Available online at www.sciencedirect.com
SCIENCE DIRECT®

Discrete Applied Mathematics 128 (2003) 337–353

DISCRETE APPLIED MATHEMATICS

www.elsevier.com/locate/dam

The base-matroid and inverse combinatorial optimization problems

Mauro Dell'Amico[a], Francesco Maffioli[b], Federico Malucelli[b]

[a] *Dipartimento di Scienze e Metodi dell'Ingegneria, Università di Modena e Reggio Emilia, Viale Allegri 13, 42100 Reggio Emilia, Italy*

[b] *Dipartimento di Elettronica e Informazione, Politecnico di Milano, Via Ponzio 34/5, 20133 Milano, Italy*

Received 6 April 2001; received in revised form 21 December 2001; accepted 15 July 2002

Abstract

A new matroid is introduced: this matroid is defined starting from any matroid and one of its bases, hence we call it *base-matroid*. Besides some properties of the base-matroid, a non-trivial algorithm for the solution of the related matroid optimization problem is presented. The new matroid has application in the field of inverse combinatorial optimization problems. We discuss in detail the LP formulation of the inverse matroid optimization problem and we propose an efficient algorithm for computing its primal and dual solutions.

Keywords: Matroids; Inverse optimization; Combinatorial optimization

1. Introduction

We introduce a new matroid, which we call *base-matroid*: the name is motivated by the fact that it is defined starting from any matroid and one of its bases. We refer to Oxley [14] for fundamentals of Matroid Theory.

Given a matroid M defined over a ground set E and having $\mathscr{F}$ as its family of independent sets, let B be one of the bases of M. Any closed set θ such that the cardinality of its intersection with B is equal to its rank is called *saturated closed set*. We denote by $\mathscr{F}_B$ the family of subsets S of E having the property that the cardinality of the intersection of S with any saturated closed set θ is not greater than the rank

E-mail address: dellamico@unimore.it (M. Dell'Amico).

PII: S0166-218X(02)00498-5

of θ. We prove in Section 2 that $M_B = (E, \mathscr{F}_B)$ is indeed a matroid on E, and more specifically a *transversal* matroid. Therefore given a non-negative weighting of the elements of E, the problem of finding a base of M_B such that the sum of the weights of its elements is maximum can be solved by the greedy algorithm. We presents an efficient implementation of the greedy algorithm requiring only $O(mn + n^3 + m\varphi)$ time, where $m = |E|$, n is the rank of E in M and φ is the complexity of finding the unique circuit in M formed by adding to the given base B an element not in the base.

An application of base-matroids is in the field of *inverse combinatorial optimization problems* (ICOP). In general these problems ask for the "smallest" perturbation of the weighting of the elements of the ground set E which would make a given feasible subset of E optimal. Many different inverse problems have been addressed in the recent literature [1,3–6,10]. Their applications range from traffic control to seismic tomography (see [3,12]). We focus on one of the most basic ICOPs, namely the *inverse matroid problem*. Given a matroid $M = (E, \mathscr{F})$ and a target base B of M, this problem looks for perturbation parameters δ to be added to the weight c of the elements of E so that B becomes optimal for the (direct) matroid optimization problem with the new weighting $w_e = c_e + \delta_e$, $\forall e \in E$, and a given function of the perturbation parameters is minimized. We consider the case in which such function is given by the sum of the absolute values of δ_e's i.e., the L_1-norm of the perturbation parameter δ. It is possible to solve this problem through a straightforward algorithm. We show in Section 3 that a more efficient algorithm can be obtained by suitably exploiting an ad hoc base-matroid.

2. Base-matroids

Consider a *matroid* $M = (E, \mathscr{F})$ defined by a *ground set of elements* E and a *family of independent sets* $\mathscr{F} \subseteq 2^E$. The three following axioms define a matroid [14]:

(a.1). $\emptyset \in \mathscr{F}$;
(a.2). if $X \in \mathscr{F}$ and $X' \subseteq X$ then $X' \in \mathscr{F}$;
(a.3). if X and Y are in $\mathscr{F}$, and $|X| < |Y|$, then there is an element e of $Y \setminus X$ such that $X \cup \{e\} \in \mathscr{F}$.

Given a weighting function $c: E \to \mathbb{R}^+$, let $C: \mathscr{F} \to \mathbb{R}^+$, be defined as $C(S) = \sum_{e \in S} c_e$. The *matroid optimization problem* is to determine

$$\max_{S \in \mathscr{F}} C(S). \tag{1}$$

The *rank* $r(S)$ of a set $S \subseteq E$ is the cardinality of the largest $S' \subseteq S$ such that $S' \in \mathscr{F}$. Given a set $S \in \mathscr{F}$, let us denote with $\sigma(S)$ the *closure* of S, i.e. the superset of S obtained by adding to S all elements e such that $r(S \cup \{e\}) = r(S)$. A set θ is *closed* if $\theta = \sigma(\theta)$, i.e. $r(\theta \cup \{e\}) = r(\theta) + 1$ for all $e \in E \setminus \theta$. In the following we denote by Θ the set of all closed sets of M. Remember also that $\mathscr{F} = \{S \subseteq E\colon |S \cap A| \leqslant r(A), \forall A \subseteq E\}$ and since $|S \cap A| \leqslant |S \cap \sigma(A)|$ and $r(A) = r(\sigma(A))$ then

$$\mathscr{F} = \{S \subseteq E\colon |S \cap \theta| \leqslant r(\theta),\ \forall \theta \in \Theta\}. \tag{2}$$

A *base* of matroid M is a set $B \in \mathscr{F}$ of maximum cardinality. The optimal solution of a matroid optimization problem is a base. Note that all the bases of a matroid have the same cardinality equal to $r(E)$. Let us define $m = |E|$ and $n = r(E) = |B|$. The following definitions refer to a given a matroid $M = (E, \mathscr{F})$ and a base B.

Definition 1. A set $\theta \subseteq E$ is called *saturated* if $|\theta \cap B| = r(\theta)$. If $\theta \in \Theta$, we have a *saturated closed* set. The set of all the saturated closed sets of M, with respect to base B, is denoted by Θ_B.

Note that given any saturated closed set θ we have $\sigma(\theta \cap B) = \theta$.

Definition 2. Given a closed set $\theta \in \Theta_B$, we call $\theta \cap B$ the *skeleton* of θ with respect to base B.

