

Some methods for evaluating the optimality of elements in matroids with ill-known weights¹

Jérôme Fortin^a, Adam Kasperski^b, Paweł Zieliński^{c,*}

^a*Institut de Recherche en Informatique de Toulouse, Université Paul Sabatier, 118 route de Narbonne, 31062, Toulouse, cedex 4, France*

^b*Institute of Industrial Engineering and Management, Wrocław University of Technology, Wybrzeże Wyspiańskiego 27, 50-370 Wrocław, Poland*

^c*Institute of Mathematics and Computer Science, Wrocław University of Technology, Wybrzeże Wyspiańskiego 27, 50-370 Wrocław, Poland*

Abstract

In this paper the class of matroidal combinatorial optimization problems with imprecise weights of elements is considered. The imprecise weights are modeled by intervals and fuzzy intervals. The concepts of possible and necessary optimality under imprecision are recalled. Some efficient methods for evaluating the possible and necessary optimality of elements in the interval-valued problems are proposed. Some efficient algorithms for computing the exact degrees of possible and necessary optimality of elements in the fuzzy-valued problems are designed.

Key words: Matroid, Fuzzy interval, Possibility theory, Combinatorial optimization, Gradual number

1 Introduction

This paper deals with combinatorial optimization problems, in which a finite set of elements $E = \{e_1, e_2, \dots, e_n\}$ and a set Φ of subsets of E , called a set of

* Corresponding author.

Email addresses: fortin@irit.fr (Jérôme Fortin), adam.kasperski@pwr.wroc.pl (Adam Kasperski), Pawel.Zielinski@pwr.wroc.pl (Paweł Zieliński).

¹ A preliminary version of this paper has appeared as an extended abstract in *Proceedings of the 6th International Workshop on Fuzzy Logic and Applications, WILF 2005*

feasible solutions, are given. A nonnegative weight w_e is associated with every element $e \in E$ and we seek a solution $X \in \Phi$ whose total weight $\sum_{e \in X} w_e$ is maximal or minimal. This formulation permits to model a lot of problems. Consider for instance the minimum spanning tree problem, in which E is a set of edges of a given undirected graph $G = (V, E)$ and Φ consists of all subsets of the edges that form spanning trees of G . We seek a spanning tree whose total weight is minimal. A wide review of the classical combinatorial optimization problems can be found for instance in [12,15].

In the deterministic case the weights of all elements are precisely known. Therefore the elements of E can be divided into two groups: those which belong to an optimal solution (*optimal elements*) and those which do not belong to an optimal one. In this paper we discuss the case in which the element weights may be imprecise. One of the simplest forms of the uncertainty representations is to assume that the value of a weight may fall within a given range specified by a closed interval $[w_e^-, w_e^+]$. In this *interval-valued* case the elements form three groups: those that are optimal for sure (*necessarily optimal elements*), those that are not optimal for sure and the elements whose optimality is unknown (*possibly optimal elements*).

A *fuzzy interval* is a generalization of a classical one and it allows us to evaluate the imprecise weights in a more sophisticated way. The membership function of a fuzzy interval can be viewed as a *possibility distribution* for the values of a weight. The interpretation of the possibility distribution as well as some methods of obtaining it from the possessed knowledge are described in a book [5], which is entirely devoted to *possibility theory*. In the *fuzzy-valued* case the notions of possible and necessary optimality can be extended and every element can be now characterized by two numbers from interval $[0,1]$, called the *degrees of possible and necessary optimality*. It turns out that the difficulty of computing the optimality degrees is in the interval-valued case. If there are efficient algorithms for deciding whether a given element is possibly (necessarily) optimal in the interval-valued case, then the degrees of optimality can be efficiently computed as well. A standard tool is the bisection algorithm which however gives only the approximate values of the optimality degrees.

In this paper, we wish to investigate a special class of combinatorial optimization problems, namely the *matroidal problems*. For this type of problems the set of feasible solution Φ is a set of bases of a given *matroid*. A good introduction to matroids can be found for instance in [14] and some matroidal problems are addressed in [12,15]. We first investigate the problem with interval weights and we then extend the results to the fuzzy-valued case. It turns out that the special structure of a matroidal problem allows us to evaluate the possible and necessary optimality of elements in the interval case and to compute efficiently the exact values of the degrees of possible and necessary optimality of a given element in the fuzzy case. It is worth pointing out that

the problem of evaluating the optimality of elements is not always an easy one. For instance, asserting whether a given element is possibly optimal in the longest (shortest) path problem with interval weights is NP-complete [2]. The same holds for the minimum assignment and minimum cut problems [11]. However, all of these problems are not matroidal ones.

This paper is organized as follows. In Section 2, we recall the definition of a matroid and a matroidal combinatorial optimization problem (that is the one with precise weights). We introduce the concept of an optimal element and we present an efficient method of deciding whether a given element is optimal. Section 3 is devoted to matroidal problems in which weights are specified as closed intervals. We introduce the notions of possible and necessary optimality of elements and we show some efficient methods of deciding whether a given element is possibly (necessarily) optimal. We refine results obtained in [10]. We then present three particular matroidal problems to which the results obtained in this section can be applied. The first one is the *minimum spanning tree problem* as described previously. The second one is the *selecting items problem* which can be viewed as a basic resource allocation problem. The last one is the *scheduling problem* in which each job has a due date, and we wish to minimize the weighted number of late jobs. In Section 4, we discuss the case in which the weights of elements are given as fuzzy intervals. We use the notion of gradual numbers [6–8] that permit to express a fuzzy interval as a pair of gradual numbers. Thus the results from the interval-valued case can be then naturally extended to the fuzzy-valued one. In consequence, we provide some efficient algorithms that compute the exact optimality degrees of elements.

2 Matroidal problem with precise weights

Consider a system (E, \mathcal{I}) , where $E = \{e_1, \dots, e_n\}$ is a nonempty ground set and \mathcal{I} is a collection of subsets of E closed under inclusion, i.e. if $B \in \mathcal{I}$ and $A \subseteq B$ then $A \in \mathcal{I}$. The elements of \mathcal{I} are called *independent sets*. The system $\mathcal{M} = (E, \mathcal{I})$ is called a *matroid* (see e.g. [14]) if it satisfies the following *growth property*: if $A \in \mathcal{I}$, $B \in \mathcal{I}$ and $|A| < |B|$, then there exists $e \in B \setminus A$ such that $A \cup \{e\} \in \mathcal{I}$. The maximal (under inclusion) independent sets in \mathcal{I} are called *bases*. The minimal (under inclusion) sets not in \mathcal{I} are called *circuits*. The construction of any base is not a difficult issue. If σ specifies an order of elements of E , then the corresponding base B_σ can be constructed by Algorithm 1. We call B_σ the *base induced by σ* .

The running time of Algorithm 1 is $\mathcal{O}(nf(n))$, where $f(n)$ is the time required for detecting a circuit in set $B \cup \{e_i\}$. This running time depends on the particular structure of a given matroid.

Algorithm 1 Constructing a base of a matroid

Require: A matroid (E, \mathcal{I}) , a sequence $\sigma = (e_1, \dots, e_n)$ of elements of E .

Ensure: A base B_σ .

```
 $B_\sigma \leftarrow \emptyset$   
for  $i \leftarrow 1$  to  $n$  do  
    if  $B_\sigma \cup \{e_i\} \in \mathcal{I}$  then  $B_\sigma \leftarrow B_\sigma \cup \{e_i\}$   
end for  
return  $B_\sigma$ 
```

Let us denote by $\text{pred}(e, \sigma)$ the set of elements which precede element e in sequence σ . The following property expresses that the choice of an element e by Algorithm 1 does not depend on the order of elements placed before e in the sequence used by the algorithm.

Proposition 1 ([10]) *Let σ and ρ be two sequences of the elements of E . Let $e \in E$ be an element such that $\text{pred}(e, \sigma) \subseteq \text{pred}(e, \rho)$. If $e \notin B_\sigma$ then $e \notin B_\rho$.*

Assume that for every element $e \in E$ there is given a nonnegative weight w_e . The set of feasible solutions consists of all bases of a given matroid (E, \mathcal{I}) . We will denote this set by \mathcal{B} . A *matroidal combinatorial optimization problem* is the following one:

$$\mathcal{M} : \max_{B \in \mathcal{B}} \sum_{e \in B} w_e. \quad (1)$$

Hence, we wish to find a solution (base) $B \in \mathcal{B}$ for which the total weight is maximal. It is well known, that this problem can be solved by means of a *greedy algorithm*, that is Algorithm 1 with sequence σ , in which the elements are sorted in the nonincreasing order of their weights [15]. The greedy algorithm constructs an *optimal base* in $\mathcal{O}(n \log n + nf(n))$ time.

