A possibilistic approach to bottleneck combinatorial optimization problems with Ill-known weights

Adam Kasperski[1] Paweł Zieliński[2]

1.Institute of Industrial Engineering and Management, Wrocław University of Technology
Wybrzeże Wyspiańskiego 27, 50-370 Wrocław, Poland
2.Institute of Mathematics and Computer Science, Wrocław University of Technology
Wybrzeże Wyspiańskiego 27, 50-370 Wrocław, Poland
Email: adam.kasperski@pwr.wroc.pl, pawel.zielinski@pwr.wroc.pl

Abstract— *In this paper a general bottleneck combinatorial optimization problem with uncertain element weights modeled by fuzzy intervals is considered. A rigorous possibilistic formalization of the problem and solution concepts in this setting that lead to finding robust solutions under fuzzy weights are given. Some algorithms for finding a solution according to the introduced concepts and evaluating optimality of solutions and elements are provided.*

Keywords— Bottleneck combinatorial optimization, Interval, Fuzzy interval, Fuzzy optimization and design, Possibility theory.

1 Introduction

A *combinatorial optimization problem* consists in finding an object composed of elements of a given ground set E. In a deterministic case, every element of E has a precise weight. In *bottleneck combinatorial optimization problems*, we wish to find an object that minimizes the weight of its heaviest element. Such formulation encompasses a large variety of classical combinatorial optimization problems, for instance: the bottleneck path [1], the bottleneck assignment [2], the bottleneck spanning tree [3] (or a more general the bottleneck matroid base problem [4]) etc. All these problems are efficiently solvable when the weights of all elements are precisely known. Unfortunately, in real world it is not easy to estimate element weights exactly. In many cases, the exact values of weights are not known in advance and the imprecision must be taken into account.

In this paper, we wish to investigate a general bottleneck combinatorial optimization problem with uncertain element weights modeled by fuzzy intervals. The membership function $\mu_{\widetilde{W}}$ of a fuzzy interval $\widetilde{W}$ is a possibility distribution describing, for each value w of the element weight, the extent to which it is a possible value. Equivalently, it means that the value of this weight belongs to a λ-cut interval $\widetilde{W}^{\lambda} = \{t : \mu_{\widetilde{W}}(t) \geq \lambda\}$ with confidence (or degree of necessity) $1 - \lambda$. Now to each feasible solution or element a *degree of possible optimality* and a *degree of necessary optimality* can be assigned. The notion of the necessary optimality of a solution may be weaken by assigning a *degree of necessary soft optimality*. Moreover, all the degrees of optimality of a solution (an element) can be derived from a *fuzzy deviation*, that is a possibility distribution representing the set of plausible values of deviations of the solution (the element) from optimum. In order to choose a "robust solution" under fuzzy weights, we adopt two criteria. The first one consists in choosing a solution of the maximum degree of necessary optimality, called a *best necessarily optimal solution*. Such a solution, if exists, seems to be an ideal choice. The second criterion is weaker than the first one and consists in choosing a solution of the maximum degree of necessary soft optimality, called a *best necessarily soft optimal solution*. This criterion has been originally proposed in [5, 6] for the linear programming problem with a fuzzy objective function. A best necessarily soft optimal solution seems to be a compromise choice and it minimizes a "distance" to the necessary optimality.

In this paper, we provide some methods for the optimality evaluation and for choosing a solution under fuzzy weights. In Section 3, we investigate the interval case, that is the problems in which the element weights are specified as closed intervals. It turns out, that it is possible to construct some polynomial algorithms for such problems if only their deterministic counterparts are polynomially solvable. In consequence, the interval bottleneck problems are easier to solve than the interval problems with a linear sum objective discussed in [7, 8]. We obtain polynomial algorithms for such problems as the bottleneck shortest path, the bottleneck assignment and the bottleneck matroid base. In Section 4, we show that the optimality evaluation and the problem of choosing a solution under fuzzy weights can be reduced to examining a family of interval problems. In particular, we show that a best necessarily soft optimal solution can be computed in polynomial time for a wide class of problems.

2 Preliminaries

In this section, we recall a formulation of a general bottleneck combinatorial problem with precise weights.

Let $E = \{e_1, \ldots, e_n\}$ be a finite ground set and let $\Phi \subseteq 2^E$ be a set of subsets of E called the set of the *feasible solutions*. A nonnegative real weight w_e is given for every element $e \in E$. A *bottleneck combinatorial optimization problem* $\mathcal{BP}$ consists in finding a feasible solution X that minimizes the weight of its heaviest element, namely:

$$\mathcal{BP}: \quad \min_{X \in \Phi} \max_{e \in X} w_e. \tag{1}$$

A solution that minimizes (1) is called an *optimal solution*. We call an element $e \in E$ *optimal* if it is a part of an optimal solution. If a solution (element) is not optimal a natural question arises how far from optimality this solution (element) is. To answer this question one can introduce the concept of a deviation. *A deviation of solution* $X \in \Phi$ and *a deviation of*

ISBN: 978-989-95079-6-8

element $f \in E$ are defined in the following way:

$$\delta_X = \max_{e \in X} w_e - \min_{Y \in \Phi} \max_{e \in Y} w_e,$$
$$\delta_f = \min_{Y \in \Phi_f} \max_{e \in Y} w_e - \min_{Y \in \Phi} \max_{e \in Y} w_e,$$

where Φ_f is the set of all feasible solutions that contain element f. It is clear that solution X (element f) is optimal if and only if $\delta_X = 0$ ($\delta_f = 0$).