A *circuit* is a minimal dependent set, i.e. a set $S \notin \mathscr{F}$ such that for each $i \in S$, $S \setminus \{i\} \in \mathscr{F}$. Given a base B and an element $i \in E \setminus B$, the *fundamental circuit* of i is the minimal subset of $B \cup \{i\}$ which is not in $\mathscr{F}$ (note that i always belongs to its fundamental circuit). More specifically, calling $\gamma(i)$ the unique minimal subset of B such that $\gamma(i) \cup \{i\} \notin \mathscr{F}$, then $\gamma(i) \cup \{i\}$ is a fundamental circuit. Moreover, let $\theta_B(i) = \sigma(\gamma(i) \cup \{i\})$ denote the closure of the fundamental circuit $\gamma(i) \cup \{i\}$. We call $\theta_B(i)$ the *fundamental closed set* associated with the base B and the element $i \in E$ (note that $\theta_B(i) = \{i\}$ when $i \in B$.) Finally, observe that the rank of a fundamental closed set is $r(\theta_B(i)) = r(\gamma(i))$, thus any fundamental closed set is also saturated.

The following definition introduces a new matroid called *base-matroid*. The name is due to the fact that the new matroid is obtained from a given matroid, by considering a subset of constraint, in particular those saturated by a given base.

Definition 3. Given a matroid $M = (E, \mathscr{F})$ and a base B let $\mathscr{F}_B = \{S \subseteq E : |S \cap \theta| \leqslant r(\theta), \forall \theta \in \Theta_B\}$ and observe that $\mathscr{F} \subseteq F_B$, see (2). We call $M_B = (E, \mathscr{F}_B)$ the *base-matroid* induced by base B.

In the following we prove that $M_B = (E, \mathscr{F}_B)$ is indeed a matroid.

Property 1. *Given any saturated closed set $\theta \in \Theta_B$ and any $e \in \theta \setminus B$, then $\gamma(e) \subseteq \theta$.*

Proof. By definition of saturated closed set $r(\theta) = |\theta \cap B|$, hence θ can be obtained as the closure of the skeleton $\theta \cap B$. It immediately follows that $e \in \theta$ implies that $r(\theta \cap B) = r(\theta \cap B \cup \{e\})$, thus $\gamma(e) \subseteq \theta \cap B$. $\square$

Note that this property does not hold for $\theta \in \Theta \setminus \Theta_B$. For instance consider the matric matroid whose elements are the columns of the following matrix:

$$\begin{pmatrix} 5 & 3 & 1 & 4 & 8 \\ 3 & 1 & 1 & 2 & 4 \\ 0 & 2 & 0 & 2 & 2 \end{pmatrix}$$

and independence is over the field of real. Consider the base $B=\{e_1,e_2,e_3\}$ and the closed set $\theta=\{e_4,e_5\}$ which is not saturated. For both elements $e_4,e_5\in\theta\setminus B$ the corresponding γ sets are $\gamma(e_4)=\{e_2,e_3\}$ and $\gamma(e_5)=\{e_1,e_2\}$ (in fact $e_4=e_2+e_3$ and $e_5=e_1+e_2$), which are not contained in θ.

Property 2 (Cunningham [7]). *The intersection and union of saturated sets are also saturated.*

Definition 4. Given a set $S\subseteq E$ we call *base-mapping* a function $a:S\to B$ such that $a(i)\in\gamma(i)$ if $i\in S\setminus B$, $a(i)=i$ if $i\in B$, and $a(i)\neq a(j)$ $i\neq j$.

In the following, we denote by $a(S)$ the mapping of a set S into B (i.e. $a(S)=\{a(i)\colon i\in S\}$).

Theorem 5. *Given set $\mathscr{F}_B$ and a set $S\subseteq E$, there exists a base-mapping $a:S\to B$ if and only if $S\in\mathscr{F}_B$.*

Proof. First we prove that if there is a base-mapping $a:S\to B$ then $S\in\mathscr{F}_B$. Consider any saturated closed set $\theta\in\Theta_B$. Since there exists the base-mapping for S, then $|S\cap\theta|=|a(S\cap\theta)|$. To prove the independence of S we show that $|a(S\cap\theta)|\leqslant r(\theta)$. We first note that $a(S\cap\theta)\subseteq\theta$, indeed, by definition of base-mapping: (i) $a(i)=i$ for each $i\in S\cap\theta\cap B$; (ii) $a(i)\in\gamma(i)$ for each $i\in(S\cap\theta)\setminus B$, but $\gamma(i)\subseteq\theta$ for each element $i\in\theta\setminus B$ (see Property 1). Recalling that $a(i)\in B$ for all $i\in S$ we obtain $|a(S\cap\theta)|\leqslant|B\cap\theta|=r(\theta)$ and the independence of S follows.

Now we prove the second part of the theorem. Suppose that $S\in\mathscr{F}_B$ and consider the family $(A_e\colon e\in B)$ with $A_e=\{f\in E\colon f=e \text{ or } e\in\theta_B(f)\}$. Define the bipartite graph $G=(V_1\cup V_2,L)$ where V_1 has one vertex v_i' for each element $i\in S$, V_2 has one vertex v_j'' for each element $j\in B$, and the edge set L has one edge (v_i',v_j'') for each $i\in S\cap B$ and $j\in A_i$. One can see that each matching $\mathscr{M}$ of G with $|\mathscr{M}|=|V_1|$ corresponds to a base-mapping for S obtained by setting $a(i)=j$ for each edge $(v_i',v_j'')\in\mathscr{M}$. To conclude the proof it is sufficient to show that G has always a matching $\mathscr{M}$ with cardinality of $|V_1|$. According to a well-known result of Hall such a matching exists if and only if for each $H\subseteq V_1$ and $N(H)=\{v_j''\colon (v_i',v_j'')\in L, v_i'\in H\}$, then $|H|\leqslant|N(H)|$. Consider the set $U=\bigcup_{i:v_i'\in H}\theta_B(i)$ and recall that: (a) $\theta_B(i)\cap B=\gamma(i)$ for $i\in E\setminus B$, $\theta_B(i)\cap B=\{i\}$ for $i\in B$; and (b) U is saturated (see Property 2).

