Solving sequential collective decision problems under qualitative uncertainty ☆

Nahla Ben Amor [a], Fatma Essghaier [a,b], Hélène Fargier [b]

[a] LARODEC, University of Tunis, Tunisia
[b] IRIT, UPS-CNRS, 118 route de Narbonne, 31062 Toulouse, France

Keywords:
Possibility theory
Qualitative uncertainty
Collective decision making
Sequential decision making
Decision tree
Dynamic Programming

A B S T R A C T

This paper addresses the question of sequential collective decision making under qualitative uncertainty. It resumes the criteria introduced in previous works [4–6] by Ben Amor et al. and extends them to a more general context where every decision maker is free to have an optimistic or a pessimistic attitude w.r.t. uncertainty. These criteria are then considered for the optimization of possibilistic decision trees and an algorithmic study is performed for each of them. When the global utility does satisfy the monotonicity property, a classical possibilistic Dynamic Programming can be applied. Otherwise, two cases are possible: either the criterion is max oriented (the more is the satisfaction of any agent, the greater is the global satisfaction), and a dedicated algorithm can be proposed, that relies on as many calls to Dynamic Programming as the number of decision makers; or the criterion is min oriented (all the agents must like the common decision) and the optimal strategy can be provided by a Branch and Bound Algorithm. The paper concludes by an experimental study that shows the feasibility of the approaches, and details to what extent simple Dynamic programming algorithms can be used as approximation procedures for the non monotonic criteria.

1. Introduction

The handling of a collective decision problem under uncertainty resorts on (i) the identification of a theory of decision making under uncertainty (DMU) that captures the decision makers' behavior with respect to uncertainty and (ii) the specification of a collective utility function (CUF) as it may be used when the problem is not pervaded with uncertainty. But also, one needs to precise when the utility of the agents is to be evaluated: before (*ex-ante*) or after (*ex-post*) the realization of the uncertain events. In the first case, the global utility function is a function of the DMU utilities of the different agents; in the second case it is an aggregation, w.r.t. the likelihood of the final states, of the collective utilities.

Following Fleming [23], Harsanyi [27] has shown that, when the uncertainty about consequences of decisions can be quantified in a probabilistic way: the collective utility should be a weighted sum of the individual expected utilities. Many contributions have been inspired by this seminal work: Some authors (such as Diamond [11]) criticized this approach because not applicable when the collective utility is more egalitarian than utilitarian. Others have developed Harsanyi's ap-

☆ This paper is an extended version of preliminary results presented in [7]; it includes the full proofs of the propositions, and new models and algorithms that allow the modeling of collective sequential problems where the agents have different attitudes w.r.t. uncertainty.

E-mail addresses: nahla.benamor@gmx.fr (N. Ben Amor), essghaier.fatma@gmail.com (F. Essghaier), fargier@irit.fr (H. Fargier).

https://doi.org/10.1016/j.ijar.2019.03.003

proach, in particular Myerson [37] who proved that only the choice of an utilitarian social welfare function can reconciliate the *ex-ante* and *ex-post* approaches. However in the probabilistic case, all other welfare functions suffer from the "timing effect" [37], i.e., lead to a discrepancy between the *ex-ante* and the *ex-post* approach.

Harsanyi's and Meyrson's results rely on the assumption that the knowledge of the agents about the consequences of their decisions is rich enough to be modeled by probabilistic lotteries. When the information about uncertainty cannot be quantified in a probabilistic way, the topic of possibilistic decision theory is often a natural one to consider [12–14,18,21,25]. Qualitative decision theory is relevant, among other fields, for applications to planning under uncertainty, where a suitable *strategy* (i.e., a set of conditional or unconditional decisions) is to be found, starting from a qualitative description of the initial world, of the available alternatives, of their (perhaps uncertain) effects and of the goal to reach (see [8,41,43]). But up to this point, the evaluation of the strategies was considered only in a simple, mono-agent context, while it is often the case that several agents are involved in the decision.

The present paper raises the question of sequential collective decision making under possibilistic uncertainty. It follows recent works [4–6] which propose a theoretical framework for multi-agent (non sequential) decision making under possibilistic uncertainty. It extends these results to consider decision problems where agents may have different attitude w.r.t. uncertainty (some may be optimistic when the others are pessimistic) and provides new decision rules for this specific case. Then, we tackle the problem of (possibilistic) sequential decision making and we provide an algorithmic study for strategy optimization in collective possibilistic decision trees.