In this paper we will assume that problem $\mathcal{BP}$ is polynomially solvable. Some polynomial algorithms for the bottleneck path, the bottleneck assignment, the bottleneck spanning tree, and the bottleneck matroid base problem can be found for instance in [1, 2, 3, 4]. For all these problems the deviations δ_X and δ_f can be computed in polynomial time.

3 Interval-valued bottleneck combinatorial optimization problems

In this section we consider an *interval* version of problem (1), in which the weights of the elements are ill-known and they are modeled by closed intervals $W_e = [\underline{w}_e, \overline{w}_e]$, $e \in E$. Assigning some interval to an element weight means that the actual weight of this element will take some value within the interval but it is not possible at present to predict which one. Every precise instantiation of the element weights is called a *scenario* and we denote it by S. Thus every scenario is a vector $S = (w_e)_{e \in E}$, $w_e \in W_e$ that expresses a realization of the weights. We denote by Γ the set of all the scenarios, i.e. $\Gamma = \times_{e \in E}[\underline{w}_e, \overline{w}_e]$ and we use $w_e(S)$ to denote the weight of element $e \in E$ in a fixed scenario $S \in \Gamma$. Among the scenarios of Γ, we distinguish the *extreme* ones, which belong to $\times_{e \in E}\{\underline{w}_e, \overline{w}_e\}$. Let $A \subseteq E$ be a fixed subset of elements. In scenario S_A^+ all elements $e \in A$ have weights $\overline{w}_e$ and all the remaining elements have weights $\underline{w}_e$. Similarly, in scenario S_A^- all elements $e \in A$ have weights $\underline{w}_e$ and all the remaining elements have weights $\overline{w}_e$. For a given solution $X \in \Phi$, we define its weight under a fixed scenario $S \in \Gamma$ as $F(X, S) = \max_{e \in X} w_e(S)$. We will denote by $F^*(S)$ the value of the weight of an optimal solution under scenario $S \in \Gamma$, that is $F^*(S) = \min_{X \in \Phi} F(X, S)$.

The optimality in the interval-valued problem $\mathcal{BP}$ can be characterized as follows: a given solution $X \in \Phi$ (element $f \in E$) is *possibly optimal* if and only if it is optimal in some scenario $S \in \Gamma$. A given solution $X \in \Phi$ (element $f \in E$) is *necessarily optimal* if and only if it is optimal in all scenarios $S \in \Gamma$. Similarly to the deterministic case, we can express the possible and necessary optimality in terms of the deviation. Let $\delta_X(S) = F(X, S) - F^*(S)$ and $\delta_f(S) = \min_{Y \in \Phi_f} F(Y, S) - F^*(S)$ denote the deviations of solution X and element f under a fixed scenario $S \in \Gamma$, respectively. Consider the following optimization problems:

$$\underline{\delta}_X = \min_{S \in \Gamma} \delta_X(S), \quad \overline{\delta}_X = \max_{S \in \Gamma} \delta_X(S)$$
$$\underline{\delta}_f = \min_{S \in \Gamma} \delta_f(S), \quad \overline{\delta}_f = \max_{S \in \Gamma} \delta_f(S).$$

The solutions to the above problems determine the so called *deviation interval* $\Delta_X = [\underline{\delta}_X, \overline{\delta}_X]$ containing all possible values of deviations for solution X. Similarly $\Delta_f = [\underline{\delta}_f, \overline{\delta}_f]$ is a deviation interval for element f. It is worth pointing that in literature (see e.g. [9]) the quantity $\overline{\delta}_X$ is called the *maximal regret* or *robust deviation* and it expresses the maximal possible deviation from optimum. Obviously, we can easily deduce the optimality of a solution from the deviation interval. Namely, X is possibly (resp. necessarily) optimal if and only if $\underline{\delta}_X = 0$ (resp. $\overline{\delta}_X = 0$). Similarly, an element f is possibly (resp. necessarily) optimal if and only if $\underline{\delta}_f = 0$ (resp. $\overline{\delta}_f = 0$).

3.1 Asserting possible and necessary optimality

In this section we establish some sufficient and necessary conditions for possible and necessary optimality of solutions and elements in the interval-valued problem. We start by proving the following proposition:

Proposition 1. *Let X be a given feasible solution. Then*

$$\underline{\delta}_X = \max\{0, \max_{e \in X} \underline{w}_e - F^*(S_E^+)\}, \tag{2}$$
$$\overline{\delta}_X = \max_{e \in X} \max\{0, \overline{w}_e - F^*(S_{\{e\}}^+)\}. \tag{3}$$

Proof. Equality (3) has been proved in [10]. We now give a proof sketch of (2). It is easy to verify that

$$\underline{\delta}_X \geq \max\{0, \max_{e \in X} \underline{w}_e - F^*(S_E^+)\}. \tag{4}$$

It remains to show that the inequality $\leq$ also holds in (4). Let Y be an optimal solution under S_E^+ and let $g = \arg\max_{e \in Y} \overline{w}_e$. We consider two cases. (i) $\max_{e \in X} \underline{w}_e > \overline{w}_g$. Denote $h = \arg\max_{e \in X} \underline{w}_e$. Consider scenario S such that $w_e^S = \min\{\underline{w}_h, \overline{w}_e\}$ for all $e \in X$ and $w_e^S = \overline{w}_e$ for all $e \in E \setminus X$. Since $\underline{w}_h \geq \underline{w}_e$ for all $e \in X$, $S \in \Gamma$. It is easy to check that $F(X, S) = \underline{w}_h$ and $F^*(S) = F(S_E^+)$. Hence $\underline{\delta}_X \leq \delta_X(S) = \max_{e \in X} \underline{w}_e - F^*(S_E^+) \leq \max\{0, \max_{e \in X} \underline{w}_e - F^*(S_E^+)\}$, which together with (4) yield (2). (ii) $\max_{e \in X} \underline{w}_e \leq \overline{w}_g$. Consider scenario S such that under this scenario all elements $e \in E \setminus X$ have weights $\overline{w}_e$ and all the elements $e \in X$ have weights $\min\{\overline{w}_e, \overline{w}_g\}$. Since $\underline{w}_e \leq \overline{w}_g$ for all $e \in X$, $S \in \Gamma$. One can easily verify that X is optimal under S, which means that $\underline{\delta}_X = 0 \leq \max\{0, \max_{e \in X} \underline{w}_e - F^*(S_E^+)\}$. This, together with (4), give (2). □