Observing that $N(H)=\{v_j''\colon j\in\gamma(i), v_i'\in H,\ i\notin B\}\cup\{v_j''\colon v_j'\in H,\ j\in B\}$ and using (a), (b) above we obtain $|N(H)|=|B\cap U|=r(U)$. Since S is independent in M_B and $\sigma(U)\in\Theta_B$ we have $|S\cap U|\leqslant|S\cap\sigma(U)|\leqslant r(\sigma(U))=r(U)$ from which follows $|S(H)\cap U|\leqslant r(U)$, where $S(H)=\{i\in S\colon v_i'\in H\}$. But $S(H)\subseteq U$ (since $i\in\theta_B(i)$ $\forall i\in S$), hence $|S(H)|\leqslant r(U)$ and $|H|\leqslant|N(H)|$ holds. □

The existence of the base-mapping a for S proves that S is a *partial transversal* of family $(A_e\colon e\in B)$ above, see [14, Chapter 1.6]. From a theorem by Edmonds and Fulkerson [8] we have:

Theorem 6. *$M_B = (E, \mathscr{F}_B)$ is a (transversal) matroid.*

2.1. Base-matroid optimization

Since M_B is transversal, the corresponding optimization problem is the "sequencing problem" (see Lawler [11, p. 278]) or the "job assignment problem" (see Oxley [14, p. 65]) and it can be solved with an adaptation of the general greedy algorithm where we need only to decide when a set S of elements is independent, i.e. we need to find a matching in G_S. More precisely in order to decide if adding an element e to and independent set S is still independent, we have to expand G_S to $G_{S\cup\{e\}}$ and to perform an $\mathscr{M}$-augmenting path starting from v'_e.

The construction of all the graphs used by the above algorithm can be done in $O(m\varphi)$ time (since for each element we have to determine its fundamental circuit), whereas the search for an $\mathscr{M}$-augmenting path requires at most $O(n^2)$ computing time (since the two vertex sets have at most n vertices each). It follows that this implementation of the greedy algorithm runs in $O(mn^2 + m\varphi)$ time.

The complexity of this greedy can be improved as follows. First observe that, at each iteration, the search for an $\mathscr{M}$-augmenting path may succeed or not. Since at most n successful augmentations are performed, the global number of operations due to such augmentations is $O(n^3)$. Now consider an iteration in which the $\mathscr{M}$-augmenting path does not exist. We will show that either the computation of the possible $\mathscr{M}$-alternating tree requires $O(n)$ time, or we can reduce the number of vertices of V_2 . It follows that all the unsuccessful iterations require $O(mn + n^3)$ time thus improving our previous bound and yielding an overall $O(mn + n^3 + m\varphi)$ algorithm.

The following property holds.

Property 3. *At any unsuccessful iteration of the algorithm consider the associated independent set S, the corresponding matching $\mathscr{M}$ (with $|\mathscr{M}| = |S|$), and the element $e \notin S$ such that there is no $\mathscr{M}$-augmenting path in $G_{S\cup\{e\}}$ emanating from v'_e. Let $R(e) \subseteq V_2$ be the set of vertices reachable from v'_e by means of $\mathscr{M}$-alternating paths. Then all edges of $\mathscr{M}$ with a vertex in $R(e)$ do not belong to any $\mathscr{M}$-augmenting path emanating from a vertex associated with an element in $E \setminus S$, in any subsequent iteration of the algorithm.*

Proof. Since there is no $\mathscr{M}$-alternating path starting from the vertex in $R(e)$ and ending with an $\mathscr{M}$-exposed vertex of V_2. It immediately follows that no $\mathscr{M}$-augmenting path starting from another vertex v'_f, corresponding to an element $f \in E \setminus S \cup \{e\}$, can use a vertex of $R(e)$, otherwise an $\mathscr{M}$-augmenting path would exist also for v'_e. □

From the above Property 3 we have that when the current element e cannot be added to the partial solution S, then all vertices in $R(e)$ can be removed from the graph. Let us consider an algorithm which, at any unsuccessful iteration, does not add vertex e to the graph and deletes the vertices of $R(e)$. In order to check if e can be added to the current solution we try to start an $\mathscr{M}$-augmenting path from each possible edge (v'_e, v''_j) with $j \in \gamma(e)$. If all the vertices $v''_j : j \in \gamma(e)$ have been removed from V_2 in previous iterations, then the number of operations performed is $O(|\gamma(e)|) \leqslant O(n)$ and this kind of iterations may occur at most $O(m)$ times thus yielding an $O(mn)$ running time. If otherwise we can grow a tree of $\mathscr{M}$-alternating paths, then we delete the corresponding vertices of V_2. In this case each iteration requires $O(n^2)$ operations, but it occurs at most n times since each iteration removes at least one vertex of V_2. Therefore the global computational effort for all the unsuccessful iterations is $O(mn + n^3 + m\varphi)$, which determines the global complexity of the algorithm.

Let us momentarily return to the previous version of the algorithm which does not reduce the vertex set. The relations among the sets $R(e)$ defined above and the saturated closed sets are exploited in the following property which will be used in the next Section 2.2.

Property 4. *At any unsuccessful iteration let $R(e)$ be defined as in Property 3. The set $B''=B''(e)=\{j\in B\colon v''_j \in R(e)\}$ is the skeleton of the saturated closed set $\theta=\sigma(B'')$ and θ is saturated for matroid M_B and base B_G obtained with the greedy algorithm.*

Proof. Consider the current matching $\mathscr{M}$ and let us define $B'=B'(e)=\{i\in S\colon (v'_i, v''_j)\in \mathscr{M}, v''_j \in R(e)\}$ and note that it is independent for M_B by construction, since it is a subset of the current solution obtained by the greedy algorithm. Observe that, by definition, $|B'| = |B''|$ and B'' is the skeleton of the set θ, which is saturated for M, moreover, $B' \subset \theta$ by construction.

We now prove that θ is saturated also for M_B. Let r_B denote the rank function of the base-matroid M_B. We have just shown that $B' \subset \theta$, hence $r_B(B') \leqslant r_B(\theta)$, but $\theta \in \Theta_B$, so from Definition 3 we have $r_B(\theta) \leqslant r(\theta)$. Now observe that the skeleton of a closed set saturated for M is independent for M_B, hence $r_B(\theta) = r(\theta)$. Further note that due to the independence of B' in M_B it is $r_B(B') = |B'|$, so we obtain

$$|B'| = r_B(B') \leqslant r_B(\theta) = r(\theta) = |B''|.$$

Recalling $|B'| = |B''|$ and since $B' \subseteq S \subseteq B_G$, we conclude that $|B'| = r_B(\theta)$ and θ is also saturated by B_G, for matroid M_B, B' being the skeleton of θ for M_B. □

2.2. Linear programming formulation

In this section we propose a linear programming model for optimizing a linear function on a base-matroid. Considering the greedy algorithm for the base-matroid presented in the previous section, the solution of the primal problem can be easily obtained. An efficient method to obtain the dual solution is less trivial and will be the main concern of this section.