A given element $f \in E$ is said to be *optimal*, if f is a part of an optimal base. Checking whether f is optimal is not a difficult issue. Let $\sigma^*(\mathbf{w}, f)$ denote a special sequence of elements of E , in which the elements are sorted in the nonincreasing order of their weights $\mathbf{w} = (w_e)_{e \in E}$. Moreover, if $w_f = w_e$, $e \neq f$, then element f precedes element e in this sequence. The following proposition gives a necessary and sufficient condition for establishing whether a given element is optimal:

Proposition 2 ([10]) *A given element f is optimal if and only if $f \in B_{\sigma^*(\mathbf{w}, f)}$*

Proposition 2 suggests an $\mathcal{O}(n \log n + nf(n))$ algorithm for evaluating the optimality of a given element f , where $\mathcal{O}(n \log n)$ is the time required for forming the sequence $\sigma^*(\mathbf{w}, f)$. We now show that this running time can be improved. Let $\sigma(\mathbf{w}, f)$, $f \in E$, denote a sequence, such that $\text{pred}(f, \sigma(\mathbf{w}, f)) = \{e \in E : w_e > w_f\}$. It is clear that sequence $\sigma(\mathbf{w}, f)$ can be constructed in

$\mathcal{O}(n)$ time since it is not necessary to order the elements (we only require elements $e \in E$, such that $w_e > w_f$, to appear before f).

Proposition 3 *A given element f is optimal if and only if $f \in B_{\sigma(\mathbf{w}, f)}$.*

PROOF. It is clear that $\text{pred}(f, \sigma(\mathbf{w}, f)) = \text{pred}(f, \sigma^*(\mathbf{w}, f))$. Thus, by Proposition 1, $f \in B_{\sigma^*(\mathbf{w}, f)}$ if and only if $f \in B_{\sigma(\mathbf{w}, f)}$. Hence, by Proposition 2, f is optimal if and only if $f \in B_{\sigma(\mathbf{w}, f)}$. \square

From Proposition 3, we immediately obtain a method for evaluating the optimality of an element $f \in E$ which requires $\mathcal{O}(n + nf(n)) = \mathcal{O}(nf(n))$ time. This can be done by means of Algorithm 1 in which the input sequence $\sigma(\mathbf{w}, f)$ is formed in $\mathcal{O}(n)$ time and base $B_{\sigma(\mathbf{w}, f)}$ is then constructed in $\mathcal{O}(nf(n))$ time. According to Proposition 3, element f is optimal if and only if $f \in B_{\sigma(\mathbf{w}, f)}$.

In the case when element $f \in E$ is not optimal (i.e. it is not a part of an optimal base), a natural question arises: how far is f from optimality. In other words, what is the minimal nonnegative real number δ_f that added to the weight of f makes it optimal. Clearly, δ_f can be calculated as follows:

$$\delta_f = \max_{B \in \mathcal{B}} \sum_{e \in B} w_e - \max_{B \in \mathcal{B}_f} \sum_{e \in B} w_e,$$

where \mathcal{B}_f is the set of all bases containing f .

3 Matroidal problem with interval weights

Suppose that the weights of elements are not precisely known, but they are known to belong to an interval $W_e = [w_e^-, w_e^+]$, $e \in E$. It means that the actual value of a weight will take some value within this interval, but it is not possible to predict at present which one.

We define a *configuration* as a precise instantiation of the weight of each element $e \in E$, that is $\mathbf{w} = (w_e)_{e \in E}$, $w_e \in W_e$. Thus, every configuration expresses a realization of the weights which may occur. We denote by Γ the set of all configurations, i.e. $\Gamma = \times_{e \in E} [w_e^-, w_e^+]$. We use $w_e(\mathbf{w})$ to denote the weight of element $e \in E$ in configuration $\mathbf{w} \in \Gamma$. Among the configurations in Γ , we distinguish two *extreme* ones. Namely, the configurations $\mathbf{w}_{\{f\}}^+$ and $\mathbf{w}_{\{f\}}^-$ such that:

$$w_e(\mathbf{w}_{\{f\}}^+) = \begin{cases} w_e^+ & \text{if } e = f, \\ w_e^- & \text{otherwise} \end{cases}, \quad w_e(\mathbf{w}_{\{f\}}^-) = \begin{cases} w_e^- & \text{if } e = f, \\ w_e^+ & \text{otherwise} \end{cases}, \quad e \in E. \quad (2)$$

A given element $f \in E$ is *possibly optimal* if and only if it is optimal in some configuration $\mathbf{w} \in \Gamma$. A given element $f \in E$ is *necessarily optimal* if and only if it is optimal in all configurations $\mathbf{w} \in \Gamma$. Instead of being optimal or not, like in the deterministic case, the elements now form three groups: those that are optimal for sure (necessarily optimal elements), those that are not optimal for sure, and the elements whose optimality is unknown (possibly optimal elements). Note that, if an element $f \in E$ is necessarily optimal, then it is also possibly optimal but the converse statement is not true.

We can obtain more information about optimality of $f \in E$. Let $\delta_f(\mathbf{w})$, $\mathbf{w} \in \Gamma$, denote the minimal nonnegative real number such that f with weight $w_f(\mathbf{w}) + \delta_f(\mathbf{w})$ becomes optimal in configuration \mathbf{w} . Let us define the widest interval (bounds) of possible values of δ_f as follows:

$$\Delta_f = [\delta_f^-, \delta_f^+], \quad \delta_f^- = \min_{\mathbf{w} \in \Gamma} \delta_f(\mathbf{w}) \quad \text{and} \quad \delta_f^+ = \max_{\mathbf{w} \in \Gamma} \delta_f(\mathbf{w}).$$

The interval $\Delta_f = [\delta_f^-, \delta_f^+]$ expresses whether f is close to be optimal, that is δ_f^- represents how far is f from being possibly optimal and δ_f^+ how far is f from being necessarily optimal. We can easily deduce the optimality of element f from δ_f^- and δ_f^+ by the following proposition:

Proposition 4 *An element f is possibly (resp. necessarily) optimal if and only if $\delta_f^- = 0$ (resp. $\delta_f^+ = 0$).*

In the next section we show that checking whether a given element is possibly (necessarily) optimal can be efficiently done.

3.1 Evaluating the optimality of elements

It turns out that due to the special matroidal structure of the problem, the optimality of a given element can be characterized in terms of two extreme configurations. The following theorem characterizes a possibly optimal element.

Theorem 5 *Element $f \in E$ is possibly optimal if and only if $f \in B_{\sigma(\mathbf{w}_{\{f\}}^+, f)}$.*

PROOF. (\Rightarrow) From the possible optimality of f it follows that there exists a configuration $\mathbf{w} \in \Gamma$ in which element f is optimal. Proposition 3 implies that f is a part of base $B_{\sigma(\mathbf{w}, f)}$ induced by $\sigma(\mathbf{w}, f)$. It is easy to observe that $\text{pred}(f, \sigma(\mathbf{w}_{\{f\}}^+, f)) \subseteq \text{pred}(f, \sigma(\mathbf{w}, f))$. Therefore Proposition 1 now yields $f \in B_{\sigma(\mathbf{w}_{\{f\}}^+, f)}$.

(\Leftarrow) If f is a part of base $B_{\sigma(\mathbf{w}_{\{f\}}^+, f)}$, then f is optimal in configuration $\mathbf{w}_{\{f\}}^+$ (see Proposition 3). Hence f is possibly optimal. \square

The following theorem characterizes a necessarily optimal element.

Theorem 6 *Element $f \in E$ is necessarily optimal if and only if $f \in B_{\sigma(\mathbf{w}_{\{f\}}^-, f)}$.*

PROOF. (\Rightarrow) If f is necessarily optimal, then it is optimal for all configurations, in particular for $\mathbf{w}_{\{f\}}^-$. Thus by Proposition 3, $f \in B_{\sigma(\mathbf{w}_{\{f\}}^-, f)}$.

(\Leftarrow) Suppose $f \in B_{\sigma(\mathbf{w}_{\{f\}}^-, f)}$. Consider any configuration $\mathbf{w} \in \Gamma$. It is easy to see that $\text{pred}(e, \sigma(\mathbf{w}, f)) \subseteq \text{pred}(e, \sigma(\mathbf{w}_{\{f\}}^-, f))$. We conclude from Proposition 3 that $f \in B_{\sigma(\mathbf{w}, f)}$ and, by Proposition 1, f is optimal in \mathbf{w} . Hence, f is optimal for all configurations $\mathbf{w} \in \Gamma$ and consequently it is necessarily optimal. \square

Making use of Theorems 5 and 6, one can easily evaluate the possible and necessary optimality of an element f . If we wish to assert whether f is possibly optimal, we apply Algorithm 1 in which the order of elements is specified by $\sigma(\mathbf{w}_{\{f\}}^+, f)$. Element f is then possibly optimal if and only if the obtained base contains f . In a similar way we assert whether f is necessarily optimal, that is we apply Algorithm 1 in which the order of elements is specified by $\sigma(\mathbf{w}_{\{f\}}^-, f)$ and check if the obtained base contains f . The running time of both methods is $\mathcal{O}(nf(n))$.

