the MAX EDGEDISJPATH problem is polynomial for fixed K in acyclic digraphs.

4.2. Trees

In trees, there exists only one path between the source and the sink. This property is useful to solve in polynomial time the problems in directed trees but, unfortunately, it does not imply the polynomiality of the different problems in undirected trees. Note that, when considering the MULTITERMINAL CUT and FLOW problems on trees, we can assume without loss of generality that there is a bijection between the set of leaves and the set of terminals, otherwise the problem is decomposed in independent subproblems by splitting up nonleaf teminal vertices [15].

4.2.1. Directed trees

The results given in Section 4.1 hold here, but stronger results can be obtained in directed trees. Costa et al. [14] proved that IMFP and IMCP are polynomial in directed trees. To obtain this result, they noticed that the constraint matrix of (IMFP) and (IMCP) in a directed tree is totally unimodular [8] (see Fig. 3). As a consequence, IMFP and IMCP can be solved by linear programming.

Note that IMFP in a directed tree can be transformed in a circulation problem [14], which can be polynomially solved [51]. Thus, in addition to the MULTITERMINAL CUT and FLOW problems, the MAX UNSPLITFLOW, the MAX CAPPATH and the MAX EDGEDISJPATH problems are also polynomial in directed trees.

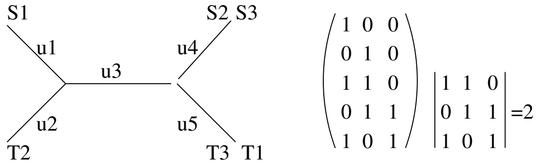
4.2.2. Rooted trees

Costa et al. proposed an $O(\min(Kn, n^2))$ greedy algorithm to solve both multiflow and multicut problems in a rooted tree [14]. This result is achieved by using duality results. First, a multiflow is computed by routing maximum flows from each source, in a well choosing order. Second, a cut verifying the complementary slackness conditions (see Section 2.1) is obtained. Unfortunately, the algorithm cannot be adapted to any directed tree.

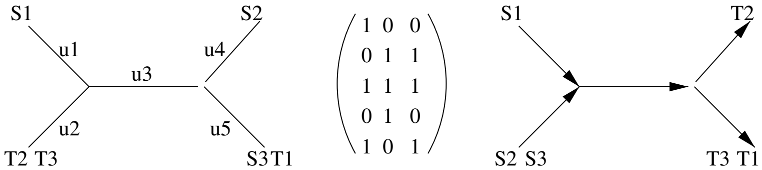
The MULTITERMINAL CUT and FLOW problems, in a rooted tree with L leaves, can be reduced to IMCP and IMFP with L sources located at the root and L sinks, one at each leaf. The MULTITERMINAL CUT problem is then solved in $O(n)$ and the MULTITERMINAL FLOW problem in $O(Lh)$ where h is the height of the tree [15].

4.2.3. Bidirected trees

A bidirected tree is the directed graph obtained from an undirected tree by replacing each edge by two directed opposite and independent arcs. Erle-



A difficult instance: the matrix is not totally unimodular.



A polynomially solvable instance and the associated directed tree: the matrix is totally unimodular.

Fig. 3. Easy and difficult instances in undirected trees.

bach and Jansen [20] proved that the MAX EDGEDISJPATH problem is Max SNP-hard in bidirected trees of any degree. They gave a linear reduction from the bounded variant of the three-DIMENSIONAL MATCHING problem. They also proposed a $(\frac{5}{3}+\varepsilon)$-approximation algorithm for the MAX EDGEDISJPATH problem in bidirected trees. Nevertheless, the MAX EDGEDISJPATH problem can be solved optimally in polynomial time if the input is restricted:

(a) if the maximum degree of the tree is bounded by a constant then the optimal solution can be obtained using dynamic programming.
(b) if the bidirected tree is a star, i.e. it contains only one vertex with outgoing degree greater than one, the MAX EDGEDISJPATH problem can be reduced to a maximum matching problem in a bipartite graph which is polynomially solvable. This latter result also applies to spiders: a spider is a bidirected tree in which at most one vertex (the center) has outgoing degree greater than two.