Given a matroid $M = (E, \mathscr{F})$ with rank function r and weighting function c, it is well known that the corresponding optimization problem is equivalent to the following continuous linear programming problem (see e.g. [13]).

$$\text{(P)} \quad \max\left\{cx\colon \sum_{e\in\theta} x_e \leqslant r(\theta)\ \forall \theta\in\Theta,\ x\in\mathbb{R}^m_+\right\}.$$

The polytope vertices of problem P belong to $\{0,1\}^n$, hence each variable x_e takes value 1 if the element e is selected, and value zero otherwise. The dual of P is

$$\text{(D)}\quad \min\left\{ry\colon \sum_{\theta: e\in\theta} y_\theta \geqslant c_e\ \forall e\in E,\ y\in\mathbb{R}_+^{|\Theta|}\right\}$$

and the complementary slackness conditions of pair P–D are

$$\left(\sum_{e\in\theta} x_e - r(\theta)\right) y_\theta = 0,\quad \theta\in\Theta, \tag{3}$$

$$\left(\sum_{\theta: e\in\theta} y_\theta - c_e\right) x_e = 0,\quad e\in E. \tag{4}$$

The base-matroid optimization problem associated with the target base B is

$$\text{(BMP)}\quad \max\{cx'\colon x'\in PB\cap\{0,1\}^m\},$$

where

$$PB = \left\{x'\in\mathbb{R}_+^m\colon \sum_{e\in\theta} x'_e \leqslant r(\theta)\ \forall\theta\in\Theta_B\right\}.$$

The continuous relaxation of this problem is

$$\text{(CBMP)}\quad \max\{cx'\colon x'\in PB,\ x'_e\geqslant 0 \text{ for } e\in B,\ 0\leqslant x'_e\leqslant 1 \text{ for } e\in E\setminus B\}.$$

Note that the unit upper bounds must be explicitly given only for variables associated with elements in $E\setminus B$: indeed for each $e\in B$ we have $\sigma(\{e\})\cap B=\{e\}, \sigma(\{e\})\in\Theta_B$ and hence,

$$x'(\sigma(\{e\})) = x'_e + x'(\sigma(\{e\})\setminus\{e\}) \leqslant r(\sigma(\{e\})) = 1 \tag{5}$$

and since $x'(\sigma(\{e\})\setminus\{e\})\geqslant 0$ we have $x'_e\leqslant 1$.

In the following we will prove that similarly to the case of the classical matroid problem, $PB\cap[0,1]^m$ is defined on an integral polytope, hence CBMP is a valid formulation for the base-matroid optimization problem. Consider the dual of CBMP

$$\text{(DCBMP)}\quad \min ry' + \mathbf{1}\mu \tag{6}$$

$$\sum_{\theta\in\Theta_B: e\in\theta} y'_\theta \geqslant c_e,\ e\in B, \tag{7}$$

$$\sum_{\theta\in\Theta_B: e\in\theta} y'_\theta + \mu_e \geqslant c_e,\ e\in E\setminus B, \tag{8}$$

$$y'\in\mathbb{R}_+^{|\Theta_B|}, \tag{9}$$

$$\mu\in\mathbb{R}_+^m \tag{10}$$

and the complementary slackness conditions of CBMP–DCBMP

$$x'_e\left(\sum_{\theta\in\Theta_B:e\in\theta} y'_\theta - c_e\right) = 0, \quad e\in B, \tag{11}$$

$$x'_e\left(\sum_{\theta\in\Theta_B:e\in\theta} y'_\theta + \mu_e - c_e\right) = 0, \quad e\in E\setminus B, \tag{12}$$

$$y'_\theta\left(\sum_{e\in\theta} x'_e - r(\theta)\right) = 0, \quad \theta\in\Theta_B, \tag{13}$$

$$\mu_e(x'_e - 1) = 0, \quad e\in E\setminus B. \tag{14}$$

The optimal solution to DCBMP can be obtained with a procedure similar to that used to compute the optimal solution of problem D (the dual of the generic matroid problem, see [13]), but giving zero value to each y'_θ with $\theta\notin\Theta_B$, and assigning suitable values to the μ variables.

Theorem 7. *Let* $B_G=\{e_1,\dots,e_n\}$ *be the solution to problem* BMP *obtained through the greedy algorithm with the ordering* $c_{e_1}\geqslant c_{e_2}\geqslant\cdots\geqslant c_{e_n}$, *let* $S_h=\{e_1,\dots,e_h\}$ *for* $h=1,\dots,n$, *and* $\theta_h = \mathrm{argmax}\{|\theta|\colon \theta\in\Theta_B, \theta\subseteq\sigma_B(S_h), e_h\in\theta, \theta \text{ is saturated w.r.t. } B_G\}$ ($\theta_h=\emptyset$ *if no such* θ *exists*), *where* σ_B *denotes the closure operator for matroid* M_B. *Moreover let* $\Gamma=\{\theta_h\neq\emptyset,\ h=1,\dots,n\}$ *and* $\pi(h)=\min\{k>h\colon e_h\in\theta_k\}$ *for* $h=0,\dots,n-1$. *An optimal solution to* DCBMP *can be computed through the following procedure*:

Step 1: *set* $y'=0$; $\mu=0$;

Step 2: *set* $y'_{\theta_n}=c_{e_n}$;

for $h=n-1,\dots,1$, *if* $\theta_h\in\Gamma$ *then set* $y'_{\theta_h}=c_{e_h}-c_{e_{\pi(h)}}$,

otherwise set $\mu_{e_h}=c_{e_h}-c_{e_{\pi(h)}}$.

Proof. The dual values (y',μ) computed through steps 1–2 are clearly non-negative. Consider the primal solution $\{x'_e=1\colon e\in B_G,\ x'_e=0 \text{ otherwise}\}$ we show that the optimality conditions hold for $(x',(y',\mu))$. Observe that $\sigma_B(S_i)\subset\sigma_B(S_j)$ for $i<j$, i.e., the family of sets $\{\sigma_B(S_i)\}$ is *nested*. Given $\theta_i,\theta_j\in\Gamma$, with $i<j$ it is also $\theta_i\subset\theta_j$ (otherwise $\sigma(\theta_i\cup\theta_j)\subseteq\sigma(S_j)$, because of Property 2, is a saturated closed set larger than θ_j, a contradiction) and also Γ is nested.

Further observe that: (i) $\theta_n=E(\in\Gamma)$; (ii) θ_h is certainly non-empty if $e_h\in B_G\cap B$, hence μ_{e_h} may have a positive value only if $e_h\in B_G\setminus B$.