Theorems 5 and 6 also allow us to determine interval $\Delta_f = [\delta_f^-, \delta_f^+]$. If $f \notin B_{\sigma(\mathbf{w}_{\{f\}}^+, f)}$ (which indicates that $\delta_f^- > 0$) then set $B_{\sigma(\mathbf{w}_{\{f\}}^+, f)} \cup \{f\}$ contains a unique circuit C . We can find an element $g \in C \setminus \{f\}$ of the minimal value of w_g^- . Then, from Theorem 5, it follows that $\delta_f^- = w_g^- - w_f^+$. Similarly, if $f \notin B_{\sigma(\mathbf{w}_{\{f\}}^-, f)}$ (which indicates that $\delta_f^+ > 0$) then set $B_{\sigma(\mathbf{w}_{\{f\}}^-, f)} \cup \{f\}$ contains a unique circuit C . We can find an element $g \in C \setminus \{f\}$ of the minimal value of w_g^+ and, by Theorem 6, $\delta_f^+ = w_g^+ - w_f^-$. It is easily seen that both values δ_f^- and δ_f^+ for a given element $f \in E$ can be computed in $\mathcal{O}(nf(n))$ time.

We now focus on the problem of evaluating the optimality of all elements of E . An obvious method of performing this task is execution of Algorithm 1 for every $f \in E$, with the order of elements specified by $\sigma(\mathbf{w}_{\{f\}}^+, f)$ (for the possible optimality) or $\sigma(\mathbf{w}_{\{f\}}^-, f)$ (for the necessary optimality). This yields complexity $\mathcal{O}(n^2f(n))$. However, the complexity of detecting all the possibly optimal elements can be additionally reduced.

Let \mathbf{w}^- be the configuration where the values of all weights are set to their lower bounds. Let $\sigma^*(\mathbf{w}^-)$ denote the sequence of elements of E , in which the elements are sorted in the nonincreasing order of their weights in \mathbf{w}^- . The base $B_{\sigma^*(\mathbf{w}^-)}$ is induced by sequence $\sigma^*(\mathbf{w}^-)$ and it is the optimal base in configuration \mathbf{w}^- . The following theorem holds:

Theorem 7 ([10]) *An element $e \notin B_{\sigma^*(\mathbf{w}^-)}$ is possibly optimal if and only if an element $g \neq e$ in the circuit $C \subseteq B_{\sigma^*(\mathbf{w}^-)} \cup \{e\}$ of the minimal value of w_g^- satisfies $w_e^+ \geq w_g^-$.*

We now describe how to use Theorem 7 to detect all the possibly optimal elements in E . First, we compute base $B_{\sigma^*(\mathbf{w}^-)}$ by applying Algorithm 1 in which the order of elements is specified by $\sigma^*(\mathbf{w}^-)$. All the elements $e \in B_{\sigma^*(\mathbf{w}^-)}$ are then possibly optimal by definition. Next, for all $e \in E \setminus B_{\sigma^*(\mathbf{w}^-)}$ we determine the element g from Theorem 7. In a typical situation this requires $\mathcal{O}(nf(n))$ time, where $f(n)$ is the time required for detecting circuit C in $B_{\sigma^*(\mathbf{w}^-)} \cup \{e\}$. Thus, all the possibly optimal elements can be detected in $\mathcal{O}(n \log n + nf(n))$ time. Unfortunately, we are not able to prove a counterpart of Theorem 7 for the necessary optimal elements.

We have assumed that the classical matroidal problem (see (1)) is a maximization one. However, all the results apply to a minimization problem as well. Suppose \mathcal{M} is an interval problem in which we seek a minimum weighted base (that is criterion (1) is replaced with the minimization one). We can replace the interval weights $[w_e^-, w_e^+]$ in \mathcal{M} with $[M - w_e^+, M - w_e^-]$ for all $e \in E$, where $M = \max_{e \in E} w_e^+$, obtaining problem \mathcal{M}' in which we seek a maximum weighted base. Since all the bases of a matroid have the same cardinality it is easy to check that element f is possibly (necessarily) optimal in \mathcal{M}' if and only if it is possibly (necessarily) optimal in \mathcal{M} .

3.2 Practical examples of matroidal problems

In the next subsections we will analyze in detail three particular interval-valued matroidal problems. The classical, deterministic versions of these problems are well-known and they arise in a number of applications.

3.2.1 Minimum spanning tree problem

Let $E = \{e_1, \dots, e_n\}$ be a set of edges of a given connected and undirected graph $G = (V, E)$. The set \mathcal{I} consists of all the subsets of edges $E' \subseteq E$ such that subgraph $G' = (V, E')$ is acyclic (that is \mathcal{I} is the set of all forests in G). The system (E, \mathcal{I}) in this problem is one of the best known examples of matroids (it is so-called *graphic matroid*). A base is a *spanning tree* of G

and a circuit is a subset of edges that forms a simple cycle in G . In the classical minimum spanning tree problem, we seek a spanning tree for which the total weight is minimal. The greedy algorithm which computes the optimal spanning tree is known in the literature as Kruskal's algorithm. The complexity of Kruskal's algorithm is $\mathcal{O}(|E| \log |E|)$ [3].

Consider the case in which the weights of edges of G are specified as closed intervals. The problem of detecting the possibly and the necessarily optimal edges have been studied in [1] and [13] (in [13] a possibly optimal edge is called *weak* and a necessarily optimal one is called *strong*). This characterization can be used for preprocessing the problem before calculating a *robust spanning tree*. A method for detecting all the possibly and the necessarily optimal edges adopted in [13] consists in executing $|E|$ times the Kruskal's algorithm for configurations $\mathbf{w}_{\{e\}}^+$ and $\mathbf{w}_{\{e\}}^-$, $e \in E$. Thus its running time is $\mathcal{O}(|E|^2 \log |E|)$. The complexity of calculation of all the possibly optimal edges can be significantly reduced. In [1] the following proposition (which is consequence of the more general Theorem 7) was proven:

Proposition 8 ([1]) *Let \mathbf{w}^+ be a configuration where all the weights are at their upper bounds and let T be a minimum spanning tree in \mathbf{w}^+ . An edge $e = (u, v) \in E \setminus T$ is possibly optimal if and only if an edge f of the maximal weight w_g^+ on the path from u to v in T satisfies $w_e^- \leq w_g^+$.*

The idea of algorithm for detecting all the possibly optimal elements, which is based on Proposition 8, is as follows. We start with computing the minimum spanning tree T in configuration \mathbf{w}^+ . All the edges $e \in T$ are then possibly optimal by definition. Then, for every edge $e \in E \setminus T$ we can compute in $\mathcal{O}(|V|)$ time the proper edge g and check whether $w_e^- \leq w_g^+$. The overall complexity becomes then $\mathcal{O}(|E| \log |E| + |E||V|)$, which can be additionally reduced to $\mathcal{O}(|E| \log |E| + |V|^2)$ (see [1] for details). Observe that in dense graphs ($|E| \approx |V|^2$) this complexity is $\mathcal{O}(|E| \log |E|)$, which is the same as the running time of Kruskal's algorithm.

Unfortunately, we are not able to prove a counterpart of Proposition 8 for detecting all the necessarily optimal edges. Therefore, all these edges can be detected in $\mathcal{O}(|E|^2 \log |V|)$ time by executing Algorithm 1, in which the order of elements is specified by $\sigma(\mathbf{w}_{\{f\}}^+, f)$, where $\log |V|$ is the time required for deciding whether an edge belongs to a cycle in G .

3.2.2 Selecting items problem

Let $E = \{e_1, \dots, e_n\}$ be a set of elements (items). The set \mathcal{I} consists of all subsets of E whose cardinalities are less than or equal to a given number p such that $1 \leq p < n$. It can be easily verified that system (E, \mathcal{I}) is a matroid

(it is so called *uniform matroid*). A base is a subset A of E such that $|A| = p$ and a circuit C is a subset of E such that $|C| = p + 1$. In the selecting items problem, we seek a subset of E whose cardinality is exactly p , that maximizes the total weight. The selecting items problem can be viewed as a basic resource allocation problem with linear cost function [9]. In the classical case the best solution can be easily obtained by selecting p elements of the greatest weights. This can be performed in $\mathcal{O}(n)$ time in the following way: first, the p -th greatest weighted element $g \in E$ is calculated (this can be done in $\mathcal{O}(n)$ time, see e.g. [3]) and added to A ; then $p - 1$ elements of the weights greater than or equal to w_g are added to A .