4.2.4. Undirected trees

Garg et al. [27] shown that both IMCP and IMFP are Max SNP-hard in an undirected tree. They use a linear reduction from the three-DIMENSIONAL MATCHING problem for IMFP, and one from the VERTEX COVER problem for IMCP. Srivastav and Stangier [62] extended the Max SNP-hardness of IMFP to trees with large capacities.

Nevertheless, when trying to solve an instance of IMFP or IMCP in an undirected tree, one can verify if it is possible to orient the edges to get an equivalent directed problem. For all k in $\{1,\dots,K\}$, there must be a directed path either from s_k to t_k or from t_k to s_k in the obtained directed tree. In this case, the instance is polynomially solvable (see Section 4.2.1). This test can be done in $O(m)$. First, orient the edges of p_1 from s_1 towards t_1; then, consider the paths p_k one by one, beginning with the paths having an edge already oriented; while it is compatible with the previous orientations, orient the edges of p_k, either from s_k towards t_k or from t_k towards s_k (in that last case, interchange s_k and t_k). If none of these orientations is compatible the process stops (see Fig. 3).

The authors of [27] also present an efficient algorithm for IMCP and IMFP such that the weight of the multicut is at most twice the value of the flow, i.e. a 2-approximation algorithm for IMCP and an $\frac{1}{2}$-approximation algorithm for IMFP on trees. Their algorithm follows a primal–dual approach and is guided by the complementary slackness conditions (see Section 2.1). They begin by rooting the tree at an arbitrary vertex and the algorithm makes two steps over the tree. In the first step, they move up the tree, routing flows as they go along and picking some saturated edges. In the second step, they move down the tree dropping redundant edges they have picked. They proved that the set of edges obtained at the end of the algorithm is a multicut and that this multicut includes at most two edges of any flowpath. Therefore, the capacity of the multicut is at most twice the value of the multiflow.

In trees of height one (stars), IMCP remains NP-hard even with unit capacities (considering the linear reduction from VERTEX COVER), although IMFP can be solved in polynomial time, because it is equivalent to the maximum B-MATCHING problem on general graphs [27]. Moreover, IMFP in a tree with unit capacities on the edges is a MAX EDGEDISJPATH problem which is polynomially solvable [27]: roughly speaking, the algorithm consists in routing flows on subtrees of height 1, with two passes on the tree, first from the leaves, and second back to the leaves.

Nevertheless, contrary to the basic problems, both MULTITERMINAL CUT and MULTITERMINAL FLOW problems are polynomial if the graph is an undirected tree. Erdos and Szekely [19] proposed an $O(n^2)$ algorithm to solve MULTITERMINAL CUT in trees: in fact, their algorithm solves a more general problem which is to separate r disjoint subsets of vertices. Costa [15] gave algorithms in $O(n)$ for the MULTITERMINAL CUT and in $O(n^2)$ for the MULTITERMINAL FLOW problems and showed that most often it exists a duality gap between the optimal integral multicut and multiflow values. Both algorithms are independent but their general schemes are similar, beginning with stars connected to the tree by an only edge, reducing the tree and reiterating the process.

4.3. Bipartite graphs

Here, we consider augmented bipartite graphs, i.e. bipartite graphs to which we add K sources and K sinks (see Fig. 4): the "supply" graph is bipartite.