The remainder of this paper is organized as follows: The next Section recalls the basic notions on which our work relies (decision under possibilistic uncertainty, collective utility functions, etc.). In Section 3, we resume the decision criteria introduced in [4] and define new ones for agents with different attitudes. Section 4 is devoted to strategy optimization in collective possibilistic decision trees, using the decision rules previously defined. The picture is finally completed by an experimental study, presented in Section 5.

2. Background and basic notions

2.1. Collective utility functions

Let us consider a set $\mathcal{A} = \{1, \ldots, p\}$ of agents that have to make a decision. Each agent $i \in \mathcal{A}$ being supposed to express his / her preferences on a set of alternatives (say, a set $\mathcal{X}$), by a ranking function or a utility function u_i that associates to each element of $\mathcal{X}$ a value in a subset of $\mathbb{R}^+$ (typically in the unit interval $[0, 1]$). In the absence of uncertainty, each decision leads to a unique consequence and an utility vector $\vec{u} = \langle u_1, \ldots, u_p \rangle$ is associated to each one of them. Besides, when agents are not equally important, we define a vector $\vec{w} = \langle w_1, \ldots, w_p \rangle$ where each i is equipped with a weight $w_i \in [0, 1]$ reflecting its importance. Thus, solving the problem comes down to compute a global utility degree that reflects the collective preference by aggregating the different u_i's.

In a qualitative framework, such aggregation shall be either conjunctive (i.e., based on a weighted min) or disjunctive (i.e., based on a weighted max) - see [15] for more details about weighted min and weighted max aggregations. Formally, these aggregations are defined as follows:

$$\text{Disjunctive aggregation}: \quad Agg^{max}(x) = \max_{i \in A} \min(w_i, u_i(x)). \tag{1}$$

$$\text{Conjunctive aggregation}: \quad Agg^{min}(x) = \min_{i \in A} \max(1 - w_i, u_i(x)). \tag{2}$$

2.2. Multi-agent decision making under risk

In a framework of decision making under risk, when the information about the consequences of decisions is probabilistic, a popular criterion to compare alternatives is the expected utility model axiomatized by Von Neumann and Morgenstern [35]: an elementary decision is modeled by a probability distribution over the set $\mathcal{X}$ of possible outcomes. It is called a simple probabilistic lottery and it is denoted by $L = \langle \lambda_1/x_1, \ldots, \lambda_n/x_n \rangle$, where $\lambda_j = p(x_j)$ is the probability that the decision leads to outcome x_j. Also, it is supposed that the preferences of a single decision maker are captured by a utility function u_j assigning a numerical value to each x_j. Solving such problems amounts to evaluate risky alternatives and choosing among them. In other words, we compute the expected utility of each lottery and we select the one with the highest value (the greater, the better).

When several agents are involved, the aggregation of individual preferences under risk raises a particular problem depending on when the utility of the agents is to be evaluated, before or after the consideration of uncertainty. This yields two different approaches, namely the *ex-ante* and the *ex-post* aggregation:

- The *ex-ante* approach consists in computing the utility of each agent, before performing the aggregation using the agents' weights.
- The *ex-post* approach consists in first determining the aggregated utility (conjunctive or disjunctive) relative to each possible outcome of $\mathcal{X}$; then consider the uncertainty and the likelihood of states.

For the same decision problem and using the same criterion, the two approaches do not always coincide: Aggregating the agents attitudes before or after the consideration of uncertainty may lead to different conclusions. This phenomenon has been identified by Myerson [37] as "timing effect".

2.3. Single agent decision making under possibilistic uncertainty

The expected utility model owes its popularity essentially to its strong axiomatic justifications [35,44]. However, it involves only the use of a quantitative representation of uncertainty and it has been proved that this formalism cannot represent all decision makers' behaviors [1,22]. Besides, when the decision maker is unable to express his / her uncertainty and preferences numerically and can only give an order among different alternatives, the probabilistic framework remains inappropriate and non probabilistic models (such as imprecise probabilities [46], evidence theory [45], rough set theory [39] and possibility theory [16,19,47,48]) become relevant alternatives. In particular, one may consider qualitative possibilistic decision rules that have emerged with the growth of possibility theory.