Proposition 1 allows us to compute the solution deviation interval Δ_X and leads to the following two corollaries:

Corollary 1. *A solution $X \in \Phi$ is possibly optimal if and only if $F(X, S_E^-) = \max_{e \in X} \underline{w}_e \leq F^*(S_E^+)$.*

Corollary 2. *A solution $X \in \Phi$ is necessarily optimal if and only if $\max_{e \in X} \max\{0, \overline{w}_e - F^*(S_{\{e\}}^+)\} = 0$.*

Making use of Corollaries 1 and 2, we can efficiently evaluate the optimality of a given solution X if only the underlying bottleneck deterministic problem $\mathcal{BP}$ is efficiently solvable. In order to evaluate the possible optimality of X, it suffices to compute the value of $F^*(S_E^+)$ and check if $F(X, S_E^-) \leq F^*(S_E^+)$. This can be done in $O(|X| + f(|E|))$ time, where $f(|E|)$ is the running time of an algorithm for the deterministic $\mathcal{BP}$ problem. Note that evaluating the possible optimality of a next solution, say $X' \in \Phi$, requires only $O(|X'|)$ time. Evaluating the necessary of optimality of X is a little more complex, since it requires computing the difference $\overline{w}_e - F^*(S_{\{e\}}^+)$ for each $e \in X$ and, consequently, the

ISBN: 978-989-95079-6-8

overall running time of evaluating the necessary optimality is $O(|X|f(|E|))$.

Let us consider the problem of computing an element deviation interval. The following proposition is true and its proof is similar to the proof of Proposition 1:

Proposition 2. *Let f be a specified element. Then*

$$\underline{\delta}_f = \max\{0, \min_{X\in\Phi_f} F(X, S^-_X) - F^*(S^+_E)\}.$$

From Proposition 2 we immediately get the following corollary:

Corollary 3. *An element $f \in E$ is possibly optimal if and only if $\min_{X\in\Phi_f} F(X, S^-_X) \le F^*(S^+_E)$.*

Corollary 3 shows a significant difference between the problems with the bottleneck objective and the problems with the linear sum objective discussed for instance in [7, 8]. For the latter problems, deciding whether $\underline{\delta}_f = 0$ for a given element $f \in E$ may be NP-hard even if a deterministic counterpart is polynomially solvable [8]. For the bottleneck problems the situation is much better. From Proposition 2, it follows that if problem $\mathcal{BP}$ is solvable in $f(|E|)$ time, then the bound $\underline{\delta}_f$ for a given element f can be determined in $O(f(|E|))$ time.

We are unable here to provide a general formula for computing the upper bound of an element deviation $\overline{\delta}_f$. Also, the complexity status of the problem of checking whether f is necessarily optimal is unknown. This is an interesting subject of further research. In the next section we show how to compute efficiently quantities $\underline{\delta}_f$ and $\overline{\delta}_f$ when $\mathcal{BP}$ has a matroidal structure.

3.1.1 Matroidal problems

Let us recall the notion of a matroid (see e.g. [11]). A *matroid* is a system $(E, \mathcal{I})$, where E is a ground set and $\mathcal{I}$ is a set of subsets of E closed under inclusions (if $A \subseteq B$ and $B \in \mathcal{I}$ then $A \in \mathcal{I}$) and fulfilling the so-called *growth* property (if $A, B \in \mathcal{I}$ and $|A| < |B|$ then there is $e \in B \setminus A$ such that $A \cup \{e\} \in \mathcal{I}$). The maximal (under inclusion) elements of $\mathcal{I}$ are called *bases*. We will denote the set of all bases by $\mathcal{B}$. The minimal (under inclusion) sets not in $\mathcal{I}$ are called *circuits*. Matroids have the following property, which will be used in this section. Namely, if $B \in \mathcal{B}$ is a base and f is an element such that $f \notin B$, then $B \cup \{f\}$ contains the unique circuit C. Furthermore, for every $e \in C$, set $(B \cup \{f\}) \setminus \{e\}$ is a base.

In a *matroidal problem* the set of feasible solutions Φ consists of all bases of a given matroid, $\Phi = \mathcal{B}$. An example of a matroidal problem is bottleneck spanning tree, where Φ consists of all bases of a *graphic matroid* [11]. Due to a matroidal structure of the interval problem $\mathcal{BP}$, we can simplify the computation of $\underline{\delta}_f$ and provide a method of computing $\overline{\delta}_f$. We first show how to evaluate the possible and necessary optimality of a given element f.

Proposition 3. *An element $f \in E$ is possibly optimal if and only if $\underline{w}_f \le F^*(S^+_E)$.*

Proof. Using Corollary 3, it is sufficient to proof that $\underline{w}_f \le F^*(S^+_E)$ if and only if there is a base $B \in \mathcal{B}_f$ such that $F(B, S^-_E) \le F^*(S^+_E)$, where $\mathcal{B}_f$ stands for the set of all bases that contain element f.