Proposition 4. *IMFP in an augmented bipartite digraph is NP-hard if $K \geqslant 3$.*

Proof. The proof uses the discrete tomography problem "recovering polyatomic structure from discrete X-rays" [11], i.e. the reconstruction of colored (a, b)-matrices from the colors projections (see, for example, [53]). Given K colors and the numbers A_{ki} (resp. B_{kj}), $k \in \{1, \dots, K\}$, of each color on each row i, $i \in \{1, \dots, a\}$ (resp. column j, $j \in \{1, \dots, b\}$), is there a coloration of the matrix according to the projections? To each term (i, j) is associated a colored (or empty) "space" (i, j). This problem is known to be NP-hard for $K \geqslant 3$ [11]. We assume that $\sum_{i=1}^{a} A_{ki} = \sum_{j=1}^{b} B_{kj}$, for all $k \in \{1, \dots, K\}$, otherwise there is no solution. We associate to the matrix a complete bipartite digraph $G = (X, Y, E)$ such that $|X| = a$, $|Y| = b$,

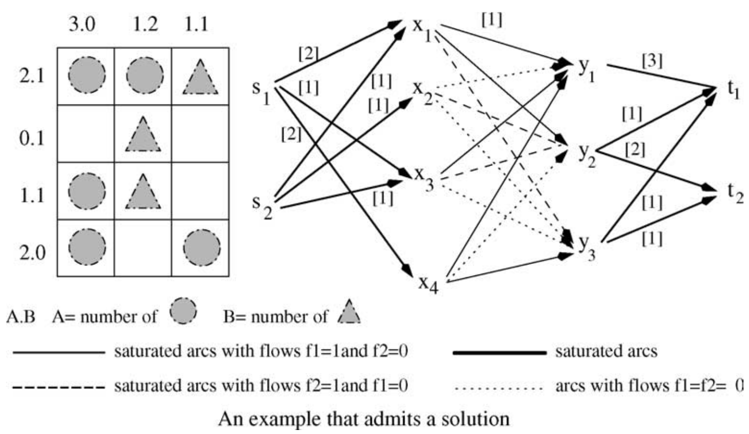


Fig. 4. Matrix reconstruction and bipartite multiflow.

$E = \{(x_i, y_j) \text{ s.t. } x_i \in X \text{ and } y_j \in Y\}$ and the capacity of an arc of E is 1. Now, let us add K sources, s_k, $k \in \{1, \ldots, K\}$, to the graph and Ka arcs, an arc with capacity A_{ki} from each source s_k to each vertex x_i of X, and then let us add K sinks t_k, $k \in \{1, \ldots, K\}$, and Kb arcs, an arc with capacity B_{kj} from each vertex y_j of Y to each sink t_k (see Fig. 4). We get an augmented bipartite capacited graph $\widehat{G}$ such that the matrix admits a coloration if and only if a maximum integer multiflow is equal to $\sum_{i=1}^{a} A_{ki}$, i.e. all the arcs with an endpoint or in a source or in a sink are saturated; the color k is assigned to the matrix space (i, j) if and only if $f_k = 1$ on the arc (x_i, y_j). □

If $K = 2$ the complexity of IMFP in an augmented bipartite digraph is still open. Nevertheless, let us recall that, if $K = 2$, the problem is polynomially solvable in an undirected graph with even capacities (see Section 3.2).

Contrary to IMFP, the MULTITERMINAL FLOW problem (and the MULTITERMINAL CUT problem) is polynomial in a directed bipartite graph, with sources in X and sink in Y, or in a directed augmented bipartite graph defined as in Fig. 4 (see Section 4.1).

4.4. Planar graphs

In a planar graph G the MAX EDGEDISJPATH problem is NP-hard [48,49], therefore IMFP is NP-hard too. Furthermore, authors often consider augmented graphs, obtained by adding to G all the edges $\{s_k, t_k\}$, $k \in \{1, \ldots, K\}$. When the augmented graph is planar, Sebo [59] proved that, for fixed K, IMFP is polynomial, and Korach and Penn [40] gave an $O(n\sqrt{\log n})$ algorithm for $K = 2$. They transform the graph in a dual graph, they calculate shortest paths, and then they solve a set of linear equations and inequalities for finding a maximum integral two flow.