Consider now the case in which the weights of elements in the problem are given as closed intervals. The following two propositions allow us to detect all the possibly and necessarily optimal elements very efficiently.

Proposition 9 *Let \mathbf{w}^- be a configuration where all the weights are at their lower bounds. Let g be the p -th greatest weighted element in \mathbf{w}^- . Then element $e \in E$ is possibly optimal if and only if $w_e^+ \geq w_g^-$.*

PROOF. (\Rightarrow) Suppose by contradiction that e is possibly optimal and $w_e^+ < w_g^-$. Condition $w_e^+ < w_g^-$ means that there are at least p elements which weights are strictly greater than the weight of e in configuration $\mathbf{w}_{\{e\}}^+$. Thus e cannot be a part of a base $B_{\sigma(\mathbf{w}_{\{e\}}^+, e)}$ induced by $\sigma(\mathbf{w}_{\{e\}}^+, e)$. This implies that e is not possibly optimal (see Theorem 5), which is a contradiction.

(\Leftarrow) If $w_e^+ \geq w_g^-$ the element e is one of the p elements of the greatest weights in configuration $\mathbf{w}_{\{e\}}^+$. Thus e is a part of a base $B_{\sigma(\mathbf{w}_{\{e\}}^+, e)}$ induced by $\sigma(\mathbf{w}_{\{e\}}^+, e)$ and by Theorem 5, it is possibly optimal. \square

Proposition 10 *Let \mathbf{w}^+ be a configuration where all the weights are at their upper bounds. Let f be the p -th greatest weighted element and let g be the $(p + 1)$ -th greatest weighted element in \mathbf{w}^+ . Then element $e \in E$ is necessarily optimal if and only if $w_e^+ \geq w_f^+$ and $w_e^- \geq w_g^+$.*

PROOF. (\Rightarrow) Suppose by contradiction that element e is necessarily optimal and $w_e^+ < w_f^+$ or $w_e^- < w_g^+$. In the first case ($w_e^+ < w_f^+$) element e cannot be a part of the optimal base in configuration \mathbf{w}^+ and in the second case ($w_e^- < w_g^+$) it cannot be a part of the optimal base in configuration $\mathbf{w}_{\{e\}}^-$. This contradicts the assumption that e is necessarily optimal.

(\Leftarrow) Conditions $w_e^+ \geq w_f^+$ and $w_e^- \geq w_g^+$ assure that element e is one of the p elements of the greatest weights in configuration $\mathbf{w}_{\{e\}}^-$. Thus e is a part of

a base $B_{\sigma(\mathbf{w}_{\{e\}^-}, e)}$ induced by $\sigma(\mathbf{w}_{\{e\}^-}, e)$ and by Theorem 6, it is necessarily optimal. \square

Propositions 9 and 10 allow us to detect all the possibly and necessarily optimal elements very efficiently. The k -th greatest element in a given configuration can be found in $\mathcal{O}(n)$ time (see e.g. [3]). Thus the overall running time of the algorithm for detecting all the possibly (necessarily) optimal elements is $\mathcal{O}(n)$.

3.2.3 Scheduling problem $1|p_i = 1|\sum w_i U_i$

Let $J = \{1, \dots, n\}$ be a set of jobs to be processed on a single machine. Every job $i \in J$ has unit processing time $p_i = 1$. For every job $i \in J$ there are given: a due date d_i and a weight w_i . A *schedule* is a sequence π of jobs. A job $i \in J$ is called *late* in π if its completion time in π is greater than d_i , otherwise job i is called *on-time* in π . We seek a schedule for which the sum of weights of all late jobs is minimal. A subset of jobs $S \subseteq J$ belongs to \mathcal{I} if and only if all the jobs in S are on-time in a certain schedule π . It is easy to decide whether a given set S belongs to \mathcal{I} . To do this, it is enough to schedule first all the jobs in S in order of nondecreasing due dates and then all the remaining jobs in an arbitrary order. Then $S \in \mathcal{I}$ if and only if all the jobs in S are on-time in the resulting schedule [3]. This schedule is said to be in the *canonical* form. It is easily seen that the problem consists now in determining a set $S \in \mathcal{I}$ with the maximal value of $F(S) = \sum_{i \in S} w_i$. The optimal solution can be then obtained by constructing the corresponding canonical schedule for S . It can be proven that system (J, \mathcal{I}) is matroid and the optimal schedule (in the canonical form) can be found in $\mathcal{O}(n^2)$ time by greedy algorithm [3,12].

Before we consider the interval case, we show how to detect efficiently a circuit in the considered problem. Let $S = \{j_1, \dots, j_k\}$, $k \leq n$, be a given base. Assume that $d_{j_1} \leq \dots \leq d_{j_k}$. Let $i \in J$ be a job such that $i \notin S$. Then set $S \cup \{i\}$ contains a circuit C . The circuit C is the minimal subset of jobs in $S \cup \{i\}$ which cannot be all scheduled on-time. The circuit C can be detected in the following way: find the smallest number $r \in \{1, \dots, k\}$ such that $d_{j_r} \geq d_i$ and $r + 1 > d_{j_r}$; set $C = \{j_1, \dots, j_r, i\}$. It is not difficult to check that such a number r must exist, since otherwise we could create a canonical schedule in which all the jobs in $S \cup \{i\}$ are on-time (this would contradict the fact that S is a base). To see that C is a circuit schedule jobs in C in order of nondecreasing due dates (note that we try to construct a canonical schedule) and denote the resulting schedule by σ . From the definition of r we conclude that only the last job in σ (either i or j_r) is late. Since all the processing times are equal to 1, we can remove any job from σ and we get a schedule in which all the jobs are on-time. This implies that all the subsets of C with cardinality

$|C| - 1$ are independent and C is the minimal dependent subset.

Assume now that the weights of jobs are uncertain and they are given as closed intervals. A possibly optimal job is on-time job in an optimal schedule for some configuration of the weights and a necessarily optimal job is on-time in some optimal schedule for all configurations of the weights. Such a characterization under the uncertain weights may be useful. For example, the necessarily optimal jobs should be always processed first, while non-possibly optimal ones should be processed last.

Now we can use Theorem 7 to detect efficiently all the possibly optimal jobs. We compute first in $\mathcal{O}(n^2)$ time the optimal schedule π in configuration of weights \mathbf{w}^- . The schedule π is in the canonical form, thus we can easily determine the set $S = \{\pi(1), \dots, \pi(k)\}$ of jobs which are on-time in π . The set S is the optimal base in configuration \mathbf{w}^- , thus all the jobs $i \in S$ are possibly optimal. Suppose that $i \notin S$. Then, we can determine in $\mathcal{O}(n)$ time the smallest number r such that $d_{\pi(r)} \geq d_i$ and $r + 1 > d_{\pi(r)}$ and check whether $w_i^+ \geq \min\{w_{\pi(1)}^-, \dots, w_{\pi(r)}^-\}$ (see Theorem 7). Therefore, all the possibly optimal jobs can be detected in $\mathcal{O}(n^2)$ time. Unfortunately, it is a difficult issue to give an $\mathcal{O}(n^2)$ algorithm for detecting all the necessarily optimal jobs. Thus, one has to run n times Algorithm 1 in which the order of elements is specified by $\sigma(\mathbf{w}_{\{f\}}^+, f)$, where $\mathcal{O}(n)$ is time required for checking independence, which gives $\mathcal{O}(n^3)$ time.

4 Matroidal problem with fuzzy weights

In this section we discuss the case in which the imprecise weights are modeled by means of fuzzy intervals. We introduce degrees of possible and necessary optimality of a given element $f \in E$, which are numbers from interval $[0,1]$ that characterize the optimality of f . We describe a recent concept of a gradual number [7,8] and its connections with the concept of a fuzzy interval. Basing on the notion of a gradual number we construct some efficient and exact methods of computing the values of degrees of optimality of a given element.

4.1 Degrees of possible and necessary optimality of elements

A *fuzzy interval* is a generalization of the classical one and it allows us to express the uncertainty connected with ill-known parameters in a more sophisticated manner. A fuzzy interval \tilde{W} is a fuzzy set defined in the space of

real numbers, whose membership function $\mu_{\tilde{W}}(x)$ is normal, quasi-concave and upper semi-continuous. A *fuzzy interval of the L-R type*, denoted as $(w^-, w^+, \alpha, \beta)_{L-R}$, is very popular and convenient in applications (see, e.g. [5]). Its membership function is of the following form:

$$\mu_{\tilde{W}}(x) = \begin{cases} 1 & \text{for } x \in [w^-, w^+], \\ L\left(\frac{w^- - x}{\alpha}\right) & \text{for } x \leq w^-, \\ R\left(\frac{x - w^+}{\beta}\right) & \text{for } x \geq w^+, \end{cases}$$

where L and R are continuous and nonincreasing functions, defined on $[0, +\infty)$, strictly decreasing to zero in those subintervals of the interval $[0, +\infty)$ in which they are positive, and fulfilling condition $L(0) = R(0) = 1$. The parameters α and β are nonnegative real numbers. If $L(x) = R(x) = \max\{0, 1 - x\}$ and $w^- = w^+$, then we obtain a *triangular fuzzy interval*, which is shortly denoted by triple (w, α, β) .

