Approximate Tradeoffs on Matroids[1]

Laurent Gourvès [2,3], Jérôme Monnot [2,3] and Lydia Tlilane [3]

Abstract. We consider problems where a solution is evaluated with a couple. Each coordinate of this couple represents an agent's utility. Due to the possible conflicts, it is unlikely that one feasible solution is optimal for both agents. Then, a natural aim is to find tradeoffs. We investigate tradeoff solutions with guarantees for the agents. The focus is on discrete problems having a matroid structure. We provide polynomial-time deterministic algorithms which achieve several guarantees and we prove that some guarantees are not possible to reach.

1 Introduction

This paper deals with the existence and computation of a solution s which is common to two agents. The interest of agent $i \in \{1, 2\}$ over the set of possible solutions is captured by a utility function u_i. When these functions are conflicting, it is unlikely that a feasible solution s, such that $u_1(s)$ and $u_2(s)$ are both nearly optimal, exists. So, one has to make a tradeoff.

A natural way to cope with several functions is to aggregate them in a weighted sum. For example, which solution s maximizes the utilitarian function $f_\lambda(s) := \lambda u_1(s) + (1-\lambda)u_2(s)$ for some $\lambda \in [0, 1]$? Unfortunately, this approach has two issues. The first issue is about computation: finding a solution which optimizes f_λ may be puzzling when u_1 and u_2, though separately solvable, require completely different algorithms. The second issue is that an optimum to f_λ may lead to unbalanced solutions. If s_i^* denotes the solution that maximizes $u_i(s_i^*)$ then it is possible that a solution s, though optimal for $f_\lambda(s)$, satisfies $u_1(s)/u_1(s_1^*) \approx 1$ and $u_2(s)/u_2(s_2^*) \approx 0$ (or conversely $u_1(s)/u_1(s_1^*) \approx 0$ and $u_2(s)/u_2(s_2^*) \approx 1$). This pathological case indicates that s can be unfair, i.e., close to optimality for one agent, and very far from optimality for the other agent.

Then we address the following questions: For which lower bounds on $u_1(s)/u_1(s_1^*)$ and $u_2(s)/u_2(s_2^*)$ a solution s is guaranteed to exists? Which algorithm can cope with a possibly different nature of the agent's utility functions and such that non trivial *a priori* lower bounds on $u_1(s)/u_1(s_1^*)$ and $u_2(s)/u_2(s_2^*)$ can be derived?

In this article, we seek for (α, β)-*approximate algorithms*, i.e., algorithms returning a solution s such that $u_1(s)/u_1(s_1^*) \geq \alpha$ and $u_2(s)/u_2(s_2^*) \geq \beta$ for every instance. Of particular interest are the vectors (α, β) which are optimal in the sense of Pareto. There are several papers that deal with (α, β)-*approximate algorithms*, including [11, 24, 19, 22, 10, 9].

The article is devoted to problems having a *matroid* structure (defined in Section 3). There is a rich literature on matroids [21, 18, 13].

They model many practical situations (e.g. schedules, forests of a graph), possess many remarkable structural properties and admit elegant polynomial-time algorithms. These are motivations for studying tradeoffs on matroids.

Throughout the paper, we assume that the utility of the first agent is additive while the utility of the second agent is more complex; it is a particular submodular function which captures more elaborate preferences.

A typical example of the instances covered in this article is the following:

Example 1 *A group of tourists is visiting Montpellier for n days and they ask a local travel agency to arrange the stay with (at most) one activity per day. There are m activities available: $a_1, \cdots, a_m$. Each a_i has*

- *a label $\mathcal{L}(a_i)$ which characterizes the activity (e.g. visiting a monument, attending a show, eating at a gastronomic restaurant, etc); there are typically several activities with the same label. The labels constitute a partition of the activity set,*
- *a list of days $T(a_i)$ during which a_i can be scheduled,*
- *a non negative weight $w(a_i)$ indicating what the travel agency earns when the activity is scheduled.*

Each label ℓ has a non negative gain*, denoted by $g(\ell)$, which captures a tourist's interest for any activity a_i such that $\mathcal{L}(a_i) = \ell$.*

If A denotes the subset of activities actually scheduled, then the travel agency would like to maximize its profit $\sum_{a \in A} w(a)$. On the other hand, the tourists's viewpoint is different since they want $\sum_{\ell \in \{\mathcal{L}(a):a \in A\}} g(\ell)$ to be maximized. Here, we assume that a tourist's utility increases when he does an activity of a new kind (no activity with the same label was done before).

The tourist/travel agency problem has a matroid structure usually called *tranversal* and is explained below.

2 Related work and contribution

In this article, we study bicriteria approximation of labeled matroid where the utility of the two agents are in conflict. Hence, it is unlikely that the best strategy for the first agent is also the best one for the second agent, as illustrated by the tourist/travel-agency problem. One way of tackling this problem is to approximate the Pareto set ie., the set of non-dominated solutions (any improvement on one objective induces a deterioration on another objective). This point of view has been studied in the literature in several papers [22, 20, 19, 3]. When we produce a unique solution to approximate the problem, this approach is similar in the spirit to the notion of max-min fairness [2, 4, 12, 14, 10]. Fairness has been initially considered in economics and social choice theory where fairness notions based

on some axiomatic characterizations such as proportional fairness, envy-freeness and max-min fairness [5, 16, 15, 14, 12, 17]. The goal of the max-min fairness criterion is to maximize the satisfaction of the least satisfied agent where the individual utilities of each agent is normalized in order to lie on the same scale [4, 12]. An (α, β)-approximation gives the satisfaction of the two agents so, the satisfaction of the least satisfied agent is $\min\{\alpha, \beta\}$.

In this article, we first recall definitions on matroids (Section 3). Section 4 presents the model studied in this paper and Section 5 gives the main definitions on multicriteria approximation.