Possibility theory is a framework to handle uncertainty issued from Fuzzy Sets theory. It has been introduced by Zadeh [47,48] and further developed by Dubois and Prade [16,17]. The basic building block in this framework is the notion of possibility distribution. It represents the knowledge of a decision maker about the state of the world. A possibility distribution is denoted by π and it maps each state s in the universe of discourse S to a bounded linearly ordered scale V, typically the unit interval $[0, 1]$. This scale can be interpreted in a quantitative or a qualitative way, the latter is the context of our work. Independently of the used scale, given $s \in S$, $\pi(s) = 1$ means that the realization of s is totally possible and $\pi(s) = 0$ means that s is impossible. π is said to be normalized if there exist at least one $s \in S$ that is totally possible. Extreme cases of knowledge in possibility theory are complete knowledge and total ignorance. In the first case, we assign 1 to a totally possible state s_0 and 0 otherwise (i.e., $\exists\ s_0,\ \pi(s_0) = 1$ *and* $\forall s \neq s_0,\ \pi(s) = 0$). In the second one, we assign 1 to all situations (i.e., $\pi(s) = 1,\ \forall s \in S$).

Possibility theory is characterized by the use of two dual measures, namely the possibility measure Π and necessity measure N defined by:

- Possibility measure: $\Pi(E) = \max_{s \in E} \pi(s)$. It denotes the possibility degree evaluating at which level an event $E \subseteq S$ is consistent with the knowledge represented by π.
- Necessity measure: $N(E) = 1 - \Pi(\neg E) = \min_{s \in E} (1 - \pi(s))$. It denotes the necessity degree evaluating at which level an event $E \subseteq S$ is certainly implied by the knowledge.

Following Dubois and Prade's possibilistic approach of decision making under qualitative uncertainty [18], a decision can be seen as a possibility distribution over a finite set of outcomes $\mathcal{X}$ called a (simple) possibilistic lottery. Such a lottery is denoted by $L = \langle \lambda_1/x_1, \ldots, \lambda_n/x_n \rangle$ where $\lambda_j = \pi_L(x_j)$ is the possibility that decision L leads to outcome x_j; this possibility degree can also be denoted by $L[x_j]$. In this framework, a decision problem is thus fully specified by a set of possibilistic lotteries on $\mathcal{X}$ and a utility function $u : \mathcal{X} \mapsto [0, 1]$ expressing the decision maker preferences. Under the assumption that the utility scale and the possibility scale are commensurate and purely ordinal, Dubois and Prade propose to evaluate each lottery by a qualitative, optimistic or pessimistic utility [18]:

$$\text{Optimistic utility}: \quad U^+(L) = \max_{x_j \in \mathcal{X}} \min(L[x_j], u(x_j)). \tag{3}$$

$$\text{Pessimistic utility}: \quad U^-(L) = \min_{x_j \in \mathcal{X}} \max(1 - L[x_j], u(x_j)). \tag{4}$$

$U^+(L)$ is a mild version of the maximax criterion: L is good as soon as it is totally plausible that it gives a good consequence. On the contrary, the pessimistic utility, $U^-(L)$ estimates the utility of an act by its worst possible consequence: its value is high whenever L gives good consequences in every "rather plausible" state. These two utilities can be seen as the ordinal counterpart of the expected utility criterion and have been axiomatized in the style of Von Neumann and Morgenstern [18] and Savage [20] frameworks.

3. A possibilistic approach to collective decision making

3.1. Collective qualitative decision rules

Let us now consider collective decision making under qualitative uncertainty: In this framework, the decision maker's attitude with respect to uncertainty can be either optimistic (U^+) or pessimistic (U^-) and the aggregation of the agents' preferences can be either conjunctive, egalitarian (Agg^{min}) or disjunctive, non egalitarian (Agg^{max}).

Consider the decision problem defined by $\mathcal{A}$ a set of agents, $\vec{u} = \langle u_1, \ldots, u_p \rangle$ a vector of utility functions, $\vec{w} = \langle w_1, \ldots, w_p \rangle$ a weighting vector and $L = \langle L[x_1]/\vec{u}(x_1), \ldots, L[x_n]/\vec{u}(x_n) \rangle$ a possibilistic lottery on a set of consequences $\mathcal{X}$. Ben Amor et al. [4–6] have proposed and axiomatized four ex-ante and four ex-post decision criteria to solve such problems.