From (i) above and the fact that Γ is nested it immediately follows that given any $e\in E$ there is at least a set of Γ containing it. Let k be the smallest index such that $e\in\theta_k$, then

$$\sum_{\theta\in\Theta_B:e\in\theta} y'_\theta = \sum_{i=k}^{n} y'_{\theta_i} = c_{e_k}. \tag{15}$$

If $e \notin B_G$ then $c_e \leqslant c_{e_k}$, indeed $c_e > c_{e_k}$ and $e \notin B_G$ implies that there exists $i < k$: $e \in \theta_i, |\theta_i \cap \{e_1, \dots, e_{k-1}\}| = r(\theta_i)$ (see Property 4), contradicting the definition of k. Using (15) and the assignment of values to y' and μ one can see that (7) and (8) are satisfied with the '$\geqslant$' sign for all $e \in E \setminus B_G$. For each $e_h \in B_G$, if $e_h \in B$, then $e_h \in \theta_h$ (see (ii) above) and $k = h$, so (7) is satisfied with the '$=$' sign. If otherwise $e_h \notin B$, then two cases may occur: (a) $\theta_h \neq \emptyset$, then $e_h \in \theta_h$ and $k = h$; (b) $\theta_h = \emptyset$, so $k = \pi(h)$ and $\mu_{e_h} = c_{e_h} - c_{e_k}$. In both cases (8) is satisfied with the '$=$' sign.

The above reasoning also proves that the terms in parenthesis in (11) and (12) have value zero when $e \in B_G$. On the other side $x'_e = 0$ for each $e \notin B_G$, hence (11) and (12) hold. The variable y'_θ may be assigned a positive value only when $\theta \in \Gamma$, i.e. θ is saturated by B and B_G. It follows that $y'_\theta > 0$ only if $\sum_{e\in\theta} x'_e = r(\theta)$, hence (13) hold. Finally, μ_e is assigned a positive value only if $e \in B_G \setminus B$ (see again (ii) above) and also the last conditions (14) hold. □

The above theorem proves that the system $\{\sum_{e\in\theta} x'_e \leqslant r(\theta)$ for $\theta \in \Theta_B$, $x'_e \leqslant 1$ for $e \in E \setminus B$, $x' \in \mathbb{R}^m_+\}$ is totally dual integral, hence $PB \cap [0,1]^m$ is an integral polytope and:

Theorem 8. CBMP *is a valid formulation for the base-matroid optimization problem.*

Let us discuss the computational complexity of finding the saturated sets $\theta_h \in \Gamma$ defined in Theorem 7 and necessary to compute the values of the non-zero dual variables. We propose an implementation in which each θ_h is computed at the end of the algorithm. During the execution of the greedy algorithm, instead, we determine sets $\eta_i = \theta_i \setminus \theta_{i-1}$.

We use a version of the greedy algorithm which terminates only when all the m elements of E have been considered. We initialize the sets $\eta_1 = \eta_2 = \cdots = \eta_n = \emptyset$. For each element e we determine the smallest index h such that $e \in \theta_h$ (see below for details), and we add e to η_h. At the end of the algorithm we compute $\theta_h = \bigcup_{i=1}^{h} \eta_i$, for each $h = 1, \dots, n$ such that $\eta_h \neq \emptyset$.

The key aspect of the procedure is the computation of the correct index h. For sake of simplicity we first introduce a method based on a (not efficient) implementation of the greedy which adds to the graph also the vertices corresponding to elements not inserted in B_G and does not delete the $R(e)$ sets from the bipartite graph used to perform the tests of independence (see Property 3). Then we show how to improve this unefficient greedy with an implementation based on a labeling technique.

Consider a generic iteration of the greedy and let e be the element currently examined. Three cases may occur:

1. $e \in B \cap B_G$: Let $e_l = e$ (i.e. e is the lth element added to the current partial solution) then the required index is $h = l$ (see the proof of Theorem 7).

2. $e \notin B_G$: Recall that we have computed $R(e)$ without finding any $\mathscr{M}$-augmenting path. From Property 4 we know that set $\theta = \sigma(B''(e))$ is saturated by base B for M and by the current partial base for M_B. Let h be the smallest index such that $e \in \theta_h$, $\theta_h \in \Gamma$. Observe that: (i) set $\theta \subseteq \sigma_B(S_h)$, since it is saturated by $\{e_1, \dots, e_h\}$; and (ii) $\theta \subseteq \theta_h$, otherwise $\theta \cup \theta_h$ is saturated for both matroids and has larger cardinality than θ_h: a contradiction. Further observe that due to the fact that $e \in \sigma_B(S_h)$ and $e \notin \sigma_B(S_i)$, for $i < h$, then e can enter into a set of Γ only together with e_h. It follows that $e_h \in \theta$, so during the execution of the algorithm we can compute the value of index h by scanning set $B'(e)$ $(=\theta \cap \{e_1, \dots, e_h\})$ and identifying the last element of B_G inserted into it.
3. $e \in B_G \setminus B$: In this case we are not guaranteed to identify efficiently the required index for all elements, so we postpone the insertion of e in the suitable η set at the end of the greedy. More precisely when all the m elements have been examined we consider, in turn, each element $e \in B_G \setminus B$. We temporary create a copy, say $\tilde{e}$, of e and we compute $R(\tilde{e})$. The required index is found as in case 2, by considering set $\theta = \sigma(B''(\tilde{e}))$.

We now show how to implement the above procedure without computing explicitly all the sets $\sigma(B''(e))$. We know that if a vertex is reached by an $\mathscr{M}$-alternating path, at an unsuccessful iteration, then in the next iterations it can not belong to any $\mathscr{M}$-augmenting path (see Property 4). In Section 2.1, we have already shown that deleting these vertices we can reduce the computational complexity of the greedy, however for computing the dual values we should entirely scan each $\mathscr{M}$-alternating tree to compute the index of the η set in which the current element has to be inserted (cases 2 and 3 above). Instead of rescanning a tree we can maintain a trace of the previously examined trees by using the following simple labeling technique. When an unsuccessful iteration occurs we associate at each vertex $v''_j \in V_2(e)$ a label storing the index of the last element inserted in B_G and related with one of the vertices of the subtree rooted at v''_j. In the next iterations if we reach vertex v''_j, we can stop the search for this branch of the $\mathscr{M}$-alternating tree since the whole information needed to compute the dual value is stored in the label. The smallest index h such that $e \in \theta_h$ is identified by considering all the elements associated to vertices explicitly reached and the labels of the leaves. Using this trick we can return to the original implementation which, at each unsuccessful iteration, does not add the vertex associated with the current element $e \notin B_G$ to the graph, and deletes all the vertices in the $R(e)$ set. The additional computational effort required to compute the dual values is $O(mn)$ for identifying the correct η sets during the execution of the greedy, plus $O(n^3)$ for the computations due to the elements in $B_G \setminus B$, so the following theorem holds.