Then we extend a result of [9] which deals with two agents having additive utility functions and willing to build a common spanning tree (a particular matroid problem). In Section 6, we propose a $(1/2, 1/4)$-approximation algorithm which simulates a natural process where two agents build a common solution. The general computational complexity is mentioned in Section 6 and followed by an algorithm finding a particular lexicographic optimum. In Section 8, we study a particular case, called uniform case, and show the following results: on the one hand, we produce a $(\frac{k-1}{k}, \frac{1}{k})$-approximation within polynomial-time in the uniform case, for any positive integer k given as the input, and on the other hand, we exhibit some instances without any $(\alpha, 1 - \alpha)$-approximation in the uniform case, for any $\alpha \in (0; 1)$ with $\alpha \notin \{\frac{k-1}{k} : k$ is a positive integer $\}$. Some open questions are indicated in Section 9.

3 Matroids

Matroids play an important role in combinatorial optimization and graph theory. We briefly mention some basic definitions, properties and algorithms on matroids and refer the reader to [21, 13, 18] for deeper expositions.

A matroid $\mathcal{M} = (X, \mathcal{F})$ consists of a finite set of n elements X and a collection $\mathcal{F}$ of subsets of X such that:

(i) $\emptyset \in \mathcal{F}$,
(ii) if $F_2 \subseteq F_1$ and $F_1 \in \mathcal{F}$ then $F_2 \in \mathcal{F}$,
(iii) for every couple F_1, $F_2 \in \mathcal{F}$ such that $|F_1| < |F_2|, \exists\, x \in F_2 \backslash F_1$ such that $F_1 \cup \{x\} \in \mathcal{F}$.

By induction (iii) is equivalent to

(iii') for every couple F_1, $F_2 \in \mathcal{F}$ where $|F_1| < |F_2|, \exists A \subseteq F_2 \backslash F_1$ with $|A| = |F_2| - |F_1|$ such that $F_1 \cup A \in \mathcal{F}$.

The elements of $\mathcal{F}$ are called *independent*, the element of $2^X \setminus \mathcal{F}$ *dependent*. Inclusion wise minimal dependent sets are called *circuits* and inclusionwise maximal independent sets are called *bases*. All bases of a matroid M have the same cardinality $r(M)$, defined as the *rank* of M. Given a matroid $\mathcal{M} = (X, \mathcal{F})$ and a subset $X' \subset X$, the *restriction* of $\mathcal{M}$ to X', denoted by $\mathcal{M}|X'$, is the structure $(X', \mathcal{F}')$ where $\mathcal{F}' = \{F \in \mathcal{F} : F \subseteq X'\}$. If $X' \in \mathcal{F}$, the *contraction*[4] of $\mathcal{M}$ by X', denoted $\mathcal{M}/X'$, is the structure $(X \setminus X', \mathcal{F}')$ where $\mathcal{F}' = \{F \subseteq X \setminus X' : F \cup X' \in \mathcal{F} :\}$. It is well known that $\mathcal{M}|X'$ and $\mathcal{M}/X'$ are matroids.

Typical examples of matroids are the following:

- The forests (set of edges which do not admit a cycle) of a multi-graph G form a matroid usually called the *graphic matroid* of G. A base in this matroid is a spanning tree.

---

[4] Actually, the contraction is defined in the literature for any $X' \subset X$, and when $X' \in \mathcal{F}$ the definition is similar to the one given in this paper.

- Given k disjoint sets $E_1, \ldots, E_k$ which form a ground set $E = \cup_{i=1}^{k} E_i$ and k non negative integers b_i $(i = 1..k)$, the sets $F \subseteq E$ satisfying $|F \cap E_i| \leq b_i$ form a matroid usually called the *partition matroid*.
- Given k (not necessarily disjoint) sets $E_1, \ldots, E_k$, subsets of a ground set E, a *partial transversal* is a set $T \subseteq E$ such that an injective map $\Phi : T \to [1..k]$ satisfying $t \in E_{\Phi(t)}$ exists. Then $(E, \mathcal{T})$ where $\mathcal{T} = \{T \in 2^E : T$ is a partial transversal of $E\}$ is a matroid usually called the *transversal matroid*.

Returning to Example 1, let $E = \{a_1, a_2, ..., a_m\}$ a set of activities and $E_i \subseteq E$ the subset of activities available on day i, $i = 1..n$. A set of activities $T \subseteq E$ is feasible if it exists an injective mapping $\Phi : T \to [1..n]$ combining at most one activity of E_i per day i, $i = 1..n$.

A matroid is said *simple* if no single element, or pair of elements, is a circuit [18]. For example, the forests of a simple graph define a simple matroid.

When every element $e \in X$ has a weight $w(e) \in \mathbb{R}^+$, a typical optimization problem consists of computing a base $B \in \mathcal{F}$ that maximizes $\sum_{e \in B} w(e)$. This problem is solved by the following algorithm:

Algorithm 1 GREEDY

Require: $\mathcal{M} = (X, \mathcal{F}), w : X \to \mathbb{R}^+$
1: Sort $X = \{e_1, \cdots, e_n\}$ such that $w(e_i) \geq w(e_{i+1})$,
   $i = 1..n - 1$
2: Set $F = \emptyset$
3: **for** $i = 1$ to n **do**
4:    **if** $F \cup \{e_i\} \in \mathcal{F}$ **then**
5:       $F \leftarrow F \cup \{e_i\}$
6:    **end if**
7: **end for**
8: **return** F

Note that the execution of GREEDY on a forest matroid coincides with Kruskal's algorithm for maximum weight spanning trees.

We always assume that an *independence oracle* can decide within polynomial time whether a set F is independent or dependent. Given two matroids $(X, \mathcal{F}_1)$ and $(X, \mathcal{F}_2)$ defined over the same set of elements X, there are algorithms (more elaborate than GREEDY) to solve the following problems in polynomial time [13, 21]:

- find an independent $F \in \mathcal{F}_1 \cap \mathcal{F}_2$ of maximum cardinality.
- when every element $e \in X$ has a weight $w(e) \in \mathbb{R}^+$, find an independent $F \in \mathcal{F}_1 \cap \mathcal{F}_2$ that maximizes $\sum_{e \in F} w(e)$.