IMCP and the MULTITERMINAL CUT problem in planar graphs have been proved to be NP-hard in [18]: the reduction is made from planar 3-SAT. The ratio of the values of the minimum multicut and the maximum multiflow in planar graphs is at most $O(1)$, and there is a constant factor approximation algorithm for IMCP. These latter results have been obtained by Tardos and Vazirani [63]. They use a decomposition algorithm and then they solve the dual linear program corresponding to the multicommodity flow problem by finding shortest paths.

The MAX EDGEDISJPATH problem is NP-hard because its associated decision problem is NP-complete [48,49]. Note that the necessary cut condition given in the introduction (Section 1) has been revised by Frank [23] when the graph is a rectilinear grid: this revised condition becomes sufficient for the existence of K disjoint paths, if it applies to every row and column of the grid. The gap between the maximum integral flow value and the maximum fractional multiflow value for grid graphs can be as high as $\frac{K}{2}$ [27]. The result given for MAX EDGEDISJPATH implies that MAX CAPPATH and MAX UNSPLITFLOW problems are NP-hard in planar graphs. Kleinberg and Tardos [37,38] gave an $O(\log n)$-approximation algorithm for the MAX EDGEDISJPATH problem in special planar graphs (nearly-Eulerian and uniformly high-diameter), which include planar interconnection networks.

4.5. Rings

A ring is a connected graph where all vertices have degree 2. Thus, all the results given for planar graphs are valid. Such a structure is often used in telecommunication networks [12]. Several simplifications can be made before solving IMCP and IMFP in rings. The main one is that a path without terminals, except for its endpoints, may be reduced to a single edge, which is the lowest weighted edge of the path. Moreover problems in bidirectional rings can be transformed in equivalent problems in directed rings by doubling the number of commodities. In fact, without loss of generality, one can assume that there is a source and/or a sink located at each vertex [43]. In this last paper, a polynomial algorithm in $O(n^3)$ is proposed to solve IMCP in ring networks. The algorithm is based on the enumeration of several minimum cuts associated with an arbitrary path p_{k^*}, each one containing one different edge of the path; these cuts are obtained by using the algorithm given for rooted trees (see Section 4.2.2). There is often a gap between the cut and integral