($\Leftarrow$) Obvious.
($\Rightarrow$) Let B^* be an optimal base in scenario S^+_E. If $f \in B^*$, then $B^* \in \mathcal{B}_f$, $F(B^*, S^-_E) \le F(B^*, S^+_E) = F^*(S^+_E)$ and we are done. Otherwise, $B^* \cup \{f\}$ contains an unique circuit C. Set $B' = (B^* \setminus \{e\}) \cup \{f\}$, $e \in C \setminus \{f\}$, is a base and $B' \in \mathcal{B}_f$. Since $F^*(S^+_E) \ge \underline{w}_f$, we have $F^*(S^+_E) = \max_{e\in B^*} \overline{w}_e \ge \max_{e\in B'} \underline{w}_e$. Thus $F^*(S^+_E) \ge F(B', S^-_E)$ and the proposition follows. □

Proposition 4. *An element $f \in E$ is necessarily optimal if and only if $\overline{w}_f \le F^*(S^+_{\{f\}})$.*

Proof. ($\Rightarrow$) If f is necessarily optimal, then it is optimal under all scenarios, in particular, under $S^+_{\{f\}}$. Thus there is a base B containing f such that $F^*(S^+_{\{f\}}) = F(B, S^+_{\{f\}})$, which yields $F^*(S^+_{\{f\}}) \ge \overline{w}_f$.
($\Leftarrow$) Consider any scenario $S \in \Gamma$. We first show that inequality $w_f(S) \le F^*(S)$ holds for S. Define scenario S' where $w_f(S') = \overline{w}_f$ and the remaining elements, $e \ne f$, have weights $w_e(S') = w_e(S)$. Using the assumption we get $\overline{w}_f \le F^*(S^+_{\{f\}}) \le F^*(S')$. After decreasing the weight of f to $w_f(S)$, we get inequality $w_f(S) \le F^*(S)$. Let B^* be an optimal base under S. If $f \in B^*$ then f is optimal under S. Otherwise, $B^* \cup \{f\}$ contains an unique circuit C. Set $B' = (B^* \setminus \{e\}) \cup \{f\}$, $e \in C \setminus \{f\}$, is a base such that $B' \in \mathcal{B}_f$. Since $w_f(S) \le F^*(S)$, $F(B', S) \le F(B^*, S) = F^*(S)$. Hence B' is an optimal base and f is optimal under S. □

Using a similar reasoning to that in Propositions 3 and 4 one can prove the following proposition, which allows us to determine the deviation interval $\Delta_f = [\underline{\delta}_f, \overline{\delta}_f]$.

Proposition 5. *Let X be a given feasible solution. Then*

$$\underline{\delta}_f = \max\{0, \underline{w}_f - F^*(S^+_E)\}, \overline{\delta}_f = \max\{0, \overline{w}_f - F^*(S^+_{\{f\}})\}.$$

Propositions 3 and 4 allow us to evaluate efficiently the possible (resp. necessary) optimality of a specified element $f \in E$. It suffices to replace the interval weights with their exact values determined according to S^+_E (resp. $S^+_{\{f\}}$), compute the optimal value of $F^*(S^+_E)$ (resp. $F(S^+_{\{f\}})$) in the resulting deterministic problem and compare it with $\underline{w}_f$ (resp. $\overline{w}_f$). This can be done in $O(|E|\log^*(|E|))$ time [4], where $\log^*|E|$ is the iterated logarithm of $|E|$. Note also that using Proposition 3 we can detect all possibly optimal elements in $O(|E|\log^*(|E|))$ because we need to execute an algorithm for the deterministic problem only once. On the other hand, Proposition 4 does not allow us to detect all necessary optimal elements without extra effort. Computing the deviation interval Δ_f costs the same time as evaluating the possible and necessary optimality of f (see Proposition 5).

3.2 Choosing a robust solution

An important task in the interval-valued problem is to choose a *robust solution*, that is the one which performs reasonably well under any possible scenario. A necessarily optimal solution ($\overline{\delta}_X = 0$) is an ideal choice because it is optimal regardless of weight realizations. On the other hand, the possible optimality of a chosen solution ($\underline{\delta}_X = 0$) is the minimum requirement that should be satisfied. A possibly optimal solution

ISBN: 978-989-95079-6-8

always exists. But a necessarily optimal solution rarely exists. Hence the possible optimality is too weak criterion while the necessary optimality seems to be too strong. A solution that minimizes the maximal regret $\overline{\delta}_X$ seems to be a compromise choice. It minimizes the maximal possible deviation from optimum. In literature [9] such a solution is called an *optimal minmax regret solution*.

Proposition 6. *Every optimal minmax regret solution X is possibly optimal and is composed of possibly optimal elements.*

Proof. We use a proof by contraposition. Assume that X is not possibly optimal. From Corollary 1, we have $\max_{e\in X} \underline{w}_e > F^*(S_E^+)$. Let Y^* be an optimal solution in scenario S_E^+. Define $g = \arg\max_{e\in Y^*} \overline{w}_e$ and $h = \arg\max_{e\in X} \underline{w}_e$. Thus $w_h(S) > \overline{w}_g$ for all scenarios $S \in \Gamma$. But $\overline{w}_g \geq w_e(S)$ for all $e \in Y^*$ in every scenario S, which yields $w_h(S) > w_e(S)$ for $e \in Y^*$. From this we conclude that $F(X,S) > F(Y^*,S)$ for all $S \in \Gamma$, which implies $\overline{\delta}_X > \overline{\delta}_Y$. In consequence, X cannot be an optimal minmax regret solution. It is obvious that every possibly optimal solution is composed of possibly optimal elements. □