4 The model: two agents on a matroid

Let $\mathcal{M} = (X, \mathcal{F})$ be a matroid and consider the following functions:

- $w : X \to \mathbb{R}^+$, where $w(x)$ is called the *weight* of $x \in X$,
- $\mathcal{L} : X \to \{\ell_1, \ldots, \ell_p\}$, where $\{\ell_1, \ldots, \ell_p\}$ is the set of labels and $\mathcal{L}(e)$ is called the label of e,
- $g : \mathcal{L} \to \mathbb{R}^+$, where $g(\ell)$ is the *gain* of label ℓ.

Note that w is additive while g has a more general form (it is a particular submodular function).

For the ease of presentation we often write $g(x)$ instead of $g(\mathcal{L}(x))$ for $x \in X$. The labels of a set $X' \subseteq X$ is a set denoted by $\mathcal{L}(X')$ and defined as $\bigcup_{x \in X'}\{\mathcal{L}(x)\}$. In the tourist/travel-agency problem, the labels $\{\ell_1, \ldots, \ell_p\}$ are the activity types.

We study a model where $\mathcal{F}$ is the set of feasible solutions. The first and second agent's utilities are w and g, respectively. A set $F \in \mathcal{F}$ is then evaluated with two objective functions: $w(F) = \sum_{x \in F} w(x)$ and $g(F) = \sum_{l \in \mathcal{L}(F)} g(l)$ which should be maximized. An instance is then a tuple $\langle \mathcal{M}, w, \mathcal{L}, g \rangle$.

As previously mentioned, GREEDY finds a base F_1 that maximizes $w(F_1)$. Finding F_2 that maximizes $g(F_2)$ can be done in polynomial time by searching a maximum cost intersection of two matroids. The first matroid is $\mathcal{M} = (X, \mathcal{F})$. The second matroid $\mathcal{M}' = (X, \mathcal{F}')$ is a partition matroid defined by the labels of X: $\mathcal{F}' := \{S \subset X : |\mathcal{L}(S)| = |S|\}$. Every element $x \in X$ has a cost $c(x)$ defined as $g(\mathcal{L}(x))$. Use any appropriate algorithm to compute a maximum cost independent in $\mathcal{F} \cap \mathcal{F}'$ (see [13, 21]) and complete it into a base F_2 if necessary. It is not difficult to show that the resulting base maximizes g.

5 Non trivial approximation

When focusing on a particular agent i, we say that an algorithm $\mathcal{A}$ is ρ-approximate if, for every instance, $\mathcal{A}$ returns a solution s satisfying $u_i(s)/u_i(s_i^*) \geq \rho$. Here, s_i^* is a solution which maximizes $u_i(s_i^*)$ and $\rho \in [0, 1]$ is called the *approximation ratio* or *performance guarantee* [23].

When dealing with $k \geq 2$ agents, we say that an algorithm $\mathcal{A}$ is $(\rho_1, \ldots, \rho_k)$-approximate if, for every instance, $\mathcal{A}$ returns a solution s satisfying $u_i(s)/u_i(s_i^*) \geq \rho_i$ for $i = 1..k$. Actually the vector $(u_1(s_1^*), \ldots, u_k(s_k^*))$ is often called the *ideal point* [8] since it is the image of an unlikely feasible solution where optimality is reached for all agents. In this paper, we propose algorithms which approximate this point.

Obviously, returning a solution that maximizes w (resp. g) gives a $(1, 0)$-approximation (resp. $(0, 1)$-approximation) but we expect α and β to be positive so that the solution constitutes a non trivial tradeoff. Next examples show that, for the general model considered in this article (described in Section 4), there is no hope for a non trivial (α, β)-approximate tradeoff (such that $\alpha > 0$ and $\beta > 0$) if we consider the whole class of matroids.

Example 2 *Consider the matroid* $(X = \{a, b\}, \mathcal{F} = \{\emptyset, \{a\}, \{b\}\}$, *where* $w(a) = 1$, $w(b) = 0$, $\ell(a) = l$, $\ell(b) = l'$, $g(l) = 0$ *and* $g(l') = 1$. *The rank is 1.*

Example 3 *Consider the matroid* $(X = \{a, b, c\}, \mathcal{F} = \{\emptyset, \{a\}, \{b\}, \{c\}, \{a, c\}, \{b, c\}\}$, *where* $w(a) = 1$, $w(b) = w(c) = 0$, $\ell(a) = \ell(c) = l$, $\ell(b) = l'$, $g(l) = 0$ *and* $g(l') = 1$.

In both examples, for every independent F, $\min\{w(F), g(F)\} = 0$ while $\max_{F \in \mathcal{F}}\{w(F)\} = \max_{F \in \mathcal{F}}\{g(F)\} = 1$. This means that either $\alpha = 0$ or $\beta = 0$ for every feasible solution. However, one can overcome this issue by considering a notion which generalizes the notion of simple matroid.

Definition 1 *A matroid* $\mathcal{M} = (X, \mathcal{F})$ *is said labeled-simple if* $\mathcal{L}(x_1) = \mathcal{L}(x_2)$ *for every circuit* $C = \{x_1, x_2\}$ *of size two.*

Note that the matroids of Examples 2 and 3 are not labeled-simple. More generally, a simple matroid is labeled-simple and when every label appears once, these two notions coincide.

As shown in the next section, excluding matroids which are not labeled-simple makes the existence of non trivial approximation possible.

6 A general greedy algorithm

We propose to analyze a simple extension of GREEDY which builds a tradeoff solution. This extension, called ALT-GREEDY, simulates a simple and natural process for the construction of a tradeoff.

At the beginning, $F = \emptyset$ and the agents alternatively add an element e to F such that $F + e \in \mathcal{F}$ until F becomes a base. If it is the first agent's turn, then we assume that he selects e that maximizes $w(F + e)$. In a symmetric way, the second agent chooses e that maximizes $g(F + e)$ during his turn.

We suppose that the first agent (the one who tries to maximize $w(F)$) plays first.