Every fuzzy interval \tilde{W} can be decomposed into its λ -cuts, $\lambda \in (0, 1]$, i.e the sets $\tilde{W}^\lambda = \{x : \mu_{\tilde{W}}(x) \geq \lambda\}$. It can be easily verified [5] that every λ -cut of \tilde{W} is a closed interval.

We now generalize the concept of interval-valued matroidal problem to the fuzzy-valued one and provide a possibilistic formulation of the problem. Assume that for every element $e \in E$ a fuzzy interval \tilde{W}_e is given. This fuzzy interval expresses the uncertainty connected with the ill-known weight \mathfrak{w}_e of element $e \in E$. The membership function $\mu_{\tilde{W}_e}(x)$ is a *possibility distribution* for the values of \mathfrak{w}_e with respect to the following formula (see [5,16]):

$$\Pi(\mathfrak{w}_e = w_e) = \mu_{\tilde{W}_e}(w_e).$$

Let $\mathbf{w} = (w_e)_{e \in E}$, $w_e \in \mathbb{R}$, be a configuration of weights that represents a state of the world. Assuming that the weights are unrelated the joint possibility distribution over configurations, induced by \tilde{W}_e , $e \in E$, is as follows:

$$\pi(\mathbf{w}) = \Pi(\bigwedge_{e \in E} (\mathfrak{w}_e = w_e)) = \min_{e \in E} \Pi(\mathfrak{w}_e = w_e) = \min_{e \in E} \mu_{\tilde{W}_e}(w_e).$$

Hence, the degrees of possible and necessary optimality of an element $f \in E$ are defined as follows:

$$\Pi(f \text{ is optimal}) = \sup_{\{\mathbf{w}: f \text{ is optimal in } \mathbf{w}\}} \pi(\mathbf{w}), \quad (3)$$

$$\mathbf{N}(f \text{ is optimal}) = \inf_{\{\mathbf{w}: f \text{ is not optimal in } \mathbf{w}\}} (1 - \pi(\mathbf{w})). \quad (4)$$

The degrees of optimality can be generalized by fuzzyfying the quantity δ_f in

the following way:

$$\mu_{\tilde{\Delta}_f}(x) = \Pi(\delta_f = x) = \sup_{\{\mathbf{w}: x = \delta_f(\mathbf{w})\}} \pi(\mathbf{w}).$$

The following relations hold:

$$\begin{aligned} \Pi(f \text{ is optimal}) &= \Pi(\delta_f = 0) = \mu_{\tilde{\Delta}_f}(0), \\ \text{N}(f \text{ is optimal}) &= \text{N}(\delta_f = 0) = 1 - \sup_{x > 0} \mu_{\tilde{\Delta}_f}(x). \end{aligned}$$

The methods used to evaluate the optimality in interval-valued matroidal problems are based on the study of some particular configurations of weights. A natural way of extending these methods to solve the fuzzy counterparts of the problems should be the use of some “fuzzy configurations”. When one deals with the classical intervals, a configuration assigns a real number to the weight of each element. This is due to the fact that an interval is a set of real numbers. The recent advances of possibility theory permit to see a fuzzy interval as a classical interval of what is called “gradual numbers”. These gradual numbers can lead to define a gradual configuration and allow us to extend the classical interval computation methods to their fuzzy counterparts. The next section describes the gradual numbers and their relationships with fuzzy intervals.

4.2 Gradual numbers

A fuzzy interval models incomplete knowledge (we know that some parameter lies between two bounds), not the fuzziness per se. The classical intervals model uncertainty in a Boolean way: a value in the interval is possible and a value outside is impossible. The idea of fuzziness is to move from the Boolean way to a gradual one. Hence fuzziness makes the boundaries of the interval softer and thus making uncertainty gradual. In order to model the essence of graduality without uncertainty the concept of a *gradual number* has been recently proposed. Following the notation of [8], a gradual number is defined as follows:

Definition 11 ([8]) *A gradual real number (a gradual number for short) \tilde{r} is defined by an assignment function $\mathcal{A}_{\tilde{r}}$ from $(0, 1]$ to \mathbb{R} .*

A gradual number can be seen as a number parametrized by the value of $\lambda \in (0, 1]$. The arithmetic operations on gradual numbers are defined by operations on their assignment functions. For instance, if \tilde{r} and \tilde{s} are two gradual numbers, then the sum of \tilde{r} and \tilde{s} is defined by summing their assignment functions, that is $\forall \lambda \in (0, 1] \mathcal{A}_{\tilde{r} + \tilde{s}}(\lambda) = \mathcal{A}_{\tilde{r}}(\lambda) + \mathcal{A}_{\tilde{s}}(\lambda)$. The subtraction, product, quotient, maximum and minimum of gradual numbers can be defined in

a similar manner. It is worth pointing out that most algebraic properties of real numbers are preserved for gradual numbers, contrary to the case of fuzzy intervals.

Making use of the notion of gradual number, one can describe a fuzzy interval \tilde{W} by an ordered pair of two gradual numbers $[\tilde{w}^-, \tilde{w}^+]$, where \tilde{w}^- is a *gradual lower bound* of \tilde{W} and \tilde{w}^+ is a *gradual upper bound* of \tilde{W} . In order to ensure the well known shape of a fuzzy interval, \tilde{w}^- and \tilde{w}^+ must satisfy the following properties:

- (a) $\mathcal{A}_{\tilde{w}^-}$ is increasing function,
- (b) $\mathcal{A}_{\tilde{w}^+}$ is decreasing function,
- (c) $\mathcal{A}_{\tilde{w}^-} \leq \mathcal{A}_{\tilde{w}^+}$, i.e $\mathcal{A}_{\tilde{w}^-}(1) \leq \mathcal{A}_{\tilde{w}^+}(1)$.

Therefore a fuzzy interval is an interval of gradual numbers bounded by \tilde{w}^- and \tilde{w}^+ (see Fig. 1). Selecting a gradual number from this interval boils down to picking an element at each λ -cut.

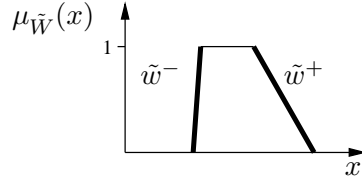


Fig. 1. The left and right bounds of fuzzy interval \tilde{W} (in bold).

Pair $[\tilde{w}^-, \tilde{w}^+]$ describes a fuzzy interval with membership function:

$$\mu_{\tilde{W}}(x) = \begin{cases} \sup\{\lambda \mid \mathcal{A}_{\tilde{w}^-}(\lambda) \leq x\} & \text{if } x \in \mathcal{A}_{\tilde{w}^-}((0, 1]) \\ 1 & \text{if } \mathcal{A}_{\tilde{w}^-}(1) \leq x \leq \mathcal{A}_{\tilde{w}^+}(1) \\ \sup\{\lambda \mid \mathcal{A}_{\tilde{w}^+}(\lambda) \geq x\} & \text{if } x \in \mathcal{A}_{\tilde{w}^+}((0, 1]) \\ 0 & \text{otherwise} \end{cases}$$

Conversely, an upper semi continuous membership function $\mu_{\tilde{W}}(x)$ of a fuzzy interval can be described by the pair of gradual numbers $[\tilde{w}^-, \tilde{w}^+]$ defined as follows:

$$\begin{aligned} \mathcal{A}_{\tilde{w}^-} &: (0, 1] \rightarrow \mathbb{R} \\ \lambda &\mapsto \mathcal{A}_{\tilde{w}^-}(\lambda) = \inf\{x \mid \mu_{\tilde{W}}(x) \geq \lambda\}, \\ \mathcal{A}_{\tilde{w}^+} &: (0, 1] \rightarrow \mathbb{R} \\ \lambda &\mapsto \mathcal{A}_{\tilde{w}^+}(\lambda) = \sup\{x \mid \mu_{\tilde{W}}(x) \geq \lambda\}. \end{aligned}$$

In particular, every fuzzy interval of the L-R type $\tilde{W} = (w^-, w^+, \alpha, \beta)_{L-R}$ can be described by a pair of gradual numbers $[\tilde{w}^-, \tilde{w}^+]$ with the following assignment functions:

$$\mathcal{A}_{\tilde{w}^-}(\lambda) = w^- - L^{-1}(\lambda)\alpha \text{ and } \mathcal{A}_{\tilde{w}^+}(\lambda) = w^+ + R^{-1}(\lambda)\beta. \quad (5)$$

For a deeper discussion on gradual numbers, we refer the reader to [6–8]. In the next section we show how to calculate efficiently the degrees of possible and necessary optimality using the concept of gradual numbers.