Table 1
Main results for IMFP, IMCP and their subproblems

	IMFP	IMCP	UnSplitFlow	CapPath	EdgeDisjPath	Multiterminal cut	Multiterminal flow
Undirected graphs	Max SNP-hard [27] NP-hard to approx. within $m^{\frac{1}{2}-\varepsilon}$ [30]	Max SNP-hard [18] O($\log K$)-approx. algo. [26] O($F^* \log(F^*)$)-approx. algo. [26]	NP-hard [49] O($\sqrt{m}$)-approx. algo. [4]	NP-hard [49]O($\sqrt{m}$)-approx. algo. [4]	NP-hard [49] O($\sqrt{m}$)-approx. algo. [4] Polyn. for fixed K [55]	Max SNP-hard [18] 1.3438-approx. algo. [34]	
Directed graphs	Max SNP-hard [27] NP-hard to approx. within $m^{\frac{1}{2}-\varepsilon}$ [30]	Max SNP-hard [18] $C^* \leqslant 108F^{*3}$ and $C^* \leqslant 39ln(K+1)F^{*2}$ [9]	NP-hard [22] O($\sqrt{m}$)-approx. algo. [4] NP-hard to approx. within $m^{\frac{1}{2}-\varepsilon}$ [30]	NP-hard [22] O($\sqrt{m}$)-approx. algo. [4] NP-hard to approx. within $m^{\frac{1}{2}-\varepsilon}$ [30]	NP-hard [22] O($\sqrt{m}$)-approx. algo. [4] NP-hard to approx. within $m^{\frac{1}{2}-\varepsilon}$ [30]	Max SNP-hard [18] $2\log K$-approx. algo. [25] Polyn. if acyclic O(n^3)	Polyn. if acyclic O(n^3)
Directed trees	Polyn. O($\max(K^2 \log n, n^2 \log^2 n)$) [14]	Polyn. [14]	Polyn. [14]	Polyn. [14]	Polyn. [14]	Polyn.	Polyn.
Rooted trees	Polyn. O($\min(Kn, n^2)$) [14]	Polyn. O($\min(Kn, n^2)$) [14]	Polyn. O($\min(Kn, n^2)$) [14]	Polyn. O($\min(Kn, n^2)$) [14]	Polyn. O($\min(Kn, n^2)$) [14]	Polyn. O(n)	Polyn.O(Kh) h = height(T)
Bidirected trees	Max SNP-hard [20]		Max SNP-hard [20]	Max SNP-hard [20]	Max SNP-hard ($5/3+\varepsilon$)-approx. algo. [20]		
Undirected trees	Max SNP-hard 1/2-approx. algo. [27]	Max SNP-hard 2-approx. algo. [27]	Max SNP-hard 1/2-approx. algo. [27]		Polyn. [27]	Polyn. O(n) [15]	Polyn. O(n^2) [15]
Augmented bipartite graphs	NP-hard for $K \geqslant 3$. Open for $K = 2$						
Planar graphs	NP-hard [49]	NP-hard [18] Constant factor approx. algo. [63]	NP-hard [49]	NP-hard [49]	NP-hard [49]	NP-hard [18]	
Rings	Polyn. O(n^3)if demands and uniform capacities [41]	Polyn. O($\min(Kn^2, n^3)$) [43]	NP-hard if demands [12]		Polyn. [23]	Polyn. [43]	Polyn. [43]

flow values in rings. An example is given by a directed ring with 3 vertices v_1, v_2, v_3, 3 edges of value 5 and 3 pairs $\{s_k, t_k\}$, such that $s_1 = t_2 = v_1$, $s_2 = t_3 = v_2$ and $s_3 = t_1 = v_3$: we get $v(C^*) = 10$ and $v(\Phi^*) = 7$.

Now, let us consider the subproblems. THE MULTITERMINAL FLOW and CUT problems are trivially solved in $O(n)$: both optimum values are equal to the sum of the values of the edges remaining in the ring after simplifications. The MAX EDGEDISJPATH problem is polynomial in rings [23].

We want also to present two variants of IMFP in ring networks. Kubat et al. [41] proposed an $O(n^3)$ algorithm for finding an integer multiflow with demands on bidirectional rings with uniform capacities. The second variant is the UNSPLITFLOW problem in ring networks, where each demand must be routed entirely in a clockwise or a counterclockwise direction; this problem was shown to be NP-hard by Cosares and Saniee in [12]. Note that the problems with demands cannot be reduced to the corresponding maximization problems (as proposed at the end of Section 1) without loosing the ring structure.

5. Conclusion

The first concluding remark to be made is the difficulty to solve efficiently max multiflow and min multicut problems except for special cases or small instances. In fact, bounds provided by linear programming are not enough tight to be used successfully in a Branch and Bound algorithm. The semidefinite programming (SDP) may be a good attempt to provide a bound good enough, although its computing cost is high. Actually, Létocart and Roupin proposed in [44] such an approach for the multicut problem in trees using the algorithm proposed in [57]: numerical results showed that SDP improve substantially the bound provided by linear programming; it can be used with LP in order to solve larger instances. This approach could be extended to unrestricted graphs.

It is interesting to note that the complexity and approximability of the multicut and integral multiflow problems, and of their subproblems, is often affected by choosing directed or undirected graphs, but not always in the same way. In some cases, problems are easier to solve in directed graphs: for instance the multicut and multiflow problems are polynomial in directed trees but Max SNP-hard in undirected trees. In other cases they are harder: approximate the multiterminal cut problem in a directed graph seems more difficult than in an undirected one; in the same way, the max edge disjoint path problem is NP-hard for $K = 2$ in directed graphs but is polynomial for any fixed K in undirected graphs.