Theorem 1 ALT-GREEDY *is* $(1/2, 1/4)$-*approximate for labeled-simple matroids.*

Proof. Let $B = \{e_1, \ldots, e_r\}$ be the base returned by ALT-GREEDY. Let us first focus on the ratio $1/2$ for the first agent. Let $B^* = \{e_1^*, \ldots, e_r^*\}$ be a base with maximum weight satisfying $w(e_1^*) \geq \ldots \geq w(e_r^*)$. Each element of B with odd index is inserted by the first agent who wants to maximize the total weight. Let i odd. When $i = 1$ the current solution F is $\emptyset$ (just before e_i is inserted). Otherwise $F = \{e_1, \ldots, e_{i-1}\}$. Set $F' = \{e_1^*\}$ if $i = 1$, otherwise $F' = \{e_1^*, \ldots, e_i^*\}$. By property (iii) of a matroid, and because $|F'| > |F|$, there exists an element $\hat{e} \in F' \setminus F$ such that $F + \hat{e} \in \mathcal{F}$. Using $w(e_1^*) \geq \ldots \geq w(e_r^*)$, we know that $w(\hat{e}) \geq w(e_i^*)$. Since the first agent selects the element e_i that maximizes $w(F + e_i)$ and because this agent is additive, we deduce that $w(e_i) \geq w(\hat{e}) \geq w(e_i^*)$, for every odd i. We get that

$$w(\bigcup_{i=1}^{r} e_i) \geq w(\bigcup_{i=1, \text{ odd}}^{r} e_i) \geq w(\bigcup_{i=1, \text{ odd}}^{r} e_i^*), \qquad (1)$$

where the non negativity of an element's weight is used. Now observe that $w(e_i^*) \geq \frac{1}{2}(w(e_i^*) + w(e_{i+1}^*))$ because $w(e_i^*) \geq w(e_{i+1}^*)$. It follows that

$$w(B) = w(\bigcup_{i=1}^{r} e_i) \geq \frac{1}{2}w(\bigcup_{i=1}^{r} e_i^*) = w(B^*)/2.$$

Thus, ALT-GREEDY is $(1/2, \cdot)$-approximate. Now, consider the second agent. Let B^g be a base with maximum gain. Let E be the elements of B^g whose label does not appear in the solution returned by ALT-GREEDY: $E := \{e \in B^g : \mathcal{L}(e) \notin \mathcal{L}(B)\}$. We suppose that the first $\nu = |\mathcal{L}(E)|$ elements of E have distinct labels and they are sorted by non increasing gain: $g(e_1^g) \geq g(e_2^g) \geq \ldots \geq g(e_\nu^g)$ where $E = \{e_1^g, \ldots, e_\nu^g, \ldots, e_{|E|}^g\}$. E is in $\mathcal{F}$ because $E \subseteq B^g$ and $B^g \in \mathcal{F}$. Note that $g(E) = g(\{e_1^g, \ldots, e_\nu^g\}) = \sum_{i=1}^{\nu} g(e_i^g)$.

Suppose that during an even step of ALT-GREEDY, the second agent can not add an element with a new label to the current solution F. We know that $F \in \mathcal{F}$ because $F \subseteq B$. If $|F| < |E|$ then, by property (iii) of matroids, one can add an element with a new label to F, contradiction. We deduce that $|F| \geq |E|$. Hence F has at least ν elements. Let $F' = \{e_1, \ldots, e_\nu\}$ be the first ν elements of F (following the order by which they are inserted during the algorithm). Every element with an even index, within F', was inserted by the second agent. By property (iii) of a matroid, we know that $g(e_i) \geq g(e_i^g)$ holds for every even i between 2 and ν. It follows that $\sum_{i=2, \text{ even}}^{\nu} g(e_i) \geq \sum_{i=2, \text{ even}}^{\nu} g(e_i^g)$. The first ν elements of E being sorted by non increasing gain, we know that $2\sum_{i=2, \text{ even}}^{\nu} g(e_i^g) \geq (\sum_{i=1}^{\nu} g(e_i^g)) - g(e_1^g)$ where $\sum_{i=1}^{\nu} g(e_i^g) =$

$g(E)$. We deduce that

$$2g(F') \geq 2\sum_{i=2,\,even}^{\nu} g(e_i) \geq \sum_{i=1}^{\nu} g(e_i^g) - g(e_1^g) = g(E) - g(e_1^g). \quad (2)$$

If $\mathcal{L}(e_1) = \mathcal{L}(e_1^g)$ then $g(B) \geq g(e_1^g)$, otherwise $\{e_1, e_1^g\} \in \mathcal{F}$ because the matroid is labeled-simple (see Definition 1) and $g(e_2) \geq g(e_1^g)$ by property (iii) [5], In any case, we get that

$$g(B) \geq g(e_1^g). \quad (3)$$

Use $F' \subseteq B$ and Inequalities (2) and (3) to derive

$$3g(B) \geq 2g(F') + g(B) \geq g(E). \quad (4)$$

By definition, $\mathcal{L}(B^g) \setminus \mathcal{L}(E)$ are the labels appearing in B. It follows that $g(B) \geq g(B^g) - g(E)$ that we add to Inequality (4) to get that

$$4g(B) \geq g(B^g).$$

Thus, ALT-GREEDY is $(\cdot, 1/4)$-approximate if, during an even step of ALT-GREEDY, the second agent could not add an element with a new label to the current solution F. Now, suppose that for every turn of the second agent, it was possible to add an element with a new label. Rename the $\xi = |\mathcal{L}(B^g)|$ first elements of B^g such that $\mathcal{L}(\{e_1^g, \ldots, e_\xi^g\}) = \mathcal{L}(B^g)$ and $g(e_1^g) \geq g(e_2^g) \geq \ldots \geq g(e_\xi^g)$. Note that $\sum_{i=1}^{\xi} g(e_i^g) = g(B^g)$. By property (iii) we have $g(e_i) \geq g(e_i^g)$ for $i \leq \xi$ and even. We deduce that $g(B) \geq \sum_{i=2,\,even}^{\xi} g(e_i) \geq \sum_{i=2,\,even}^{\xi} g(e_i^g)$. Since the first ξ elements of B^g are sorted by non increasing gain, we know that $2\sum_{i=2,\,even}^{\xi} g(e_i^g) \geq \left(\sum_{i=1}^{\xi} g(e_i^g)\right) - g(e_g^1)$. Hence $2g(B) \geq \left(\sum_{i=1}^{\xi} g(e_i^g)\right) - g(e_g^1) = g(B^g) - g(e_g^1)$. Since inequality (3) holds, we deduce that $3g(B) \geq g(B^g)$ which is better than $4g(B) \geq g(B^g)$.