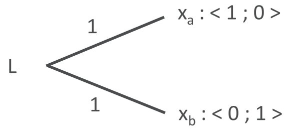

Fig. 1. Possibilistic (constant) lottery of Example 1.

$$U_{ante}^{-\min}(L) = \min_{i\in\mathcal{A}} \max\left((1-w_i), \quad \min_{x_j\in\mathcal{X}} \max\left(u_i(x_j), (1-L[x_j])\right)\right). \tag{5}$$

$$U_{ante}^{-\max}(L) = \max_{i\in\mathcal{A}} \min\left(w_i, \quad \min_{x_j\in\mathcal{X}} \max\left(u_i(x_j), (1-L[x_j])\right)\right). \tag{6}$$

$$U_{ante}^{+\min}(L) = \min_{i\in\mathcal{A}} \max\left((1-w_i), \quad \max_{x_j\in\mathcal{X}} \min\left(u_i(x_j), L[x_j]\right)\right). \tag{7}$$

$$U_{ante}^{+\max}(L) = \max_{i\in\mathcal{A}} \min\left(w_i, \quad \max_{x_j\in\mathcal{X}} \min\left(u_i(x_j), L[x_j]\right)\right). \tag{8}$$

$$U_{post}^{-\min}(L) = \min_{x_j\in\mathcal{X}} \max\left((1-L[x_j]), \min_{i\in\mathcal{A}} \max\left(u_i(x_j), (1-w_i)\right)\right). \tag{9}$$

$$U_{post}^{-\max}(L) = \min_{x_j\in\mathcal{X}} \max\left((1-L[x_j]), \max_{i\in\mathcal{A}} \min\left(u_i(x_j), w_i\right)\right). \tag{10}$$

$$U_{post}^{+\min}(L) = \max_{x_j\in\mathcal{X}} \min\left(L[x_j], \min_{i\in\mathcal{A}} \max\left(u_i(x_j), (1-w_i)\right)\right). \tag{11}$$

$$U_{post}^{+\max}(L) = \max_{x_j\in\mathcal{X}} \min\left(L[x_j], \max_{i\in\mathcal{A}} \min\left(u_i(x_j), w_i\right)\right). \tag{12}$$

For the notations: the subscript indicates the used approach (*ex-ante* or *ex-post*) and the superscript denotes the decision makers' attitude w.r.t. uncertainty (pessimistic "-" or optimistic "+") and the agents' preferences aggregation (conjunctive "min" or disjunctive "max").

The $U_{ante}^{-\min}$ utility for instance, considers that the decision makers are pessimistic and computes the pessimistic utility of each one of them. Then, the U_i^-'s are aggregated on a cautious basis: the higher the satisfaction of the least satisfied of the important agents, the better is the lottery. Using the same notations, $U_{post}^{-\max}$ considers that a consequence x_i is good as soon as one of the important agents is satisfied: a max-based aggregation of the utilities is performed, yielding a unique utility function $Agg()$ on the basis of which the pessimistic utility is computed.

In Ref. [6], authors have proposed a qualitative counterpart of Harsanyi's theorem [27], and have shown that the fully min oriented and fully max oriented *ex-ante* utilities are equivalent to their *ex-post* counterparts, i.e., $U_{ante}^{-\min} = U_{post}^{-\min}$ and $U_{ante}^{+\max} = U_{post}^{+\max}$. But $U_{ante}^{-\max}$ (resp. $U_{ante}^{+\min}$) may differ from $U_{post}^{-\max}$ (resp. from $U_{post}^{+\min}$). These criteria suffer from timing effect.

Example 1. Consider two equally important agents 1 and 2 ($w_1 = w_2 = 1$), and a lottery $L = \langle 1/x_a, 1/x_b\rangle$ defining a state of total ignorance about consequences x_a and x_b ($\pi(x_a) = \pi(x_b) = 1$) (see Fig. 1). The first consequence is good for 1 and bad for 2, and the second one is bad for 1 and good for 2: $u_1(x_a) = u_2(x_b) = 1$ and $u_2(x_a) = u_1(x_b) = 0$.

It is easy to check that $U_{ante}^{+\min}(L) = 0 \neq U_{post}^{+\min}(L) = 1$ where:

$$U_{post}^{+\min}(L) = max\left(min\left(1, min\left(max(1-1, 1), max(1-1, 0)\right)\right),\right.$$
$$\left. min\left(1, min\left(max(1-1, 0), max(1-1, 1)\right)\right)\right) = 0.$$
$$U_{ante}^{+\min}(L) = min\left(max\left(1-1, max\left(min(1, 1), min(1, 0)\right)\right),\right.$$
$$\left. max\left(1-1, max\left(min(1, 0), min(1, 1)\right)\right)\right) = 1.$$