Beside the difficult general problems, several special but important cases are polynomial. Let us quote the edge disjoint path problem and the multiterminal cut and flow problems when the graph is an undirected tree, the two flow problem in planar graphs, the multiterminal cut and integral flow problems in acyclic graphs and the multicut and integral flow problems in directed trees.

Finally, we would like to point out that there are still some interesting opened questions: what is the complexity of the two flow problem in directed bipartite graphs? Is it possible to approximate within a constant ratio the minimum multicut problem in unrestricted graphs?

To conclude this survey, we give Table 1 which summarizes the most significant results.

References

[1] R. Ahuja, T. Magnanti, J. Orlin, Networks Flows, Theory, Algorithms, Applications, Prentice Hall, Englewood Cliffs, NJ, 1993.

[2] S. Arora, C. Lund, R. Motwani, M. Sudan, M. Szegedy, Proof verification and hardness of approximation problems, in: Proceedings 33rd Annual Symposium On Foundations of Computer Science, IEEE Computer Society, Los Angeles, CA, 1992, pp. 14–23.

[3] C. Barnhart, C. Hane, P. Vance, Using branch-and-price to solve origin–destination integer multicommodity flow problem, Operations Research 32 (3) (1998) 208–220.

[4] A. Baveja, A. Srinisvasan, Approximation algorithms for disjoint paths and related routing and packing problems, Mathematics of Operations Research 25 (2000) 255–280.

[5] D. Bertsimas, C.-P. Teo, R. Vohra, Analysis of LP relaxations for multiway and multicut problems, Networks 34 (1999) 102–114.