Therefore, any optimal minmax regret solution X is possibly optimal and it minimizes a distance to the necessary optimality. In other words its deviation interval is of the form $\Delta_X = [0, \overline{\delta}_X]$ where $\overline{\delta}_X$ is the smallest among all $X \in \Phi$. Fortunately, the problem of determining an optimal minmax regret solution can be efficiently solved if the underlying bottleneck deterministic problem $\mathcal{BP}$ is polynomially solvable [10]. Using Equality (3) we get

$$\min_{X\in\Phi} \overline{\delta}_X = \min_{X\in\Phi} \max_{e\in X} \hat{w}_e, \tag{5}$$

where weights $\hat{w}_e = \max\{0, \overline{w}_e - F^*(S_{\{e\}}^+)\}$, $e \in E$, are deterministic. So, the minmax regret problem can be reduced to a deterministic problem $\mathcal{BP}$. It can be shown [10] that the minmax regret problem can be solved in $O(|E| + |X^*| f(|E|))$ time, where X^* is such that $F(X^*, S_E^-) = F^*(S_E^-)$ and $f(|E|)$ is the running time of an algorithm for $\mathcal{BP}$.

4 Fuzzy-valued bottleneck combinatorial optimization problems

In this section, we study problem $\mathcal{BP}$ with uncertain element weights, where the uncertainty is modeled by fuzzy intervals. We give a rigorous possibilistic interpretation, solution concepts and some solution algorithms in this setting.

4.1 A possibilistic formalization of the problem

We now give a formalization of problem $\mathcal{BP}$, in which the weights of elements of E are uncertain and they are modeled by fuzzy intervals $\widetilde{W}_e$, $e \in E$, in the setting of *possibility theory* [12]. Let us recall that a *fuzzy interval* $\widetilde{W}_e$ is a fuzzy set in the space of real numbers $\mathbb{R}$, whose membership function $\mu_{\widetilde{W}_e} : \mathbb{R} \to [0,1]$ is normal, quasiconcave and upper semicontinuous on $\mathbb{R}$. We will additionally assume that the support of $\widetilde{W}_e$ is bounded. A membership function of $\widetilde{W}_e$ is regarded as a *possibility distribution* for the values of the unknown weight $\mathfrak{w}_e$. The possibility degree of the assignment $\mathfrak{w}_e = v$ is $\Pi(\mathfrak{w}_e = v) = \pi_{\mathfrak{w}_e}(v) = \mu_{\widetilde{W}_e}(v)$. Let $S = (v_e)_{e\in E}$ be a scenario that represents a state of the world in which $\mathfrak{w}_e = v_e$, for all $e \in E$. Assuming that the weights are unrelated to one another, the fuzzy intervals associated with the weights induce the following possibility distribution over all scenarios in $\mathbb{R}^n$ (see [13]): $\pi(S) = \min_{e\in E} \Pi(\mathfrak{w}_e = v_e) = \min_{e\in E} \mu_{\widetilde{W}_e}(v_e)$ The degrees of possibility and necessity that a solution $X \in \Phi$ is optimal are defined as follows:

$$\begin{aligned} \Pi(X \text{ is optimal}) &= \sup_{\{S:\, X \text{ is optimal in } S\}} \pi(S), \\ \mathrm{N}(X \text{ is optimal}) &= \inf_{\{S:\, X \text{ is not optimal in } S\}} (1 - \pi(S)). \end{aligned} \tag{6}$$

As in the deterministic and interval cases we can characterize the optimality of a solution using the concept of deviation. In the fuzzy problem a solution deviation belongs to a fuzzy interval $\tilde{\Delta}_X$, whose membership function $\mu_{\tilde{\Delta}_X}$ is a possibility distribution for δ_X defined as follows:

$$\mu_{\widetilde{\Delta}_X}(v) = \Pi(\delta_X = v) = \sup_{\{S:\, \delta_X(S) = v\}} \pi(S). \tag{7}$$

Since the statement "X is optimal in S" is equivalent to condition $\delta_X(S) = 0$, it is easily seen that (7) generalizes the degrees of optimality (6), that is

$$\begin{aligned} \Pi(X \text{ is optimal}) &= \Pi(\delta_X = 0) = \mu_{\widetilde{\Delta}_X}(0), \\ \mathrm{N}(X \text{ is optimal}) &= \mathrm{N}(\delta_X = 0) = 1 - \sup_{v>0} \mu_{\widetilde{\Delta}_X}(v). \end{aligned} \tag{8}$$

Replacing X with f in (6)-(8) we get the same formulae for element f.

In practice the requirement $\delta_X = 0$ may be very strong and the degree of necessary optimality of every solution $X \in \Phi$ may be very small or even equal to 0. Suppose that a decision maker knows his/her preferences about δ_X and expresses it by a *fuzzy goal* $\widetilde{G}$. The membership function of the fuzzy goal $\mu_{\widetilde{G}}$ is a nonincreasing mapping from $[0,\infty]$ into $[0,1]$ such that $\mu_{\widetilde{G}}(0) = 1$. The value of $\mu_{\widetilde{G}}(\delta_X)$ expresses the degree to which deviation δ_X satisfies the decision maker. We can now replace the strong requirement "X is optimal" ($\delta_X = 0$) with weaker "X is soft optimal" ($\delta_X \in \widetilde{G}$). Notice that $\delta_X \in \widetilde{G}$ is a fuzzy event and the necessity that it holds can be computed as follows [5, 6]:

$$\mathrm{N}(X \text{ is soft optimal}) = \inf_S \max\{1 - \pi(S), \mu_{\widetilde{G}}(\delta_X(S))\}. \tag{9}$$

One can check that $\mathrm{N}(X \text{ is soft optimal}) = \alpha$ means that for all scenarios S such that $\pi(S) > 1 - \alpha$ it holds $\mu_{\widetilde{G}}(\delta_X(S)) \geq \alpha$ or equivalently $\delta_X(S) \in \widetilde{G}^\alpha = [0, \mu_{\widetilde{G}}^{-1}(\alpha)]$, which represents the suboptimality of X. Function $\mu_{\widetilde{G}}^{-1} : [0,1] \to \mathbb{R} \cup \{+\infty\}$ is a pseudo-inverse of $\mu_{\widetilde{G}}$ that is $\mu_{\widetilde{G}}^{-1}(\alpha) = \sup\{v : \mu_{\widetilde{G}}(v) \geq \alpha\}$.