Property 5. *The dual values defined by Theorem 7 can be determined during the execution of the greedy algorithm, without increasing its computational complexity.*

Example 9. Let us consider the graphic matroid depicted in Fig. 1, and let us be given the target base $B = \{a, b, c, d, e, j, l, m, n\}$ (thick edges). The weights associated with the

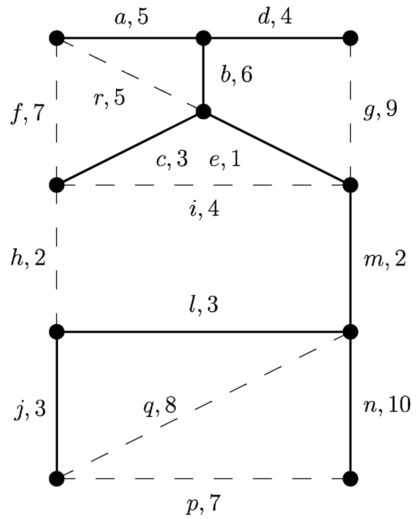


Fig. 1. The graph and the target base (thick edges).

edges are reported in the following table, sorted by non-increasing value (breaking ties by the lexicographic order of the names):

	1	*2*	*3*	*4*	*5*	*6*	*7*	*8*	*9*	*10*	*11*	*12*	*13*	*14*	*15*	*16*
Element	**n**	g	q	f	p	**b**	**a**	r	**d**	i	**c**	**j**	**l**	h	**m**	**e**
Cost	*10*	*9*	*8*	*7*	*7*	*6*	*5*	*5*	*4*	*4*	*3*	*3*	*3*	*2*	*2*	*1* .

By applying the greedy algorithm of Section 2.1 we can augment the solution until we examine edge r. At that point the partial solution is $\{n, g, q, f, p, b, a\}$, the currently defined η sets are $\eta_1 = \{n\}$, $\eta_2 = \eta_3 = \eta_4 = \eta_5 = \emptyset$, $\eta_6 = \{b\}$ and $\eta_7 = \{a\}$. The non-empty η sets correspond to elements in $B \cap B_G$ (case 1). The elements $\{g, q, f, p\}$ will be inserted in the suitable η set at the end of the greedy (case 3). The bipartite graph used to determine the independence of element r is reported in Fig. 2a (the thick edges give the current base-mapping). The $\mathscr{M}$-alternating tree starting from vertex r' is given in Fig. 3a (in square brackets we report the label associated at each vertex, whereas in parenthesis we report the labels that will be associated at each vertex after the computation of the tree). No $\mathscr{M}$-augmenting path exists and set $\{a, b, r\}$ is dependent for the base-matroid. Examining set $B'(r) = \{a, b\}$ we find that element a is the last one added, so $\eta_h = \eta_7 = \{a, r\}$. In the next iteration we add d to B_G, we set $\eta_8 = \{d\}$ and we update the base-mapping (see Fig. 2b). Then we examine element i, we find the $\mathscr{M}$-alternating tree of Fig. 3b and we add i to η_8. Then we add c, j, l to η_7, η_5 and η_5, respectively. Element h enter in the solution with matching $[h', m'']$, element m is added to η_9 and element e is added to η_8. We have thus obtained the η sets: $\eta_1 = \{n\}$, $\eta_2 = \eta_3 = \eta_4 = \emptyset$,

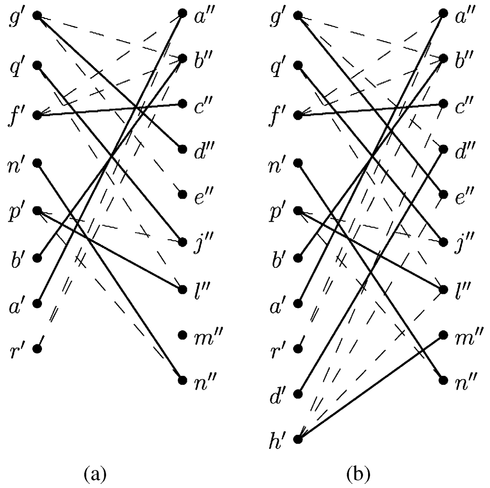


Fig. 2. The bipartite graphs and the base-mappings (thick edges).

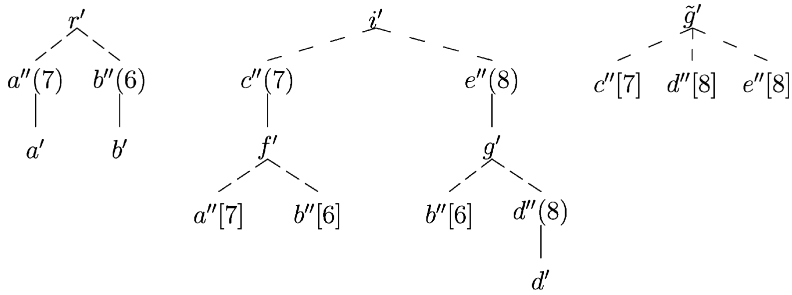
$\eta_5=\{j,l\}$, $\eta_6=\{b\}$, $\eta_7=\{a,c,r\}$, $\eta_8=\{d,e,i\}$, $\eta_9=\{m\}$. The base mapping is reported in Fig. 2b. The optimal solution of the base-matroid is thus: $\{g,q,f,n,p,b,a,d,h\}$; it should be observed that the solution is not feasible for the graphic matroid as it contains a cycle ($\{q,n,p\}$).

We now consider the elements in $B_G \setminus B$. We first duplicate g' obtaining $\tilde{g}'$ and we compute the corresponding $\mathscr{M}$-alternating tree, see Fig. 3c: element g is added to η_8. The next elements $\{q,f,p,h\}$ are added to η_5,η_7,η_5 and η_9 respectively. The final η and θ sets are:

$$\begin{aligned}
&\eta_1=\{n\} && \theta_1=\{n\},\\
&\eta_5=\{j,l,p,q\} && \theta_5=\{j,l,n,p,q\},\\
&\eta_6=\{b\} && \theta_6=\{b,j,l,n,p,q\},\\
&\eta_7=\{a,c,f,r\} && \theta_7=\{a,b,c,f,j,l,n,p,q,r\},\\
&\eta_8=\{d,e,g,i\} && \theta_8=\{a,b,c,d,e,f,g,i,j,l,n,p,q,r\},\\
&\eta_9=\{h,m\} && \theta_9=E.
\end{aligned}$$

The primal and dual solution are summarized in the following table.