3.2. Possibilistic decision rules for agents with different attitude w.r.t. to uncertainty

The decision rules presented in the previous section assume that all the agents are either purely optimistic or purely pessimistic. However, the attitude of each decision maker has a major impact on the chosen alternative or decision. For the same decision problem only varying the attitude of a decision maker may radically change the results. Besides, in real-world problems, it is seldom that all the decision makers have same attitude: the group of decision makers may gather pessimistic as well as optimistic persons. So, the decision have to be made by considering the individual differences in tolerance and intolerance for uncertainty. In the following, we get rid of the assumption of the same attitude for the collectivity and we propose more general decision rules that are appropriate to handle situations where each agent is free to express his / her attitude w.r.t. uncertainty.

Nevertheless, dealing with such problems imposes the use of *ex-ante* aggregation and forces us to give up the *ex-post* utilities. The *ex-post* approach that aggregates the preference utilities of the agents into a global, mono-agent one (this agent representing the collectivity), *imposes* the use of one and only one attitude. Hence, it is not possible to respect each agent's attitude towards uncertainty with a such method. The *ex-ante* approach, on the contrary, allows the handling of the different attitudes of heterogeneous agents. This leads us to extend the *ex-ante* decision rules as follows:

Definition 1. Given a possibilistic lottery L on $\mathcal{X}$, a set of agents $\mathcal{A}$ (where each i is either optimistic or pessimistic), a vector of utility functions $\vec{u}$ and a weighting vector $\vec{w}$, let:

$$U_{ante}^{max}(L) = \max_{i \in \mathcal{A}} \min \left(w_i, \bigotimes_{x_j \in \mathcal{X}} \oplus (u_i(x_j), \Lambda[x_j])\right). \tag{13}$$

$$U_{ante}^{min}(L) = \min_{i \in \mathcal{A}} \max \left(1 - w_i, \bigotimes_{x_j \in \mathcal{X}} \oplus (u_i(x_j), \Lambda[x_j])\right), \tag{14}$$

where $\otimes$ = min (resp. max), $\oplus$ = max (resp. min) and $\Lambda[x_j]$ = 1 - $L[x_j]$ (resp. $L[x_j]$) if the agent i is pessimistic (resp. optimistic).

Example 2. Consider two agents 1 and 2 having the same importance ($w_1 = w_2 = 1$) such that 1 is optimistic and 2 is pessimistic, and consider the lottery L defined in Example 1: $L = \langle 1/\langle 1, 0\rangle, 1/\langle 0, 1\rangle\rangle$. We get the following results:

$$\begin{aligned}
U_{ante}^{\max}(L) &= max\ (min(w_1, U_1^{+}(L)), min(w_2, U_2^{-}(L))). \\
&= max\ (\ min\ (1, max\ (\ min\ (1, 1),\ min\ (1, 0)))\ , \\
&\qquad min\ (1, min\ (max\ (1-1, 0), max\ (1-1, 1)))) = 1. \\
U_{ante}^{\min}(L) &= min\ (max\ (1 - w_1, U_1^{+}(L)), max\ (1 - w_2, U_2^{-}(L))). \\
&= min\ (\ max\ (1-1, max\ (\ min\ (1, 1),\ min\ (1, 0)))\ , \\
&\qquad max\ (1-1, min\ (max\ (1-1, 0), max\ (1-1, 1)))) = 0.
\end{aligned}$$

These criteria can be considered as a generalization of the four *ex-ante* utilities: Using the min (resp. max) oriented aggregation, U_{ante}^{min} (resp. U_{ante}^{max}) is equal to U_{ante}^{-min} (resp. U_{ante}^{-max}) if all agents are pessimistic and it is equal to U_{ante}^{+min} (resp. U_{ante}^{+max}) if there are optimistic. Obviously, the egalitarian utility U_{ante}^{min} is related to its non egalitarian counterpart by duality. Formally, it holds that:

Proposition 1. *Let $P = \langle \mathcal{L}, \vec{w}, \vec{u}\rangle$ be a qualitative collective decision problem and let $P^{\tau} = \langle \mathcal{L}, \vec{w}, \vec{u}^{\tau}\rangle$ be its dual problem, i.e., the problem such that for each agent we consider his / her dual attitude and for any $x_j \in \mathcal{X}$, $i \in \mathcal{A}$, we define $u_i^{\tau}(x_j) = 1 - u_i(x_j)$. Then, for any $L \in \mathcal{L}$:*

$$U_{ante}^{\tau\ \max}(L) = 1 - U_{ante}^{\min}(L) \text{ and},$$
$$U_{ante}^{\tau\ \min}(L) = 1 - U_{ante}^{\max}(L).$$