In conclusion, ALT-GREEDY is $(1/2, 1/4)$-approximate. $\square$

Note that if we apply ALT-GREEDY by inverting the role of the agents (even steps are for the weight and odd steps are for the labels), then we can prove that ALT-GREEDY is $(1/3, 1/3)$-approximate for labeled-simple matroids.

Next example shows that the analysis of ALT-GREEDY is tight.

Example 4 *Consider the graphic matroid of the following graph.*
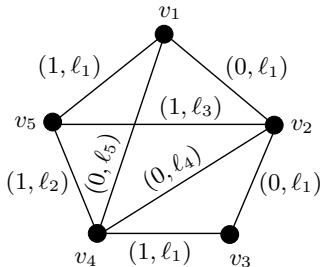
Each edge e has a pair $(w(e), \mathcal{L}(e))$. We suppose that $g(\ell_2) = 0$ and $g(\ell_i) = 1$ for any $i \neq 2$. The tree $\{(v_1, v_5), (v_2, v_5), (v_3, v_4), (v_4, v_5)\}$ has weight 4 while the tree $\{(v_1, v_4), (v_2, v_3), (v_2, v_4), (v_2, v_5)\}$ uses the labels $\{\ell_1, \ell_3, \ell_4, \ell_5\}$ so its gain is 4. ALT-GREEDY may output a tree containing (v_4, v_5), (v_1, v_2), (v_1, v_5) and (v_2, v_3) (edges are listed by the order they enter the solution). This tree has weight 2 and its gain is 1.

---

[5] Note that the rank is at least 2 by assumption.

7 General computational complexity and a particular solution

Theorem 1 is a constructive proof that every instance of the model admits a $(1/2, 1/4)$-approximate solution. More generally, given an instance $\langle (X, \mathcal{F}), w, \mathcal{L}, g \rangle$ and two bounds k_w and k_g, what is the computational complexity of the following decision problem?

Π: *Is there any $F \in \mathcal{F}$ such that $w(F) \geq k_w$ and $g(F) \geq k_g$?*

Π generalizes the minimal spanning tree problem with a side constraint which was shown **NP**-complete [1], see also [7] for matroids. Actually, by considering the graphic matroid where each edge e has a distinct label e, the problem dealt with in the paper is exactly the minimal spanning tree problem with a side constraint.

Though **NP**-complete in general, next result states that a particular Pareto optimal solution can be computed in polynomial time. Within the set of optimal solutions for the weight, let $F_{w,g}^*$ be the one which maximizes g. $F_{w,g}^*$ is a lexicographic Pareto optimal solution.

Theorem 2 *$F_{w,g}^*$ can be computed in polynomial time.*

Before giving a proof of Theorem 2, we give an intermediate result.

Lemma 1 *Let $\mathcal{M} = (X, \mathcal{F})$ be a matroid, $\mathcal{B}$ its set of bases and $w : X \to \mathbb{R}^+$ a weight function. Let $\mathcal{B}_w \subseteq \mathcal{B}$ be the set of all bases which are optimal for w. Then $\mathcal{M}_w = (X, \mathcal{F}_w)$ where $\mathcal{F}_w := \{F \subseteq B : B \in \mathcal{B}_w\}$ is a matroid.*

Due to space limitation, the proof of Lemma 1 is skipped. The proof relies on the next property which follows from results of [6, 21].

Property 1 *Let $B_1, B_2 \in \mathcal{B}_w$ with $B_1 \neq B_2$. Then, $\forall e_1 \in B_1 \setminus B_2$, $\exists e_2 \in B_2 \setminus B_1$ such that $(B_1 \setminus \{e_1\}) \cup \{e_2\}$ and $(B_2 \setminus \{e_2\}) \cup \{e_1\}$ are in $\mathcal{B}_w$.*

Now, we are ready to give a proof of Theorem 2.

Proof. Start with $\mathcal{M} = (X, \mathcal{F})$ and define $\mathcal{M}_w = (X, \mathcal{F}_w)$ as in Lemma 1. Let $\mathcal{M}' = (X, \mathcal{F}')$ be a partition matroid where $F \in \mathcal{F}'$ iff $F \in \mathcal{F}_w$ and $|\mathcal{L}(F)| = |F|$. Find $\hat{F} \in \mathcal{F}_w \cap \mathcal{F}'$ with maximizes $g(\hat{F})$. Since $\mathcal{M}_w$ and $\mathcal{M}'$ are two matroids (cf Lemma 1), this can be done in polynomial time [13, 21]. Complete $\hat{F}$ into a base $\hat{B}$ in a greedy manner with elements of X sorted by non increasing weight (like in GREEDY). We claim that $\hat{B} = F_{w,g}^*$.

$\hat{F} \in \mathcal{F}_w$ so its completion leads to a base of maximum weight: $\hat{B} \in \mathcal{B}_w$. Observe that $g(\hat{F}) = g(\hat{B})$ since otherwise $\hat{F}$ does not maximize $g(\hat{F})$. Now suppose, by contradiction, that a base $\tilde{B} \in \mathcal{B}_w$ is such that $g(\tilde{B}) > g(\hat{F})$. Retain exactly one element per label of $\tilde{B}$ to get an independent $\tilde{F}$. Since $\tilde{F} \subseteq \tilde{B}$ we have $\tilde{F} \in \mathcal{F}_w$. Moreover, $g(\tilde{F}) = g(\tilde{B})$ because every label of $\tilde{B}$ appears in $\tilde{F}$. It follows that $g(\tilde{F}) > g(\hat{F})$, contradicting the optimality of $\hat{F}$. $\square$

Note that the complexity of the algorithm that finds $F_{w,g}^*$ depends on Edmonds' Matroid Intersection Algorithm runs in $O(|X|^4 + Y * |X|^3)$ time where Y is the complexity of the independence oracle [13]. Y is not given explicitly, it depends on the matroid under consideration. In our study, we suppose that Y is a polynomial (see Section 3).