4.2 Computing the optimality degrees

Every fuzzy interval $\widetilde{U}$ can be decomposed into its λ-cuts, i.e. sets $\widetilde{U}^\lambda = \{u : \mu_{\widetilde{U}}(u) \geq \lambda\}$, $\lambda \in (0,1]$. We will assume that $\widetilde{U}^0$ is the smallest closed set containing the support of $\widetilde{U}$. It can be easily verified (see e.g. [12]) that if $\widetilde{U}$ is a fuzzy interval with a bounded support, then $\widetilde{U}^\lambda$ is a closed interval for every $\lambda \in [0,1]$. The membership function $\mu_{\widetilde{U}}$ can be retrieved from the family of λ-cuts in the following way:

$$\mu_{\widetilde{U}}(u) = \sup\{\lambda \in [0,1] : u \in \widetilde{U}^\lambda\}, \tag{10}$$

ISBN: 978-989-95079-6-8

and $\mu_{\widetilde{U}}(u) = 0$ if $v \notin \widetilde{U}^0$.

Let us denote by $\mathcal{BP}^\lambda$, $\lambda \in [0,1]$, the interval-valued problem $\mathcal{BP}$ with element weights $\widetilde{W}_e^\lambda = [\underline{w}_e^\lambda, \overline{w}_e^\lambda]$, $e \in E$. Note that the scenario set in $\mathcal{BP}^\lambda$ is composed of all scenarios S such that $\pi(S) \geq \lambda$. Now $\widetilde{\Delta}_X^\lambda = [\underline{\delta}_X^\lambda, \overline{\delta}_X^\lambda]$ is the interval of possible values of the deviation of solution X in problem $\mathcal{BP}^\lambda$. It contains all values of the deviation of X, whose possibility of occurrence is not less than λ. From (8) and (10), it follows easily that

$$\Pi(X \text{ is optimal}) = \sup\{\lambda \in [0,1] : \underline{\delta}_X^\lambda = 0\} \quad (11)$$

and $\Pi(X \text{ is optimal}) = 0$ if $\underline{\delta}_X^0 > 0$. A similar reasoning leads to the following equality:

$$\mathrm{N}(X \text{ is soft optimal}) = 1 - \inf\{\lambda : \overline{\delta}_X^\lambda \leq \mu_{\widetilde{G}}^{-1}(1-\lambda)\}. \quad (12)$$

Replacing expression $\mu_{\widetilde{G}}^{-1}(1-\lambda)$ with 0 in (12) we get a formula for computing the degree of necessary optimality of X. Exactly the same considerations apply to elements. It is enough to replace X with f in formulae (11)–(12).

Equations (11) and (12) form a theoretical basis for calculating the values of the optimality degrees. The function $\underline{\delta}_X^\lambda$ is nondecreasing, so in order to compute the degree of possible optimality of solution X we can apply a binary search technique on $\lambda \in [0,1]$. Condition $\underline{\delta}_X^\lambda = 0$ means that X is possibly optimal in $\mathcal{BP}^\lambda$ and it can be checked efficiently by using Corollary 1. Similarly, $\overline{\delta}_X^\lambda$ is nonincreasing and $\mu_{\widetilde{G}}^{-1}(1-\lambda)$ is nondecreasing function of λ. Hence the degree of necessary soft optimality (and necessary optimality) can also be computed by using a binary search. We obtain $\overline{\delta}_X$ using Corollary 2. The calculations can be done in $\mathcal{O}(I(|E|)\log\epsilon^{-1})$ time, where $\epsilon > 0$ is a given precision and $I(|E|)$ is time required to find the bound (the lower or the upper) of deviation interval $\widetilde{\Delta}_X^\lambda$ in the corresponding interval problem $\mathcal{BP}^\lambda$. Of course the same reasoning can be repeated for elements.

Fuzzy deviations $\widetilde{\Delta}_X$ and $\widetilde{\Delta}_f$ can also be determined via the use of λ-cuts. That is, we compute interval $\widetilde{\Delta}_X^\lambda = [\underline{\delta}_X^\lambda, \overline{\delta}_X^\lambda]$ (resp. $\widetilde{\Delta}_f^\lambda$) in the interval valued $\mathcal{BP}^\lambda$ for suitably chosen λ-cuts. Then the fuzzy quantity $\widetilde{\Delta}_X$ (resp. $\widetilde{\Delta}_f$) is reconstructed from its λ-cuts by using equality (10). This method gives an approximation of $\widetilde{\Delta}_X$ (resp. $\widetilde{\Delta}_f$). Its overall running time is $\mathcal{O}(rI(|E|))$ time, r is the number of chosen λ-cuts and $I(|E|)$ is time required to determine $\widetilde{\Delta}_X^\lambda$ (resp. $\widetilde{\Delta}_f^\lambda$) in the interval problem $\mathcal{BP}^\lambda$.