4.3 Exact methods for computing the optimality degrees of elements

Denote by $\mathcal{M}(\lambda)$, $\lambda \in (0, 1]$, the interval-valued matroidal problem with weights \tilde{W}_e^λ , $e \in E$, being the λ -cuts of the fuzzy weights. A link between the interval case and the fuzzy one, resulting from formulae (3) and (4) and the fact that if $\alpha < \beta$ then $\tilde{W}_e^\beta \subseteq \tilde{W}_e^\alpha$, $e \in E$, is as follows:

$$\Pi(f \text{ is optimal}) = \sup\{\lambda : f \text{ is possibly optimal in } \mathcal{M}(\lambda)\}, \quad (6)$$

$$N(f \text{ is optimal}) = 1 - \inf\{\lambda : f \text{ is necessarily optimal in } \mathcal{M}(\lambda)\}. \quad (7)$$

If f is not possibly optimal in $\mathcal{M}(0)$, then $\Pi(f \text{ is optimal}) = 0$ and if f is not necessarily optimal in $\mathcal{M}(1)$, then $N(f \text{ is optimal}) = 0$.

Equations (6) and (7) form the theoretical basis for calculating the values of the optimality degrees. They suggest a standard bisection method for determining the optimality degrees (3) and (4) of a fixed element with a given accuracy $\epsilon \in (0, 1)$ via the use of λ -cuts. At each iteration the possible (necessary) optimality of the element is evaluated in the interval-valued matroid $\mathcal{M}(\lambda)$ according to Theorem 5 (Theorem 6). The calculations take $\mathcal{O}(|\log \epsilon^{-1}|nf(n))$ time. Unfortunately, this method gives only approximate values of the optimality degrees.

Further in this section, we propose some efficient algorithms for computing the exact values of the degrees of optimality. Moreover, if $f(n)$ is a polynomial, then the algorithms will be strongly polynomial. These algorithms will exploit the notion of gradual number described in Section 4.2. We will show that expressing a fuzzy interval as a pair of two gradual numbers makes it possible to apply the interval methods (see Section 3.1) to the fuzzy counterpart of the problem.

Suppose that the weights are given by means of fuzzy intervals $\tilde{W}_e = [\tilde{w}_e^-, \tilde{w}_e^+]$, where \tilde{w}_e^- is a gradual lower bound and \tilde{w}_e^+ is a gradual upper bound of \tilde{W}_e . We assume that the support of \tilde{W}_e is compact. To be consistent with this assumption, we have to extend the domains of assignment functions $\mathcal{A}_{\tilde{w}_e^-}$ and $\mathcal{A}_{\tilde{w}_e^+}$ to interval $[0, 1]$. Moreover, we will also assume that function $\mathcal{A}_{\tilde{w}_e^-}$ is continuous and increasing and function $\mathcal{A}_{\tilde{w}_e^+}$ is continuous and decreasing. It is easy to check that these assumptions hold for a wide class of fuzzy intervals of the L-R type (see [5]), which are typically used in applications.

In order to apply the interval methods given in Section 3.1 to the fuzzy interval computations, we need to extend two extreme configurations $\mathbf{w}_{\{f\}}^+$ and $\mathbf{w}_{\{f\}}^-$ (see (2)) to the fuzzy case. Now the *fuzzy extreme configurations* $\tilde{\mathbf{w}}_{\{f\}}^+$ and $\tilde{\mathbf{w}}_{\{f\}}^-$ are the vectors of gradual lower and gradual upper bounds of fuzzy weights defined as follows:

$$\tilde{w}_e(\tilde{\mathbf{w}}_{\{f\}}^+) = \begin{cases} \tilde{w}_e^+ & \text{if } e = f, \\ \tilde{w}_e^- & \text{otherwise} \end{cases}, \quad \tilde{w}_e(\tilde{\mathbf{w}}_{\{f\}}^-) = \begin{cases} \tilde{w}_e^- & \text{if } e = f, \\ \tilde{w}_e^+ & \text{otherwise} \end{cases}, e \in E. \quad (8)$$

Hence $\tilde{w}_e(\tilde{\mathbf{w}}_{\{f\}}^+)$ and $\tilde{w}_e(\tilde{\mathbf{w}}_{\{f\}}^-)$ are gradual weights of $e \in E$ in fuzzy extreme configurations $\tilde{\mathbf{w}}_{\{f\}}^+$ and $\tilde{\mathbf{w}}_{\{f\}}^-$, respectively.

4.3.1 Computing the degree of possible optimality

Assume that we intend to calculate the value of $\Pi(f \text{ is optimal})$ for a given element $f \in E$. The key observation is that in order to do this, it is enough to analyze only the fuzzy configuration $\tilde{\mathbf{w}}_{\{f\}}^+$. This is similar to the interval case in which the extreme configuration \mathbf{w}_f^+ was explored (see Theorem 5). Moreover, it is sufficient to take into account only the intersection points of gradual weight \tilde{w}_f^+ with gradual weights \tilde{w}_e^- , $e \neq f$, in configuration $\tilde{\mathbf{w}}_{\{f\}}^+$ (more precisely the intersection points of their corresponding assignment functions). For simplicity of the notation we will write $\tilde{w}_e^+(\lambda)$ instead of $\mathcal{A}_{\tilde{w}_e^+}(\lambda)$ and $\tilde{w}_e^-(\lambda)$ instead of $\mathcal{A}_{\tilde{w}_e^-}(\lambda)$ for $\lambda \in [0, 1]$.

Let e_1, \dots, e_m be the elements in E , whose gradual lower bounds intersect with the gradual upper bound of the element f in $\tilde{\mathbf{w}}_{\{f\}}^+$. Let the real numbers $\lambda_1, \dots, \lambda_m \in [0, 1]$ denote the cuts such that $\tilde{w}_{e_i}^-(\lambda_i) = \tilde{w}_f^+(\lambda_i)$, $i = 1, \dots, m$. We assume that $\lambda_1 \leq \dots \leq \lambda_m$, thus $\tilde{w}_{e_1}^-(\lambda_1) \geq \dots \geq \tilde{w}_{e_m}^-(\lambda_m)$. Let us also distinguish the elements v_1, \dots, v_r in E whose gradual lower bounds are strictly lower than the gradual upper bound \tilde{w}_f^+ and the elements u_1, \dots, u_q whose gradual lower bounds are strictly greater than the gradual upper bound \tilde{w}_f^+ . The partition of elements is of the form $E = \{u_1, \dots, u_q\} \cup \{e_1, \dots, e_m\} \cup \{v_1, \dots, v_r\} \cup \{f\}$ and it is shown in Fig. 2.

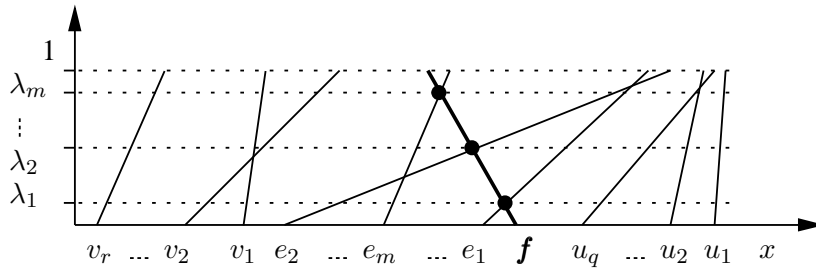


Fig. 2. The partition of E with respect to the intersection points of the gradual upper bound \tilde{w}_f^+ with the gradual lower bounds \tilde{w}_e^- , $e \neq f$, in configuration $\tilde{\mathbf{w}}_{\{f\}}^+$.

Let us now define the following sequences of elements:

$$\begin{aligned}\sigma_0 &= (u_1, \dots, u_q, \mathbf{f}, e_1, \dots, e_m, v_1, \dots, v_r), \\ \sigma_i &= (u_1, \dots, u_q, e_1, \dots, e_i, \mathbf{f}, e_{i+1}, \dots, e_m, v_1, \dots, v_r), \quad i = 1, \dots, m-1, \\ \sigma_m &= (u_1, \dots, u_q, e_1, \dots, e_m, \mathbf{f}, v_1, \dots, v_r).\end{aligned}$$

Note that the sequences differ from each other only in the position of the element f , which depends on the cut λ_i . Let us define $\lambda_0 = 0$.