[6] L. Brunetta, M. Conforti, M. Fischetti, A polyhedral approach to an integer multicommodity flow problem, Discrete Applied Mathematics 101 (2000) 13–26.
[7] G. Calinescu, H. Karloff, Y. Rabani, An improved approximation algorithm for Multiway Cut, Proceedings of the Thirtieth Annual ACM Symposium on Theory of Computing, Dallas, TX, 23–26 May, 1998, pp. 48–52.
[8] P. Camion, Matrices totalement unimodulaires et problèmes combinatoires, Université de Bruxelles, Thése et Rapport Euratom, 1963.
[9] J. Cheriyan, H. Karloff, Y. Rabani, Approximating directed multicuts, FOCS (2001) 320–328.
[10] S. Chopra, M.R. Rao, On the multiway cut polyhedron, Networks 21 (1991) 51–89.
[11] M. Chrobak, Reconstructing polyatomic structures from discrete X-rays: NP-completeness proof for three atoms, Theoretical Computer Science 259 (2001) 81–98.
[12] S. Cosares, I. Saniee, An optimization problem related to balancing loads on SONET rings, Telecommunication Systems 3 (1994) 165–181.
[13] M.-C. Costa, A. Hertz, M. Mittaz, Bounds and heuristics for the shortest capacitated paths problem, Journal of Heuristics 8 (4) (2002) 449–466.
[14] M.-C. Costa, L. Létocart, F. Roupin, A greedy algorithm for multicut and integral multiflow in rooted trees, Operations Research Letters 31 (1) (2003) 21–27.
[15] M.-C. Costa, Polynomial algorithms to solve the multiway cut and integer flow problems on trees, ECCO XV, Lugano. CEDRIC report (2002).
[16] M.-C. Costa, F.-R. Monclar, M. Zrikem, Variable neighborhood search for the optimization of cable layout problem, Journal of Intelligent Manufacturing 13 (5) (2002) 353–365.
[17] W.H. Cunningham, The optimal multiterminal cut problem, DIMACS Series in Discrete Mathematics and Theoretical Computer Science 5 (1991) 105–120.
[18] E. Dalhaus, D.S. Johnson, C.H. Papadimitriou, P.D. Seymour, M. Yannakakis, The complexity of multiway cuts, SIAM Journal of Computers 23 (1994) 864–894.
[19] P. Erdos, L.A. Szekely, On weighted multiway cuts in trees, Mathematical Programming 65 (1994) 93–105.
[20] T. Erlebach, K. Jansen, The maximum edge-disjoint paths problem in bidirected trees, SIAM Journal on Discrete Mathematics 14 (3) (2001) 326–355.
[21] S. Even, A. Itai, A. Shamir, On the complexity of timetable and multicommodity flow problems, SIAM Journal of Computers 5 (1976) 691–703.
[22] S. Fortune, J. Hopcroft, J. Wyllie, The directed subgraph homeomorphism problem, Theoritical Computer Science 10 (4) (1980) 111–121.
[23] A. Frank, Packing paths, circuits and cuts–a survey, in: B. Korte, L. Lovasz, H.J. Prömel, A. Schrijver (Eds.), Paths, Flows and VLSI-Layout. Algorithms and Combinatorics 9, Springer-Verlag, Berlin, 1990, pp. 47–100.
[24] A. Frank, A.V. Karzanov, A. Sebö, On integer multiflow maximization, SIAM Journal on Discrete Mathematics 10 (1) (1997) 158–170.
[25] N. Garg, V.V. Vazirani, M. Yannakakis, Multiway cuts in directed and node weighted graphs, in: Proceedings ICALP, 1994, pp. 487–498.
[26] N. Garg, V.V. Vazirani, M. Yannakakis, Approximate max-flow min-(multi)cut theorems and their applications, SIAM Journal of Computers 25 (2) (1996) 235–251.
[27] N. Garg, V.V. Vazirani, M. Yannakakis, Primal–dual approximation algorithms for integral flow and multicut in trees, Algorithmica 18 (1997) 3–20.
[28] O. Goldschmidt, D.S. Hochbaum, A polynomial algorithm for the k-cut problem for fixed k, Mathematics of Operations Research 19 (1) (1994).
[29] M. Gondran, M. Minoux, Graphes et Algorithmes. Eyrolles. Paris. 1995 (revision of previous editions 1971, 1985). 588 pp. (in French). Graphs and Algorithms. Wiley. 1986 (in English).
[30] V. Guruswani, S. Khanna, R. Rajaraman, B. Shepherd, M. Yannakakis, Near-optimal hardness results and approximation algorithms for edge-disjoint paths and related problems. Proceedings of the ACM Symposium on Theory Computers, ACM Atlanta, GA, 1999, pp. 19–28.
[31] W. Hasan, R. Motwani, Coloring away communication in parallel query optimization, Proceedings of the 21st VLDB Conference, Swizerland, 1995, pp. 239–250.
[32] T.C. Hu, Multicommodity network flows, Operational Research 11 (1963) 344–360.
[33] T.C. Hu, Two-commodity cut-packing problem, Discrete Mathematics 4 (1973) 108–109.
[34] D. Karger, P. Klein, M. Thorup, C. Stein, N. Young, Better rounding algorithms for a geometric embedding relaxation of minimum multiway cut, STOC (1999).
[35] R.M. Karp, On the complexity of combinatorial problems, Networks 5 (1975) 45–68.
[36] J. Kleinberg, Approximation algorithms for disjoint paths problems. PhD thesis, MIT, Cambridge, MA, May 1996.
[37] J. Kleinberg, E. Tardos, Disjoint paths in densely embedded graphs, Proceedings of the 36th IEEE FOCS, 1995.
[38] J. Kleinberg, E. Tardos, Approximations for the disjoint paths problem in high-diameter planar networks, Proceedings of the 27th ACM STOC, 1995, pp. 26–35.
[39] P. Kolman, C. Scheideler, Improved bounds for the unsplittable flow problem, SODA (2002).
[40] E. Korach, M. Penn, A fast algorithm for maximum integral two-commodity flow in planar graphs, Discrete Applied Mathematics 47 (1993) 77–83.
[41] P. Kubat, A. Shulman, R. Vachani, J. Ward, Multicommodity flows in ring networks, INFORMS Journal on Computing 8 (3) (1996) 235–242.
[42] F.T. Leighton, S. Rao, An approximate max-flow min-cut theorem for uniform multicommodity flow problems with applications to approximation algorithms, Journal of ACM 46 (6) (1999) 787–832.
[43] L. Létocart. Problémes de multicoupe et multiflot en nombres entiers, Thèse de doctorat en informatique, CEDRIC-CNAM, 2002.