8 The uniform subcase

In this section, we consider a particular case, called *uniform*, where $g(l) = 1$ for all $l \in \mathcal{L}$. In fact, $g(F) = |\mathcal{L}(F)|$ holds in this case

so the second agent's goal is to maximize the number of distinct labels. As previously mentioned, finding a base $B^{\mathcal{L}}$ that maximizes $|\mathcal{L}(B^{\mathcal{L}}|$ can be done in polynomial time [13, 21]. In the following, L denotes $|\mathcal{L}(B^{\mathcal{L}})|$. In this section, we do not assume that the matroid is labeled-simple.

One can analyze ALT-GREEDY and show that it is $(1/2, 1/3 + 1/r(X))$-approximate for any matroid $\mathcal{M} = (X, \mathcal{M})$ in the uniform subcase. It is noteworthy that we do not need to restrict ourselves to labeled-simple matroids anymore. The proof is skipped because it follows the line of Theorem 1's proof and we are able to propose another algorithm, called 3-PHASES, with better guarantees.

Indeed, we will prove that 3-PHASES is $(\frac{k-1}{k}, \frac{1}{k})$-approximate for every positive integer k taken as the input. Therefore, $k = 2$ gives a $(1/2, 1/2)$-approximation which improves the guarantees of ALT-GREEDY.

3-PHASES starts with an empty solution F and agent 1 adds $\lfloor \frac{k-1}{k} L \rfloor$ elements in a greedy manner (trying to maximize the weight). Afterwards, agent 2 adds to F a set of at most $\lfloor \frac{L}{k} \rfloor$ new elements so that the number of new labels (ie. labels not present in the first phase) is maximized. During the last phase, agent 1 completes F in a greedy way by adding elements according to their weight. In the following, B_j denotes the elements inserted during phase j. Hence, 3-PHASES returns the independent set $B_1 \cup B_2 \cup B_3$.

The first and third phase of the algorithm are both greedy and clearly polynomial. Only the second phase of the algorithm is not greedy. It consists of finding an independent of limited cardinality (at most $\lfloor L/k \rfloor$ elements) in the intersection of two matroids.

The first matroid is $\mathcal{M}' = (X', \mathcal{F}')$ where $X' := X \setminus \{x \in X : \mathcal{L}(x) \notin \mathcal{L}(B_1)\}$ and $\mathcal{F}' := \{F \subset X' : B_1 \cup F \in \mathcal{F}\}$. Actually, $\mathcal{M}'$ can be obtained by applying first the contraction of $\mathcal{M}$ by B_1, $\mathcal{M}/B_1$ and then, the restriction of $\mathcal{M}/B_1$ to $X' = X \setminus \{x \in X : \mathcal{L}(x) \notin \mathcal{L}(B_1)\}$ (note that $X' \subseteq X \setminus B_1$). Because, the contraction and the restriction of matroids are matroids, $\mathcal{M}'$ is a matroid. The second matroid $\mathcal{M}_2 = (X', \mathcal{F}_2)$ is the partition matroid of X' induced by the labels, ie., assume that $\mathcal{L}(X) = \{\ell_1, \ldots, \ell_p\}$ and if $X_i = \{x \in X : \mathcal{L}(x) = \ell_i\}$ for $i = 1..p$ denotes the elements with label ℓ_i, then $\mathcal{F}_2 = \{F \subseteq X' : \forall i = 1..p, |X_i \cap F| \leq 1\}$.

Find a set S of maximum cardinality in $\mathcal{F} \cap \mathcal{F}'$ (this can be done in polynomial time [13, 21]) and if $|S| > \lfloor L/k \rfloor$, retain a subset of only $\lfloor L/k \rfloor$ elements. By property (ii) of a matroid, the resulting set is independent. Thus, the second phase of 3-PHASES runs in polynomial time.

Theorem 3 *Let k be a positive integer taken as the input,* 3-PHASES *is $(\frac{k-1}{k}, \frac{1}{k})$-approximate in the uniform case.*

Proof. The algorithm is clearly $(0, 1)$-approximate when $k = 1$ because no element is picked during the first phase and during the second phase one can insert $\lfloor L \rfloor = L$ elements with distinct labels.

From now on, we suppose that $k \geq 2$. Let $B = \{e_1, \ldots, e_r\}$ be the base returned by 3-PHASES. The elements of B are numbered to the order which they are inserted in the solution. Let us first focus on the ratio $\frac{k-1}{k}$ for the first agent. Let $B^* = \{e_1^*, \ldots, e_r^*\}$ be a base with maximum weight satisfying $w(e_1^*) \geq \ldots \geq w(e_r^*)$. By the third property of a matroid, we know that $w(e_i) \geq w(e_i^*)$ for $i = 1..|B_1|$ and $|B_1| = \lfloor \frac{k-1}{k} L \rfloor$. The elements of B^* being sorted by non increasing weight, we also have

$$w(e_i) \geq w(e_j^*) \tag{5}$$

for any pair $(i, j) \in \{1, \ldots, |B_1|\} \times \{|B_1| + 1, \ldots, |B_1| + |B_2|\}$. Note that B_2 is the set of elements added during the second phase

by the second agent and $|B_2| \leq \lfloor L/k \rfloor$. Since $|B_1| = \lfloor \frac{k-1}{k} L \rfloor \geq (k-1) \lfloor \frac{L}{k} \rfloor \geq (k-1)|B_2|$, one can split B_1 in $k - 1$ disjoints sets $B_1^1, \ldots, B_1^{k-1}$ so that every one contains at least $|B_2|$ elements. Using Inequality (5) we get that

$$w(B_1^p) \geq \sum_{j=|B_1|+1}^{|B_1|+|B_2|} w(e_j^*) \text{ for } p = 1..k - 1.$$