Likewise, if agents are equally important, weights can be ruled out and the proposed criteria can be simplified as follows:

Definition 2. Given a possibilistic lottery L on $\mathcal{X}$, a set of equally important agents $\mathcal{A}$ where each agent is either optimistic or pessimistic and a vector of utility functions $\vec{u}$ let:

$$U_{ante}^{max}(L) = \max_{i \in \mathcal{A}} \left(\bigotimes_{x_j \in \mathcal{X}} \oplus (u_i(x_j), \Lambda[x_j])\right). \tag{15}$$

$$U_{ante}^{min}(L) = \min_{i \in \mathcal{A}} \left(\bigotimes_{x_j \in \mathcal{X}} \oplus (u_i(x_j), \Lambda[x_j])\right), \tag{16}$$

where $\otimes$ = min (resp. max), $\oplus$ = max (resp. min) and $\Lambda[x_j]$ = 1 - $L[x_j]$ (resp. $L[x_j]$) if the agent i is pessimistic (resp. optimistic).

4. Collective sequential decision making under qualitative uncertainty

4.1. Definition of collective possibilistic decision trees

Representation formalisms such as decision trees [40], influence diagrams [28] and Markov decision process [3], offer a clear description of sequential decision problems and allow the definition of optimal strategies. In recent years, there has been a growing interest in more complex problems (namely multi-criteria or multi-objectives decision making) and several extensions of classical graphical models [9,26,32,33] have emerged to present such cases. These research proposals rely on the probability theory and the well known expected utility criterion to solve the problem. However, this criterion fails to represent all the decision makers' behaviors.

With the growth of the qualitative frameworks, especially possibility theory, many authors have advocated this ordinal view of decision making and have gave rise to qualitative decision models i.e., possibilistic decision tree [24], possibilistic influence diagrams [24], and possibilistic Markov decision processes [42]. We notice that other non possibilistic qualitative paradigms have been developed, we cite among others, works presented in [10,31,34]. But, in this paper we focus only on possibilistic formalisms because of the use of pessimistic and optimistic utilities that are the ordinal counter part of the expected utility criterion and have received an axiomatic justification by Dubois and Prade [18,20].

In the remaining of this Section, we propose to solve sequential collective decision problems under qualitative uncertainty. For the best of our knowledge, this work is the first attempt to solve such problems and we propose to limit ourselves to the use of decision trees; because even in this simple, explicit formalism, the set of potential strategies is combinatorial (i.e., its size increases exponentially with the size of the tree) and the determination of an optimal strategy for a given problem and using the different proposed decision rules is an algorithmic issue in itself.

Decision trees proposed by Raiffa [40] in 1968 are the most popular graphical models. They encode the structure of sequential problems by representing all possible scenarios. The graphical component of a decision tree is a directed labeled tree $\mathcal{DT} = (\mathcal{N}, \mathcal{E})$ where $\mathcal{E}$ is the set of edges and $\mathcal{N} = \mathcal{D} \cup \mathcal{C} \cup \mathcal{LN}$ is the set of nodes that contains three kinds of nodes: $\mathcal{D}$ the set of decision nodes (represented by squares); $\mathcal{C}$ the set of chance nodes (represented by circles) and $\mathcal{LN}$ the set of leaves. In this formalism, the root of the tree is generally a decision node, denoted by D_0. $Succ(N)$ denotes the set of children nodes of node N. For any $D_i \in \mathcal{D}, Succ(D_i) \subseteq \mathcal{C}$ i.e., a chance node (an action) must be chosen at each decision node. For any $C_i \in \mathcal{C},\ Succ(C_i) \subseteq \mathcal{LN} \cup \mathcal{D}$: the set of outcomes of an action is either a leaf node or a decision node (and then a new action should be executed).

The numerical component of decision trees consists on assigning utility values to leave nodes and labeling the edges outgoing from chance nodes. The quantification of a decision tree depends essentially on the nature of uncertainty pertaining the problem and the theory used to represent it. In its classical version, decision trees are probabilistic. However, when the available information is ordinal, Garcia et al. [24] propose to label the leaves by utility degrees in the unit scale [0, 1] and to represent the uncertainty pertaining to the possible outcomes of each C_i by a *conditional possibility distribution* π_i on $Succ(C_i)$, such that $\forall N \in Succ(C_i)$, $\pi_i(N) = \Pi(N|path(C_i))$ where $path(C_i)$ denotes all the value assignments of chance and decision nodes on the path from the root D_0 to C_i.