From the above it follows that the running time of the proposed methods heavily relies on the interval case, because they can be reduced to the optimality analysis in a sequence of the interval-valued problems $\mathcal{BP}^\lambda$. Thus the main difficulty lies in the interval case. Fortunately, the optimality analysis in the interval case can be efficiently done, see Corollaries 1,2 and Proposition 1 for a solution and Corollary 3 and Proposition 2 for an element. Unfortunately, we cannot give an efficient method for asserting necessary optimality and determining the upper bound on possible values of the deviation of an element in a general problem $\mathcal{BP}^\lambda$. However, we have proposed such method when problem $\mathcal{BP}^\lambda$ has a matroidal structure (see Section 3.1.1).

4.3 Choosing a robust solution

We now propose some concepts of choosing a robust solution in the fuzzy-valued problem $\mathcal{BP}$. An ideal choice is a solution with the highest degree of certainty that it is optimal under all possible scenarios, i.e. an optimal solution to the following problem:

$$\max_{X\in\Phi} \mathrm{N}(X \text{ is optimal}) = \max_{X\in\Phi} \mathrm{N}(\delta_X = 0). \quad (13)$$

A solution of (13) is called a *best necessarily optimal solution*. However, this concept has a drawback, since the criterion used in (13) is very strong. Namely, a solution X such that $\mathrm{N}(X \text{ is optimal}) > 0$ may not exist or even if it exists, its necessary optimality degree may be very small. To overcome this drawback, we can replace the degree of necessary optimality with the degree of soft necessary optimality, which leads to the following problem:

$$\max_{X\in\Phi} \mathrm{N}(X \text{ is soft optimal}). \quad (14)$$

A solution to (14) is called a *best necessarily soft optimal solution*. Let us recall that this solution concept was first applied to fuzzy linear programming in [5, 6]. We can see from (12) that problem (14) is equivalent to the following one:

$$\begin{array}{ll} \min & \lambda \\ s.t. & \overline{\delta}_X^\lambda \leq \mu_{\widetilde{G}}^{-1}(1-\lambda), \\ & \lambda \in [0,1], \\ & X \in \Phi. \end{array} \quad (15)$$

If problem (15) is infeasible then $\mathrm{N}(X \text{ is soft optimal}) = 0$ for all solutions $X \in \Phi$. If λ^* is the optimal objective value of (15) and X is a best necessarily soft optimal solution, then $\mathrm{N}(X \text{ is soft optimal}) = 1 - \lambda^*$. Since $\overline{\delta}_X^\lambda$ is nonincreasing and $\mu_{\widetilde{G}}^{-1}(1-\lambda)$ is nondecreasing function of λ, problem (15) can also be solved by the binary search technique on $\lambda \in [0,1]$. Note that $\overline{\delta}_X^\lambda$ is the maximal regret of X in $\mathcal{BP}^\lambda$ (see Section 3.2). Thus to find the optimal λ^*, we seek solution that minimizes $\overline{\delta}_X^\lambda$ for a fixed λ, i.e. an optimal minmax regret solution in $\mathcal{BP}^\lambda$ (see problem (5)). An optimal solution X^* to problem (15), determined by finding an optimal minmax regret solution at λ-cut, has a nice property. Namely, it is an optimal minmax regret solution in $\mathcal{BP}^{\lambda^*}$ and, by Proposition 6, it is also possibly optimal one in $\mathcal{BP}^{\lambda^*}$. Thus $\mathrm{N}(X^* \text{ is soft optimal}) \leq \Pi(X^* \text{ is optimal})$. Consequently,

$$\begin{array}{rcl} \mathrm{N}(X^* \text{ is optimal}) & \leq & \mathrm{N}(X^* \text{ is soft optimal}) \\ & \leq & \Pi(X^* \text{ is optimal}). \end{array}$$

From this, we conclude that the concept of a necessary soft optimality is a natural extension of the minmax regret approach to the fuzzy case. The running time of the method based on the binary search is $\mathcal{O}(I(|E|)\log\epsilon^{-1})$ time, where $\epsilon > 0$ is a given precision and $I(|E|)$ is time required for seeking an optimal minmax regret solution in $\mathcal{BP}^\lambda$. It is polynomial for the bottleneck path, the bottleneck assignment, the bottleneck spanning tree and the bottleneck matroid base problems.

ISBN: 978-989-95079-6-8

4.4 A parametric approach

We now present a parametric approach to finding a best necessary optimal solution. A similar method can be applied to compute fuzzy deviations $\widetilde{\Delta}_X$ and $\widetilde{\Delta}_f$. It is easy to check that the following parametric problem may be associated with problem (15) (see also problems (15) and (5))

$$\overline{\delta}^\lambda = \min_{X\in\Phi} \overline{\delta}^\lambda_X = \min_{X\in\Phi} \max_{e\in X}\{\hat{w}^\lambda_e\}, \;\; \lambda \in [0, 1], \qquad (16)$$