Observation 1 *If $f \in B_{\sigma_{i-1}}$, $i = 1, \dots, m$, then f is possibly optimal in the interval-valued matroidal problem $\mathcal{M}(\lambda)$ for all $\lambda \in [0, \lambda_i]$.*

PROOF. Observe, that it is sufficient to show that f is possibly optimal in $\mathcal{M}(\lambda_i)$. It is easy to see that the extreme configuration $\mathbf{w}_{\{f\}}^+$ in $\mathcal{M}(\lambda_i)$ is as follows: $w_f(\mathbf{w}_{\{f\}}^+) = \tilde{w}_f^+(\lambda_i)$ and $w_e(\mathbf{w}_{\{f\}}^+) = \tilde{w}_e^-(\lambda_i)$ if $e \neq f$. From the construction of the sequence σ_i , it follows that $\text{pred}(\sigma(\mathbf{w}_{\{f\}}^+, f), f) \subseteq \text{pred}(\sigma_{i-1}, f)$. Thus, from Proposition 1 and the assumption $f \in B_{\sigma_{i-1}}$, we see that f is a part of base $B_{\sigma(\mathbf{w}_{\{f\}}^+, f)}$ in $\mathcal{M}(\lambda_i)$ and, by Theorem 5, it is possibly optimal in $\mathcal{M}(\lambda_i)$. \square

Observation 2 *If $f \notin B_{\sigma_i}$, $i = 0, \dots, m$, then f is not possibly optimal in the interval-valued matroidal problem $\mathcal{M}(\lambda)$, $\lambda \in (\lambda_i, 1]$.*

PROOF. From the definition of sequence σ_i it follows that $\tilde{w}_e^-(\lambda_i) \geq \tilde{w}_f^+(\lambda_i)$ for all $e \in \text{pred}(f, \sigma_i)$ (see also Fig. 2). Let $\lambda > \lambda_i$. From the strict monotonicity of the assignment functions $\mathcal{A}_{\tilde{w}_e^-}$ and $\mathcal{A}_{\tilde{w}_e^+}$ of the gradual lower and upper bounds of \tilde{W}_e , we obtain $\tilde{w}_e^-(\lambda) > \tilde{w}_f^+(\lambda)$ for all $e \in \text{pred}(f, \sigma_i)$. Thus in the interval weighted matroid $\mathcal{M}(\lambda)$ all the elements $e \in \text{pred}(f, \sigma_i)$ must also precede f in the corresponding sequence $\sigma(\mathbf{w}_{\{f\}}^+, f)$ in matroid $\mathcal{M}(\lambda)$. Therefore, according to Proposition 1 and Theorem 5, element f is not possibly optimal in $\mathcal{M}(\lambda)$. \square

Observations 1 and 2, together with formula (6) yield:

Proposition 12 *If $f \in B_{\sigma_m}$ then $\Pi(f \text{ is optimal}) = 1$. Otherwise, let k be the smallest index in $\{0, 1, \dots, m\}$ such that $f \notin B_{\sigma_k}$. Then $\Pi(f \text{ is optimal}) = \lambda_k$.*

Proposition 12 allows us to construct an efficient algorithm (Algorithm 2) for computing the value of $\Pi(f \text{ is optimal})$ of a given element $f \in E$.

Algorithm 2 Computing $\Pi(f \text{ is optimal})$.

Require: A fuzzy weighted matroid (E, \mathcal{I}) , a distinguished element $f \in E$.

Ensure: $\Pi(f \text{ is optimal})$.

Find all elements e_1, \dots, e_m and numbers $\lambda_1, \dots, \lambda_m$ so that $\lambda_1 \leq \dots \leq \lambda_m$

Form $\sigma_0 = (u_1, \dots, u_q, f, e_1, \dots, e_m, v_1, \dots, v_r)$

$B \leftarrow \emptyset$

for $i \leftarrow 1$ **to** q **do**

if $B \cup \{u_i\} \in \mathcal{I}$ **then** $B \leftarrow B \cup \{u_i\}$

end for

if $B \cup \{f\} \notin \mathcal{I}$ **then return** 0 /* $f \notin B_{\sigma_0}$ */

for $i \leftarrow 1$ **to** m **do**

if $B \cup \{e_i\} \in \mathcal{I}$ **then** $B \leftarrow B \cup \{e_i\}$

if $B \cup \{f\} \notin \mathcal{I}$ **then return** λ_i /* $f \notin B_{\sigma_i}$ */

end for

return 1 /* $f \in B_{\sigma_m}$ */

The key of Algorithm 2 is that there is no need to apply Algorithm 1 to every sequence $\sigma_0, \dots, \sigma_m$ for checking whether f is a part of base B_{σ_i} . Using the fact that the sequences differ from each other only in the position of the element f , we only need to test whether f can be added to the base constructed in Algorithm 2 after choosing element e_i . It is easily seen that Algorithm 2 implicitly checks whether $f \in B_{\sigma_i}$ for all $i = 0, \dots, m$, deciding in this way if f is possibly optimal in the corresponding interval problem. Hence, Algorithm 2 is equivalent to one course of Algorithm 1. Since, in a typical situation, finding all the intersection points requires $\mathcal{O}(n)$ time (for instance if all the fuzzy intervals are of the L-L type [5]), it is easily seen that Algorithm 2 runs in $\mathcal{O}(n \log n + nf(n))$ time, where $\mathcal{O}(n \log n)$ is time required for sorting e_1, \dots, e_m with respect to the values of λ_i .

Let us now illustrate the computation of the degree of possible optimality by an example. Consider the maximum spanning tree problem shown in Fig. 3a. In this problem the element weights are specified as triangular fuzzy intervals. Recall that a triangular fuzzy interval (w, α, β) can be represented as a pair of gradual numbers of the form $[w - \alpha(1 - \lambda), w + \beta(1 - \lambda)]$, $\lambda \in [0, 1]$. For instance $\tilde{W}_{e_1} = (8, 3, 1)$ can be represented as $[8 - 3(1 - \lambda), 8 + (1 - \lambda)]$, $\lambda \in [0, 1]$. In Fig. 3b the fuzzy configuration $\tilde{\mathbf{w}}_{\{f\}}^+$ is shown. The partition of the edges is $\{e_1, e_2, e_3, e_4\} \cup \{u_1\} \cup \{v_1\} \cup \{f\}$. The gradual lower bounds of the weights of e_1, \dots, e_4 intersect with the gradual upper bound of the weight of f and the corresponding sequence of cuts is $1/3 \leq 1/2 \leq 3/5 \leq 5/7$.

We can now form the following sequences of the elements:

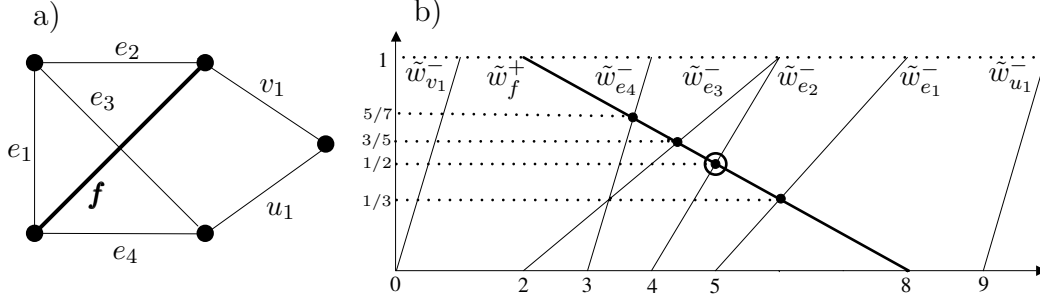


Fig. 3. a) The maximum spanning tree problem, where $\tilde{W}_{v_1} = (1, 1, 1)$, $\tilde{W}_f = (2, 2, 6)$, $\tilde{W}_{e_1} = (8, 3, 1)$, $\tilde{W}_{e_2} = (6, 2, 1)$, $\tilde{W}_{e_3} = (6, 4, 5)$, $\tilde{W}_{e_4} = (4, 1, 2)$, $\tilde{W}_{u_1} = (10, 1, 3)$. b) The fuzzy extreme configuration $\tilde{w}_{\{f\}}^+$.

$$\begin{aligned}
\sigma_0 &= (u_1, \mathbf{f}, e_1, e_2, e_3, e_4, v_1), \\
\sigma_1 &= (u_1, e_1, \mathbf{f}, e_2, e_3, e_4, v_1), \\
\sigma_2 &= (u_1, e_1, e_2, \mathbf{f}, e_3, e_4, v_1), \\
\sigma_3 &= (u_1, e_1, e_2, e_3, \mathbf{f}, e_4, v_1), \\
\sigma_4 &= (u_1, e_1, e_2, e_3, e_4, \mathbf{f}, v_1).
\end{aligned}$$

Algorithm 2 starts with forming the sequence σ_0 . It verifies then that $\{u_1, f\} \in \Phi$, $\{u_1, e_1, f\} \in \Phi$ and $\{u_1, e_1, e_2, f\} \notin \Phi$ because $\{e_1, e_2, f\}$ is a cycle. Hence σ_2 is the sequence with the smallest index such that $f \notin B_{\sigma_2}$. We thus conclude that $\Pi(f \text{ is optimal}) = \lambda_2 = 1/2$.