Summing up these inequalities gives

$$w(B_1) = \sum_{p=1}^{k-1} w(B_1^p) \geq (k-1) \sum_{j=|B_1|+1}^{|B_1|+|B_2|} w(e_j^*).$$

Thus,

$$\frac{1}{k-1} w(B_1) \geq \sum_{j=|B_1|+1}^{|B_1|+|B_2|} w(e_j^*). \tag{6}$$

Using again (5) gives

$$w(B_1) \geq \sum_{j=1}^{|B_1|} w(e_j^*). \tag{7}$$

Adding (6) and (7), we get that

$$\frac{k}{k-1} w(B_1) \geq \sum_{j=+1}^{|B_1|+|B_2|} w(e_j^*). \tag{8}$$

Concerning the third phase, Property (iii) implies $w(e_i) \geq w(e_i^*)$ for $r \geq i \geq |B_1| + |B_2| + 1$. Summing up these inequalities gives $w(B_3) = \sum_{i=|B_1|+|B_2|+1}^{r} w(e_i) \geq \sum_{i=|B_1|+|B_2|+1}^{r} w(e_i^*)$ that we plug in the previous inequality to get that

$$\frac{k}{k-1} w(B_1) + w(B_3) \geq \sum_{i=1}^{|B_1|+|B_2|} w(e_i^*) + \sum_{i=|B_1|+|B_2|+1}^{r} w(e_i^*),$$

$$\frac{k}{k-1} w(B) \geq w(B^*).$$

Then, 3-PHASES is $(\frac{k-1}{k}, \cdot)$-approximate. Now, consider the second agent and let $B^{\mathcal{L}}$ be a base of $\mathcal{M}$ with L labels (the maximum number). Let $E^{\mathcal{L}}$ be a subset of $B^{\mathcal{L}}$ containing one element per label in $B^{\mathcal{L}}$, ie. $|E^{\mathcal{L}}| = |\mathcal{L}(E^{\mathcal{L}})| = |\mathcal{L}(B^{\mathcal{L}})| = L$. The second phase of the algorithm consists of adding to B_1, at most $\lfloor L/k \rfloor$ elements with labels which do not appear in $\mathcal{L}(B_1)$. Let $E = \{e \in E^{\mathcal{L}} : \mathcal{L}(e) \notin \mathcal{L}(B_1)\}$. Denote by p the number of labels that B_1 and $E^{\mathcal{L}}$ share. We have $|\mathcal{L}(B_1)| \geq p$ and $|E| = L - p$.

Suppose $|E| \leq |B_1|$. We get that $L - p = |E| \leq |B_1| = \lfloor L/k \rfloor$. Then $p \geq L - \lfloor L/k \rfloor = \lceil L/k \rceil \geq L/k$. Use $|\mathcal{L}(B_1)| \geq p$ to observe that B_1, and a fortiori B, contains at least L/k labels.

Now, suppose that $|E| > |B_1|$. At least $|E| - |B_1| = L - p - \lfloor \frac{k-1}{k} L \rfloor = \lceil L/k \rceil - p$ elements with new labels are added during the second phase, since on the one hand, by Property (iii') $\exists A \subseteq E \setminus B_1$ with $|A| = |E| - |B_1|$ such that $A \cup B_1 \in \mathcal{F}$ (i.e., A is an independent set of $\mathcal{M}' = (X', \mathcal{F}')$ where we recall that $X' = X \setminus \{x \in X : \mathcal{L}(x) \notin \mathcal{L}(B_1)\}$ and $\mathcal{F}' := \{F \subset X' : B_1 \cup F \in \mathcal{F}\}$) and on the other hand, A is an independent set of the partition matroid $\mathcal{M}_2 = (X', \mathcal{F}_2)$ because $|\mathcal{L}(A)| = |A|$.

Thus, A is a feasible solution of the intersection of matroids $\mathcal{M}'$ and $\mathcal{M}_2$. So, $|\mathcal{L}(B_2)| \geq |\mathcal{L}(A)|$. Now, because $|\mathcal{L}(B_1)| \geq p$, at least $|\mathcal{L}(B_1 \cup B_2)| = |\mathcal{L}(B_1)| + |\mathcal{L}(B_2)| \geq |\mathcal{L}(B_1)| + |\mathcal{L}(A)| =$

$\lceil L/k \rceil \geq L/k$ labels appear at the end of the second phase. The elements added during the last phase can only increase the number of labels so we know that in every case 3-PHASES is $(\cdot, 1/k)$-approximate.

In conclusion, 3-PHASES is $\left(\frac{k-1}{k}, \frac{1}{k}\right)$-approximate. $\square$

Next example shows that the guarantees of 3-PHASES are essentially Pareto optimal.

Lemma 2 *Let $\alpha \in (0; 1)$ and $\beta \in (0; 1)$. There are instances without any (α, β)-approximation in the uniform case in the following cases:*

(a) $\alpha + \beta \geq 1$ *and* $\alpha \neq \frac{k-1}{k}$ *for every positive integer k.*
(b) $\alpha > 0$ *and* $\beta > 1/2$.

Proof. We prove these results for a particular matroid, the graphic matroid. Let $\alpha > 0$ and $\beta > 0$.
(a) We prove that the result holds for $\alpha + \beta = 1$, and then a fortiori for $\alpha + \beta \geq 1$. So, assume $\alpha + \beta = 1$ and $\alpha \neq \frac{k-1}{k}$ for every positive integer k. Thus, there is a unique integer $L \geq 2$ such that $\frac{L-2}{L-1} < \alpha < \frac{L-1}{L}$. Therefore, we have $\alpha L > \frac{L(L-2)}{L-1}$ and $(1 - \alpha)L > 1$. Consider the following multigraph G_L, instance of the uniform case for the graphic matroid.
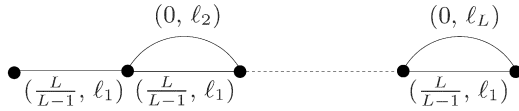
The ideal point of G_L is $x^L = (L, L)$ by considering the trees given by the bottom edges for the weight and the top edges for the labels.