[44] L. Létocart, F. Roupin, A semidefinite approach to solve multicut in trees, JOPT (Montréal) CEDRIC report. 2002.
[45] M.S. Levine, Fast randomized algorithms for computing minimum {3,4,5,6}-way cuts, SODA (2000) 735–742.
[46] A. Löbel, Vehicle scheduling in public transit and lagrangian pricing, ZIB Berlin, 1997, Preprint SC 97-16.
[47] M.V. Lomonosov, Multiflow feasibility depending on cuts, Graph Theory Newsletters 9 (1979) 4.
[48] J.F. Lynch, The equivalence of theorem proving and the interconnection problem, ACM SIGDA Newsletters 5 (1975) 31–65.
[49] M. Middendorf, F. Pfeiffer, On the complexity of the disjoint path problem, Combinatorica 13 (1993) 97–107.
[50] J. Naor, L. Zosin, A 2-approximation algorithm for the directed multiway cut problem, Proceedings of the 38th Annual Symposium on Foundations of Computer Science FOCS, 1997, pp. 548–553.
[51] J.B. Orlin, A faster strongly polynomial minimum cost flow algorithm, Operations Research 41 (2) (1993) 338–349.
[52] C.H. Papadimitriou, M. Yannakakis, Optimization, approximation and complexity classes, Journal of Computer System and Science 43 (1991) 425–440.
[53] C. Picouleau, Ordonner-Réordonner. Habilitation à Diriger les Recherches, Paris VI and CEDRIC-CNAM, 2000.
[54] S. Rajagopalan, Two commodity flows, Operations Research Letters 15 (1994) 151–156.
[55] N. Robertson, P.D. Seymour, Graphs minors XIII: The disjoint paths problem, Journal of Combinatorial Theory, Series B 63 (1995) 65–110.
[56] B. Rothschild, A. Whinston, On two-commodity network flows, Operations Research 14 (1966) 377–387.
[57] F. Roupin, From Linear to Semidefinite Programming: An Algorithm to obtain Semidefinite Relaxations for bivalent Quadratic Problems, Cedric report RC388 http://cedric.cnam.fr. (2002).
[58] A. Schrijver, Homotopic routing methods, in: B. Korte, L. Lovasz, H.J. Prömel, A. Schrijver (Eds.), Paths, Flows and VLSI-Layout. Algorithms and Combinatorics 9, Springer-Verlag, Berlin, 1990, pp. 329–371.
[59] A. Sebo, Integer plane multicommodity with a fixed number of demands, Journal of Combinatorial Theory, Series B 59 (1993) 163–171.
[60] P.D. Seymour, A two-commodity cut survey, Discrete Mathematics 23 (1978) 177–181.
[61] D.B. Shmoys, Cut problems and their application to divide-and-conquer, in: D. Hochbaum (Ed.), Approximation Algorithms for NP-hard Problems, PWS Publishing Company, Boston, 1997, 192–234.
[62] A. Srivastav, P. Stangier, On complexity, representation and approximation of integral multicommodity flows, Discrete Applied Mathematics 99 (2000) 183–208.
[63] E. Tardos, V.V. Vazirani, Improved bounds for the max-flow min-multicut ratio for planar and $K_{r,r}$-free graphs, Information, Proceedings and Letters 47 (1993) 77–80.