where $\hat{w}^\lambda_e = \max\{0, \overline{w}_e - F^*(S^{+\lambda}_{\{e\}})\}$, $e \in E$, are parametric weights (functions of a parameter λ), $S^{+\lambda}_{\{e\}}$ is the scenario in which we fix the weight of element e to $\overline{w}^\lambda_e$ and the weights of elements $g \in E \setminus \{e\}$ to $\underline{w}^\lambda_g$. In order to solve (16), we need to compute parametric weights $\hat{w}^\lambda_e = \max\{0, \overline{w}_e - F^*(S^{+\lambda}_{\{e\}})\}$, $e \in E$. Determining these weights requires computing functions $F^*(S^{+\lambda}_{\{e\}})$ for each $e \in E$. This can be done by applying known parametric techniques. Observe that $F^*(S^{+\lambda}_{\{e\}})$ is a function of parameter $\lambda \in [0, 1]$. We wish to compute sequences $0 = \lambda_0 < \lambda_1 < ... < \lambda_k = 1$ and $X_0, \ldots, X_{k-1}$ such that X_i is an optimal solution for $\lambda \in [\lambda_i, \lambda_{i+1}]$, $i = 0, \ldots, k-1$. Having these sequences it is easy to describe function $F^*(S^{+\lambda}_{\{e\}})$ and, in consequence, $\hat{w}^\lambda_e$ for $\lambda \in [0, 1]$. It turns out that if elements weights are linear functions of λ for each $e \in E$, then for some particular bottleneck problems their parametric counterparts can be efficiently solved (see e.g. [14]). Now, we are ready to solve (16) with parametric weights $\hat{w}^\lambda_e$. Again, we compute sequences $0 = \lambda_0 < \lambda_1 < ... < \lambda_k = 1$ and optimal solutions (optimal minmax regret solutions) $X_0, \ldots, X_{k-1}$ and the upper bounds for their deviations $\overline{\delta}^\lambda_{X_i}$, $\lambda \in [\lambda_i, \lambda_{i+1}]$, $i = 0, \ldots, k-1$. Values $\overline{\delta}^\lambda_{X_i}$, $\lambda \in [\lambda_i, \lambda_{i+1}]$, $i = 0, \ldots, k-1$, provide an analytical description of function $\overline{\delta}^\lambda$ (deviations of optimal minmax regret solutions) for $\lambda \in [0, 1]$. To find a best necessary optimal solution, we need solve the following problem

$$\lambda^* = \arg \min_{\lambda\in[0,1]} \{\overline{\delta}^\lambda, \mu^{-1}_{\widetilde{G}}(1 - \lambda)\},$$

which is equivalent to finding an intersection of $\overline{\delta}^\lambda$ and $\mu^{-1}_{\widetilde{G}}(1 - \lambda)$. An optimal minmax regret solution that corresponds to the optimal value λ^* is a best necessary optimal solution.

5 Conclusions

In this paper, we have studied a general bottleneck combinatorial optimization problem with uncertain element weights modeled by fuzzy intervals, whose membership functions are regarded as possibility distributions for the values of the unknown weights. We have described, in this setting, the notions of possible and necessary optimality of a solution and an element and the necessary soft optimality of a solution. These notions are natural generalizations of the ones introduced in the interval-valued case. In order to choose a robust solution, we compute a best necessary soft optimal solution. This concept of choosing a solution is also a generalization, to the fuzzy case, of the known from literature minmax regret criterion [9]. Thus, we have shown in the paper that there exists a link between interval and possibilistic uncertainty. Hence, we have discussed first the interval-valued case and then we have extended the notions and the methods introduced for the interval-valued problem to the fuzzy-valued one. Indeed, the optimality evaluation and choosing a robust solution in the fuzzy problem boil down to solving a number of interval problems $\mathcal{BP}^\lambda$. Both problems can be solved in polynomial time if the corresponding interval counterparts are polynomially solvable, that holds true for a wide class of classical bottleneck combinatorial problems.

Acknowledgment

The first author was partially supported by Polish Committee for Scientific Research, grant N N111 1464 33.

References

[1] Abraham P. Punnen. A linear time algorithm for the maximum capacity path problem. *European Journal of Operational Research*, 53:402–404, 1991.

[2] Harold N. Gabow and Robert E. Tarjan. Algorithms for two bottleneck optimization problems. *Journal of Algorithms*, 9:411–417, 1988.

[3] Paolo M. Camerini. The minimax spanning tree problem and some extensions. *Information Processing Letters*, 7:10–14, 1978.

[4] Abraham P. Punnen. A fast algorithm for a class of bottleneck problems. *Computing*, 56:397–401, 1996.

[5] Masahiro Inuiguchi and Masatoshi Sakawa. Robust optimization under softness in a fuzzy linear programming problem. *International Journal of Approximate Reasonning*, 18:21–34, 1998.

[6] Masahiro Inuiguchi. On possibilistic/fuzzy optimization. In *Foundations of Fuzzy Logic and Soft Computing, IFSA 2007*, Lecture Notes in Artificial Intelligence 4529, pages 351–360. Springer-Verlag, 2007.

[7] Adam Kasperski and Paweł Zieliński. On possibilistic combinatorial optimization problems. In *Proceedings of the IEEE International Conference on Fuzzy Systems*, pages 324–329, Hong Kong, 2008.

[8] Adam Kasperski and Paweł Zieliński. Minmax regret approach and optimality evaluation in combinatorial optimization problems with interval and fuzzy weights. *European Journal of Operational Research*, to appear.

[9] Panos Kouvelis and Gang Yu. *Robust Discrete Optimization and its applications*. Kluwer Academic Publishers, 1997.

[10] Igor Averbakh. Minmax regret solutions for minimax optimization problems with uncertainty. *Operations Research. Letters*, 27:57–65, 2000.

[11] C. Papadimitriou and K. Steiglitz. *Combinatorial Optimization*. Dover Publications, INC, Mineola, New York, 1998.

[12] Didier Dubois and Henri Prade. *Possibility theory: an approach to computerized processing of uncertainty*. Plenum Press, New York, 1988.

[13] Didier Dubois, Hélène Fargier, and Vincent Galvagnon. On latest starting times and floats in activity networks with ill-known durations. *European Journal of Operational Research*, 147:266–280, 2003.

[14] Timothy M. Chan. Finding the shortest bottleneck edge in a parametric minimum spanning tree. In *Proceedings of the 16th Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 232—-240, 2005.

ISBN: 978-989-95079-6-8