By contradiction, assume that T is a spanning tree of G_L with weight $w(T) \geq \alpha L$ and number of labels $|\mathcal{L}(T)| \geq (1 - \alpha)L$. If T contains at least $L - 1$ edges of weight $L/(L - 1)$, then T contains one label. Hence, $1 = |\mathcal{L}(T)| \geq (1 - \alpha)L > 1$, contradiction. Thus, T contains at most $L - 2$ edges of weight $L/(L - 1)$ and then $L(L - 2)/(L - 1) \geq w(T) \geq \alpha L > L(L - 2)/(L - 1)$, contradiction.
(b) Consider the multigraph G_2, ie., $L = 2$ described previously. If $\beta > 1/2$, then $|\mathcal{L}(T)| \geq 2\beta > 1$. So, $|\mathcal{L}(T)| = 2$ and T is given by the top edges of G_2. In this case, $w(T) = 0$. $\square$

Lemma 2.b shows that $\beta = 1/2$ is a tradeoff on the number of labels that we can reach if some positive guarantee on the weight is achieved. Hence, Theorem 3 gives the best results that we can hope.

9   Conclusion and future directions

For the general case, we have proposed a polynomial-time $(1/2, 1/4)$-approximation for labeled-simple matroids (a $(1/3, 1/3)$-approximation exists also by inverting the agents' role). An important question consists of improving these approximations. We believe that a $(1/2, 1/3)$-approximation exists and that it offers Pareto optimal guarantees, i.e., there are small instances without any (α, β)-approximation with $\alpha \geq 1/2$ and $\beta \geq 1/3$.

For the uniform subcase, we proposed polynomial-time deterministic algorithms which achieve several tradeoffs and we prove that some tradeoffs are not possible. Note that the complexity result of Section 7 does not hold for the uniform case so the exact complexity of the uniform case is open. A first step in this direction would be to consider the open problem of computing $F_{g,w}^*$ (a Pareto optimal

solution with maximum weight among solutions of maximum gain). Another open question is the case where both agents' utility functions are of type "g" (not only the second agent's utility function).

REFERENCES

[1] V. Aggarwal, Yash P. Aneja, and K. P. K. Nair, 'Minimal spanning tree subject to a side constraint', *Computers & OR*, **9**(4), 287–296, (1982).
[2] A. Asadpour and A. Saberi, 'An approximation algorithm for max-min fair allocation of indivisible goods', in *STOC*, eds., D. S. Johnson and U. Feige, pp. 114–121. ACM, (2007).
[3] A. Berger, V. Bonifaci, F. Grandoni, and G. Schäfer, 'Budgeted matching and budgeted matroid intersection via the gasoline puzzle', *Math. Program.*, **128**(1-2), 355–372, (2011).
[4] D. Bertsimas, V.F. Farias, and N. Trichakis, 'The price of fairness', *Operations Research*, **59**(1), 17–31, (2011).
[5] S. Bouveret and J. Lang, 'Efficiency and envy-freeness in fair division of indivisible goods: Logical representation and complexity', *Journal of Artificial Intelligence Research*, **32**, 525–564, (2008).
[6] R.A. Brualdi, 'Comments on bases in different structures', *Bull. Austral. Math. Soc.*, **1**, 161–167, (1969).
[7] M. Ehrgott, 'On matroids with multiple objectives', *Optimization*, **38**, 73–84, (1996).
[8] M. Ehrgott, *Multicriteria Optimization*, Springer, 2005.
[9] B. Escoffier, L. Gourvès, and J. Monnot, 'Fair solutions for some multiagent optimization problems', *Autonomous Agents and Multi-Agent Systems*, 1–18. 10.1007/s10458-011-9188-z.
[10] A. Goel and A. Meyerson, 'Simultaneous optimization via approximate majorization for concave profits or convex costs', *Algorithmica*, **44**(4), 301–323, (2006).
[11] P. Hansen, 'Bicriterion path problems', *Proc. 3rd Conf. Multiple Criteria Decision Making Theory and Application*, 109–127, (1998).
[12] E. Kalai and M. Smorodinsky, 'Other solutions to Nash's bargaining problem', *Econometrica*, **43**, 513–518, (1975).
[13] B. Korte and J. Vygen, *Combinatorial Optimization: Theory and Algorithms*, Springer-Verlag, 2007.
[14] A. Kumar and J.M. Kleinberg, 'Fairness measures for resource allocation', *SIAM J. Comput.*, **36**(3), 657–680, (2006).
[15] R.J. Lipton, E. Markakis, E. Mossel, and A. Saberi, 'On approximately fair allocations of indivisible goods', in *ACM Conference on Electronic Commerce*, eds., J.S. Breese, J. Feigenbaum, and M.I. Seltzer, pp. 125–131. ACM, (2004).
[16] M.T. Marsh and D. Schilling, 'Equity measurement in facility location analysis: a review and framework', *European Journal of Operational Research*, **74**(1), 1–17, (1994).
[17] J. Nash, 'The bargaining problem', *Econometrica*, **18**, 155–162, (1950).
[18] J.G. Oxley, *Matroid Theory*, Oxford University Press, 1992.
[19] C H. Papadimitriou and M. Yannakakis, 'On the approximability of trade-offs and optimal access of web sources', in *FOCS*, pp. 86–92, (2000).
[20] R. Ravi and M. X. Goemans, 'The constrained minimum spanning tree problem (extended abstract)', in *SWAT*, eds., R. G. Karlsson and A. Lingas, volume 1097 of *LNCS*, pp. 66–75. Springer, (1996).
[21] A. Schrijver, *Combinatorial Optimization: Polyhedra and Efficiency*, Springer-Verlag Berlin and Heidelberg GmbH & Co., 2003.
[22] C. Stein and J. Wein, 'On the existence of schedules that are near-optimal for both makespan and total weighted completion time', *Oper. Res. Lett.*, **21**(3), 115–122, (1997).
[23] V.V. Vazirani, *Approximation algorithms*, Springer, 2004.
[24] A. Warburton, 'Approximation of pareto optima in multiple-objective, shortest-path problems', *Operations Research*, **35**(1), 70–79, (1987).