	1	2	3	4	5	6	7	8	9
Element	n	g	q	f	p	b	a	d	h
Cost	*10*	*9*	*8*	*7*	*7*	*6*	*5*	*4*	*2*
Variable	y_{θ_1}	μ_g	μ_q	μ_f	y_{θ_5}	y_{θ_6}	y_{θ_7}	y_{θ_8}	y_{θ_9}
Value	*3*	*5*	*1*	*2*	*1*	*1*	*1*	*2*	*2*



(a) (b) (c)

Fig. 3. Three $\mathscr{M}$-alternating trees.

3. Inverse matroid problem

The *inverse matroid problem* can be stated as follows. Given a matroid $M=(E,\mathscr{F})$, a non-negative weighting function c, and a *target base* B of M (not necessarily optimal) find the *perturbation parameters* δ_e to be added to the weighting coefficients c_e, for each $e\in E$, such that B is optimal for the matroid problem defined by the new weights $w_e=c_e+\delta_e$, and a function of the values δ_e is minimized. In this paper we focus on the objective function given by the sum of the absolute values of δ_e, that is $\sum_{e\in E}|\delta_e|$.

Since B must be optimal for the weighting w, one can prove (see, e.g. [9]) that in the optimal solution of the inverse matroid problem $w_e\geqslant c_e$ for each $e\in B$ and $w_e\leqslant c_e$ for each $e\in E\setminus B$. Therefore, the inverse matroid problem is equivalent to finding the vector d which minimizes $\sum_{e\in E} d_e$ and such that B is an optimal base for the matroid problem with weights

$$w_e=\begin{cases} c_e+d_e & e\in B,\\ c_e-d_e & e\in E\setminus B.\end{cases} \tag{16}$$

A simple method for computing the optimal perturbations is based on the fact that a base is optimal if, for any $e\in B$, and each $f\in\gamma(e)$, $w_f\geqslant w_e$ holds. Therefore the problem is

$$\min\sum_{f\in B} d_f+\sum_{e\in E\setminus B} d_e$$

$$c_f+d_f\geqslant c_e-d_e e\in E\setminus B,\ \ f\in\gamma(e)$$

$$d_e\geqslant 0\ \ e\in E.$$

Its dual turns out to be a maximum weight matching problem on a bipartite graph with left vertex set in one-to-one correspondence with $E\setminus B$, right vertex set in one-to-one correspondence with B and one edge (e,f) for each $e\in E\setminus B, f\in\gamma(e)$ with cost (c_e-c_f). The computational complexity for solving the inverse matroid problem with this approach is thus $O(mn^2)$ plus $O(m\varphi)$ for constructing the graph. We next show how this complexity may be reduced by using the greedy algorithm for the base-matroid.

The target base B is optimal with respect to the new weights (16) if there exists a dual feasible vector y which satisfies the complementary slackness conditions (3)–(4), written with w instead of c. Reminding that the only saturated closed sets, with respect to B, are those of Θ_B, then (3) implies $y_\theta = 0$ for $\theta \notin \Theta_B$ and the inverse matroid problem can be formulated as follows:

$$(\mathrm{PI}) \quad \min \sum_{e \in E} d_e, \tag{17}$$

$$\sum_{\theta \in \Theta_B : e \in \theta} y_\theta = c_e + d_e \quad e \in B, \tag{18}$$

$$\sum_{\theta \in \Theta_B : e \in \theta} y_\theta \geqslant c_e - d_e \quad e \in E \setminus B, \tag{19}$$

$$y \in \mathbb{R}_+^{|\Theta_B|}, \tag{20}$$

$$d \in \mathbb{R}_+^m. \tag{21}$$

Constraint (18) derive from the optimality conditions (4), whereas (19) impose the feasibility of the dual solution y. In order to construct the optimal solution of problem PI let us consider the dual:

$$(\mathrm{DI}) \quad \max \sum_{e \in E} c_e x_e \tag{22}$$

$$\sum_{e \in \theta} x_e \leqslant 0 \quad \theta \in \Theta_B, \tag{23}$$

$$x_e \geqslant -1 \quad e \in B, \tag{24}$$

$$0 \leqslant x_e \leqslant 1 \quad e \in E \setminus B. \tag{25}$$

Using the transformation

$$x'_e = \begin{cases} 1 + x_e & e \in B, \\ x_e & e \in E \setminus B \end{cases} \tag{26}$$

problem DI can be rewritten as

$$(\mathrm{DI}') - \sum_{e \in B} c_e + \max \sum_{e \in E} c_e x'_e, \tag{27}$$

$$\sum_{e \in \theta} x'_e \leqslant r(\theta) \quad \theta \in \Theta_B, \tag{28}$$

$$x'_e \geqslant 0 \quad e \in B, \tag{29}$$

$$0 \leqslant x'_e \leqslant 1 \quad e \in E \setminus B. \tag{30}$$

Problem DI$'$ is a base-matroid optimization problem (see Section 2.2, problem CBMP) hence it can be efficiently solved by means of the greedy algorithm described in the previous section. In order to construct the solution of the inverse problem PI let us introduce the complementary slackness condition of PI–DI.

$$x_e\left(\sum_{\theta:\theta\in\Theta_B,e\in\theta} y_\theta + d_e - c_e\right) = 0 \quad e\in E\setminus B, \tag{31}$$

$$y_\theta\left(\sum_{e\in\theta} x_e\right) = 0 \quad \theta\in\Theta_B, \tag{32}$$

$$(x_e+1)d_e = 0 \quad e\in B, \tag{33}$$

$$(x_e-1)d_e = 0 \quad e\in E\setminus B. \tag{34}$$

Given the optimal solution x', obtained through the greedy algorithm of Section 2.1, the optimal solution of the inverse problem is obtained with the following procedure.
Procedure *InverseMatroid*()
step i: Determine the optimal solution (y,μ) of the dual of problem DI$'$ (see Theorem 7).
step ii: Determine the values of x_e through the inverse of (26), that is

$$x_e = \begin{cases} x'_e - 1 & e\in B, \\ x'_e & e\in E\setminus B \end{cases}$$

and note that this solution is optimal for problem DI.
step iii: For each $e\in B$ define the value of d_e as follows: if $x'_e=1$ set $d_e=0$, otherwise $(x'_e=0)$ set $d_e=\sum_{\theta\in\Theta_B:e\in\theta} y_\theta - c_e$.
step iv: For each $e\in E\setminus B$ define the value of d_e as follows: if $x'_e=1$ set $d_e=c_e-\sum_{\theta\in\Theta_B:e\in\theta} y_\theta$, otherwise $(x'_e=0)$ set $d_e=0$.