4.3.2 Computing the degree of necessary optimality

An approach to compute $N(f \text{ is optimal})$ for a given element $f \in E$ is symmetric. In this case one need to consider the fuzzy configuration $\tilde{w}_{\{f\}}^-$ (see Theorem 6) and take into account the intersection points of gradual weight \tilde{w}_f^- with gradual weights \tilde{w}_e^+ , $e \neq f$, in configuration $\tilde{w}_{\{f\}}^-$. Let e_1, \dots, e_m be the elements in E , whose gradual upper bounds intersect with the gradual lower bound of the element f in $\tilde{w}_{\{f\}}^-$. The numbers $\lambda_1, \dots, \lambda_m$ denote the cuts such that $\tilde{w}_{e_i}^+(\lambda_i) = \tilde{w}_f^-(\lambda_i)$, $i = 1, \dots, m$. We assume that $\lambda_1 \leq \dots \leq \lambda_m$, hence $\tilde{w}_{e_1}^+(\lambda_1) \leq \dots \leq \tilde{w}_{e_m}^+(\lambda_m)$. The partition of the elements is shown in Fig. 4.

We define the sequences of elements $\sigma_1, \dots, \sigma_{m+1}$ in the following way:

$$\begin{aligned}
\sigma_1 &= (u_1, \dots, u_q, e_m, \dots, e_1, \mathbf{f}, v_1, \dots, v_r), \\
\sigma_i &= (u_1, \dots, u_q, e_m, \dots, e_i, \mathbf{f}, e_{i-1}, \dots, e_1, v_1, \dots, v_r), \quad i = 2, \dots, m, \\
\sigma_{m+1} &= (u_1, \dots, u_q, \mathbf{f}, e_m, \dots, e_1, v_1, \dots, v_r).
\end{aligned}$$

Let us define $\lambda_{m+1} = 1$. The following proposition is symmetric to Proposition 12 (the proof goes in the similar manner).

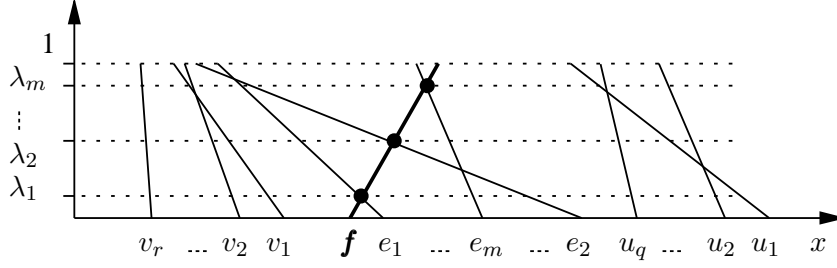


Fig. 4. The partition of E with respect to the intersection points of gradual weight \tilde{w}_f^- with gradual weights \tilde{w}_e^+ , $e \neq f$, in configuration $\tilde{w}_{\{f\}}^-$.

Proposition 13 *If $f \in B_{\sigma_1}$ then $N(f \text{ is optimal}) = 1$. Otherwise, let k be the largest index in $\{1, \dots, m+1\}$ such that $f \notin B_{\sigma_k}$. Then $N(f \text{ is optimal}) = 1 - \lambda_k$.*

Algorithm 3 Computing $N(f \text{ is optimal})$.

Require: A fuzzy weighted matroid (E, \mathcal{I}) , a distinguished element $f \in E$.

Ensure: $N(f \text{ is optimal})$.

Find all elements e_1, \dots, e_m and numbers $\lambda_1, \dots, \lambda_m$ so that $\lambda_1 \leq \dots \leq \lambda_m$

Form $\sigma_{m+1} = (u_1, \dots, u_q, f, e_m, \dots, e_1, v_1, \dots, v_r)$

$B \leftarrow \emptyset$

for $i \leftarrow 1$ **to** q **do**

if $B \cup \{u_i\} \in \mathcal{I}$ **then** $B \leftarrow B \cup \{u_i\}$

end for

if $B \cup \{f\} \notin \mathcal{I}$ **then return** 0 /* $f \notin B_{\sigma_{m+1}}$ */

for $i \leftarrow m$ **downto** 1 **do**

if $B \cup \{e_i\} \in \mathcal{I}$ **then** $B \leftarrow B \cup \{e_i\}$

if $B \cup \{f\} \notin \mathcal{I}$ **then return** $1 - \lambda_i$ /* $f \notin B_{\sigma_i}$ */

end for

return 1 /* $f \in B_{\sigma_1}$ */

Algorithm 3 is similar in a spirit to Algorithm 2. Here, there is also no need to apply Algorithm 1 to every sequence $\sigma_{m+1}, \dots, \sigma_1$ for checking whether f is a part of base B_{σ_i} , $i = m+1, \dots, 1$. Algorithm 3 implicitly checks whether $f \in B_{\sigma_i}$ for $i = m+1, \dots, 1$, evaluating in this way the necessary optimality of f . This is due to the fact that sequences $\sigma_{m+1}, \dots, \sigma_1$ differ from each other only in the position of element f . Obviously, computing $N(f \text{ is optimal})$ also requires $\mathcal{O}(n \log n + nf(n))$ time.

5 Conclusions

In this paper we have discussed the problem of evaluating the optimality of elements in a matroidal problem with ill-known weights. We have proposed some efficient methods for evaluating the possible and necessary optimality of

a given element in the interval-valued case. We have also considered the problem of evaluating the optimality of all elements for three particular interval-valued matroidal problems. We have then extended the obtained results to the fuzzy-valued matroidal problems exploiting the very recent notion of gradual number. The gradual numbers provide a new outlook on fuzzy intervals as the classical intervals of gradual numbers. This allows us to apply the interval methods to the problems with fuzzy weights. In consequence, we have designed two algorithms for computing efficiently the exact values of degrees of possible and necessary optimality of elements in the fuzzy-valued matroidal combinatorial optimization problems.

Acknowledgements

The second and the third author of this paper were partially supported by Polish Committee for Scientific Research, grant 3T11C05430.

References

- [1] I. D. Aron, P. van Hentenryck. A Constraint Satisfaction Approach to the Robust Spanning Tree with Interval Data. *Proceedings of the 18th Conference on Uncertainty in Artificial Intelligence*, Edmonton, Canada, 18–25 August 2002.
- [2] S. Chanas, P. Zieliński. The computational complexity of the criticality problems in a network with interval activity times. *European Journal of Operational Research*, 136: 541–550, 2002.
- [3] T. H. Cormen, C. E. Leiserson, R. L. Rivest. *Introduction to Algorithms*. MIT Press, 1994.
- [4] D. Dubois, H. Prade, Operations on fuzzy numbers. *International Journal of Systems Science*, 30:613–626, 1978.
- [5] D. Dubois, H. Prade. *Possibility theory: an approach to computerized processing of uncertainty*. Plenum Press, New York, 1988.
- [6] D. Dubois, H. Fargier, J. Fortin. A generalized vertex method for computing with fuzzy intervals. *Proceedings of the IEEE International Conference on Fuzzy Systems (Reno, Nevada)*, 541-546, 2004.
- [7] D. Dubois, H. Prade. Gradual Elements in a Fuzzy Set. *Soft Computing*. To appear.
- [8] J. Fortin, D. Dubois, H. Fargier. Gradual Numbers and their Application to Fuzzy Interval Analysis. *IEEE Transactions on Fuzzy Systems*. To appear.

- [9] T. Ibaraki, N. Katoh. *Resource Allocation Problems*. MIT Press, Cambridge, Massachusetts, 1988.
- [10] A. Kasperski, P. Zieliński. On combinatorial optimization problems on matroids with uncertain weights. *European Journal of Operational Research*, 177:851–864, 2007.
- [11] A. Kasperski, P. Zieliński. On the minmax regret approach and evaluating the optimality in combinatorial optimization problems with interval weights submitted to *European Journal of Operational Research*.
- [12] E. L. Lawler. *Combinatorial optimization: networks and matroids*. Holt, Rinehart and Winston, 1976.
- [13] H. Yaman, M. C. Pinar, O. E. Karasan. The robust spanning tree problem with interval data. *Operations Research Letters*, 29:31–40, 2001.
- [14] J. G. Oxley. *Matroid Theory*. Oxford University Press, New York, 1992.
- [15] C. Papadimitriou, K. Steiglitz. *Combinatorial optimization, algorithms and complexity*. Dover Publications Inc., Mineola, New York, 1998.
- [16] L. Zadeh. Fuzzy Sets. *Information and Control*, 8:338–353, 1965.