Solving a decision tree amounts at building a *complete strategy* that selects an action (a chance node) for each decision node: a strategy is a mapping $\delta : \mathcal{D} \mapsto \mathcal{C} \cup \{\perp\}$. $\delta(D_i) = \perp$ means that no action has been selected for D_i (δ is partial). To select the optimal strategy, authors in [24] propose to evaluate and compare strategies w.r.t. the pessimistic and optimistic utilities axiomatized in [18]. Leaf nodes being labeled with utility degrees, the rightmost chance nodes (i.e., chance nodes on the far right-hand side) can be seen as *simple possibilistic lotteries*. Then, each strategy δ can be viewed as a connected sub-tree of the decision tree and is identified with a *possibilistic compound lottery* L_δ, i.e., with a possibility distribution over a set of (simple or compound) lotteries. Any compound lottery is denoted by $\langle \lambda_1/L_1, \ldots, \lambda_m/L_m \rangle$ and it can be reduced into an equivalent simple lottery[1] as follows [18]:

$$Reduction(\langle \lambda_1/L_1, ..., \lambda_m/L_m \rangle) = \langle \max_{k=1,m} (\min(\lambda_1^k, \lambda_k))/x_1, \ldots, \max_{k=1,m} (\min(\lambda_n^k, \lambda_k))/x_n \rangle,$$

where λ_k is the possibility of getting lottery L_k according to $\mathcal{L}$ and $\lambda_j^k = \pi_{L_k}(x_j)$ is the conditional possibility of getting x_j from L_k. Hence, the pessimistic and optimistic utility of a strategy δ can be computed on the basis of the reduction of L_δ: the utility of a strategy δ is then the one of $Reduction(L_\delta)$.

To define collective qualitative decision trees, we resume the same graphical and numerical component of possibilistic decision trees [24] except for leave nodes that are evaluated according to several agents instead of a single one.

Each leaf node LN is now labeled by a *vector* $\vec{u}(LN) = \langle u_1(LN), \ldots, u_p(LN) \rangle$ rather than by a single utility degree (see Fig. 2). A strategy still leads to a compound lottery, and can be reduced, thus leading in turn to a simple (but multi-agent) lottery. We can now compare strategies according to any of the collective decision rules O previously presented (U_{post}^{-min}, U_{post}^{+min}, U_{post}^{-max}, U_{post}^{+max}, U_{ante}^{-min}, U_{ante}^{+min}, U_{ante}^{-max}, U_{ante}^{+max}, U_{ante}^{min} and U_{ante}^{max}) by comparing their reductions. Formally, given two strategies δ_1 and δ_2, and a collective decision rule O:

$$\delta_1 \succeq_O \delta_2 \text{ iff } U_O(\delta_1) \geq U_O(\delta_2), \text{ where } \forall \delta, U_O(\delta) = U_O(Reduction(L_\delta)). \quad (17)$$

Example 3. Consider the tree of Fig. 2, involving two equally important agents and the strategy $\delta(D_0) = C_1$, $\delta(D_1) = C_3$, $\delta(D_2) = C_5$. It holds that:

$$L_\delta = \langle 1/L_{C_3}, 0.9/L_{C_5} \rangle \text{ with } L_{C_3} = \langle 0.5/x_a, 1/x_b \rangle,\ L_{C_5} = \langle 0.2/x_a, 1/x_b \rangle.$$

The reduction of L_δ can be computed:

[1] Obviously, the reduction of a simple lottery is the simple lottery itself.

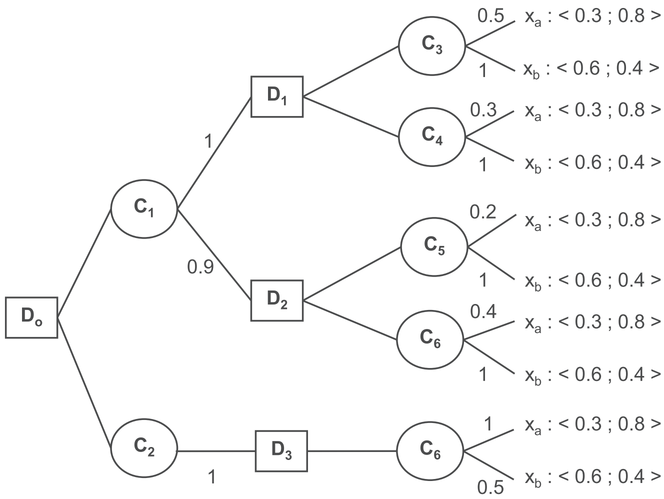

Fig. 2. Collective possibilistic decision tree of Example 3.