Theorem 10. *The solution d,y determined through the above procedure* InverseMatroid *is optimal for* PI.

Proof. We have already observed that x defined at step ii is a feasible solution for DI. To prove the thesis we show that d,y is a feasible solution for PI, and that x,d,y, satisfy the complementary slackness conditions (31)–(34).

First note that y has non-negative values (see Theorem 7), then consider separately the case $e\in B$ and $e\in E\setminus B$.

Case $e\in B$ (step iii). From (7) we know that $d_e=\sum_{\theta\in\Theta_B:e\in\theta} y_\theta - c_e \geqslant 0$ satisfying (18). If we set $d_e=0$, we have $x'_e=1$ and from (11) condition (18) follows again.

Case $e\in E\setminus B$ (step iv). From (12) we know that when $x'_e=1$ we have $c_e-\sum_{\theta:e\in\theta} y_\theta \geqslant 0$ hence d_e is assigned a non-negative value and both (19) and (31) hold. When $x'_e=0$ (31) trivially holds, whereas form (14) we have $\mu_e=0$ and (8) implies that (19) holds.

We conclude the proof by observing that the remaining condition (32) directly descends from (13) by applying transformation (26), and that (33) and (34) hold by construction of d. □

Corollary 11. *The computational complexity of procedure* Inverse Matroid *is* $O(mn + n^3 + m\varphi)$.

Proof. Step i requires to apply the greedy algorithm of Section 2.1, whereas steps ii–iv can be implemented in $O(m)$ time. □

Example 12 (continued). The original base of the graphic matroid of Fig. 1 has value 37 and applying procedure *Inverse Matroid* we obtain a dual solution having value 58. Then at step iii we set to zero the perturbation associated with the elements in $B \cap B_G$ and we compute the following values for the elements in $B \setminus B_G$: $d_c = 2$, $d_e = 3$, $d_j = 4$, $d_l = 4$, $d_m = 0$. At step iv the computation of the perturbation of the elements in $B_G \setminus B$ gives $d_f = 2$, $d_g = 5$, $d_h = 0$, $d_p = 0$, $d_q = 1$, whereas the remaining values are set to zero. The optimal solution of the inverse problem has value 21.

The particular case of graphic matroids has been studied by Ahuja et al. [2]. It is well known that a spanning tree of a graph is the base of a graphic matroid, therefore the inverse spanning tree problem (ISTP) can be immediately modeled by means of a base-matroid and procedure *InverseMatroid* is an alternative approach for solving ISTP. In this section we briefly compare the two approaches.

Given a graph with n vertices and m edges the basic algorithm of Ahuja et al. [2], solves ISTP in $O(n^3)$ time. Using a cost scaling algorithm ISTP can be solved in $O(n^2 m \log(nC))$ time, where C denotes the largest cost in the data. The key step of the algorithm is the solution of an assignment problem with a special structure. Our method for the solution of the generic inverse matroid problem starts by solving problem DI$'$ of Section 3 with the greedy algorithm of Section 2.1, which runs in $O(mn + n^3 + m\varphi)$ time. For a graphic matroid the value φ of the computational complexity of a procedure which determines a fundamental circuit, is bounded by n and m is bounded by n^2, therefore our greedy algorithm takes $O(n^3)$. During the execution of the greedy we compute the dual values with no additional cost (Property 5). Since the number of positive dual values is bounded by $O(n)$, steps iii and iv can certainly be performed in $O(mn)$, hence our algorithm runs in $O(n^3)$, as the basic algorithm of Ahuja et al. [2].

4. Conclusions

In this paper we have presented the base-matroid defined starting from a matroid and one of its bases. After some general properties, we show that the base-matroid is actually a transversal matroid; we devise a non-trivial efficient greedy algorithm to compute the optimal base of the corresponding base-matroid optimization problem. One of the applications of the base-matroid is in the field of inverse matroid optimization.

For this reason we discuss in detail the LP formulation of the base-matroid and we propose an efficient algorithm for computing the primal and dual solutions.

It is interesting to note that, being a matroid, the definition of the base-matroid can be iterated. That is we can define the base-matroid of a base-matroid and so on. Provided that the bases used to define the sequence of base-matroids are different from the optimal base of the previous base-matroid, the process can be iterated n times. Each time the number of constraints defining the matroid decreases. At the last iteration the only constraint defining the matroid is $|S| \leqslant n$, that is the uniform matroid.

Acknowledgements

The author are gratefully indebted with the anonimous referees for their helpfull comments and suggestions on how to improve and simplify the presentation. This work was partially supported by CNR and MIUR.

References

[1] R. Ahuja, J. Orlin, Inverse optimization, Oper. Res. 49 (2001) 771–783.
[2] R. Ahuja, J. Orlin, P.T. Sokkalingam, Solving inverse spanning tree problems through network flow techniques, Oper. Res. 47 (1999) 291–298.
[3] D. Burton, L. Toint, On an instance of the inverse shortest path problem, Math. Programming 53 (1992) 45–61.
[4] M. Cai, Inverse problems of matroid intersection, J. Combin. Optim. 3 (1999) 465–474.
[5] M. Cai, Y. Li, Inverse matroid intersection problem, ZOR-Math. Methods Oper. Res. 45 (1997) 235–243.
[6] M. Cai, X. Yang, Y. Li, Inverse polymatroid flow problem, J. Combin. Theory 3 (1999) 125–126.
[7] W.H. Cunningham, On submodular function minimization, Combinatorica 5 (1985) 185–192.
[8] J. Edmons, D.R. Fulkerson, Transversals and matroid partition, J. Res. Nat. Bur. Standards Sect. B 69B (1965) 147–153.
[9] A. Frank, A weighted matroid intersection algorithm, J. Algorithms 2 (1981) 328–336.
[10] X. Hu, Z. Liu, A strongly polynomial algorithm for the inverse shortest arborescence problem, Discrete Appl. Math. 82 (1998) 135–154.
[11] E.L. Lawler, Combinatorial Optimization: Networks and Matroids, Holt, Reinehart and Winston, New York, 1976.
[12] T.J. Moser, Shortest paths calculation of seismic rays, Geophysics 56 (1991) 59–67.
[13] G.L. Nemhauser, L.A. Wolsey, Integer and Combinatorial Optimization, Wiley, New York, 1988.
[14] J.G. Oxley, Matroid Theory, Oxford University Press, New York, 1992.