$$Reduction(L_\delta) = \langle max(0.5, 0.2)/x_a, max(1, 0.9)/x_b\rangle = \langle 0.5/x_a, 1/x_b\rangle.$$

So, if we consider for instance the U_{ante}^{+min} criterion, we get:

$$U_{ante}^{+min}(\delta) = \min(\max\min(0.5, 0.3), \min(1, 0.6), \max(\min(0.5, 0.8)\min(1, 0.4))) = 0.5.$$

The definition proposed by Eq (17) is intuitive but raises an algorithmic challenge: the set of strategies to compare is exponential w.r.t. the size of the tree which makes the explicit evaluation of strategies not realistic. The sequel of the paper provides an algorithmic study of the problem - applying variants of Dynamic Programming when it is possible.

4.2. Optimization in collective possibilistic trees

4.2.1. Dynamic Programming as a tool for ex-post *utilities*

Dynamic Programming [2] is an efficient procedure of strategy optimization. It proceeds by *backward induction* where the problem is handled from the end (in our case, from the leafs); the last decision nodes are considered first, and recursively until reaching the root. More specifically, the algorithm can be described as follows: when a chance node C_i is reached, an optimal sub-strategy is built for each of its children. These sub-strategies are combined w.r.t. their uncertainty degrees. Then, the resulting compound strategy is reduced to an equivalent simple lottery representing the current optimal sub-strategy. When a decision node D_i is reached, we select a decision D^* among all the possible alternatives $N \in Succ(D)$ leading to an optimal sub-strategy w.r.t. $\succeq_O$. The choice is performed by comparing the simple lotteries equivalent to each sub-strategy.

This algorithm is sound and complete as soon as $\succeq_O$ is complete, transitive and satisfies the principle of weak monotonicity,[2] that ensures that each sub strategy of an optimal strategy is optimal in its sub-tree. Formally, a decision rule O is weakly monotonic iff whatever L, L' and L'', whatever (α, β) such that max $(\alpha, \beta) = 1$:

$$L \succeq_O L' \Rightarrow \langle \alpha/L, \beta/L''\rangle \succeq_O \langle \alpha/L', \beta/L''\rangle.$$

Each of the *ex-post* criteria satisfies transitivity, completeness and weak monotonicity, because collapsing to either classical pessimistic (U^-) or optimistic (U^+) utility, which satisfies these properties [8,24].

The adaptation of Dynamic Programming to the *ex-post* decision rules is detailed in Algorithm 1. This algorithm computes the collective utility relative to each possible consequence by aggregating the utility values of each leaf, and then builds an optimal strategy from the last decision nodes to the root of the tree using the principle defined in [24,43] for classical (mono-agent) possibilistic decision trees.

4.2.2. Dynamic Programming for ex-ante *utilities*

When the decision rule follows the *ex-ante* approach, the application of Dynamic Programming is a little more tricky. The *ex-ante* Dynamic Programming we propose (see Algorithm 2) keeps at each node a vector of p pessimistic (resp. optimistic) utilities, one for each agent. The computation of the *ex-ante* utility can then be performed each time a decision is to be made. Recall that $U_{ante}^{-\min} = U_{post}^{-\min}$ and $U_{ante}^{+\max} = U_{post}^{+\max}$. Hence, for these two criteria the optimization could also be performed using the *ex-post* algorithm.

[2] It is a common knowledge in sequential decision making that monotonicity is a necessary and a sufficient condition for the optimality of Dynamic Programming. This property guarantees also the completeness of a polytime algorithm [30,36,38].

Algorithm 1: DynProgPost: *Ex-post* Dynamic Programming.

Data: T: a decision tree, N: a node of T
Result: u^*: the value of the optimal strategy δ^* - δ^* is stored as a global variable

begin
  $u^* = 0$; // Initialization
  if $N \in \mathcal{LN}$ **then** // Leaf: CDM aggregation
    for $i \in \{1, \dots, p\}$ **do** $u_N \leftarrow (u_N \oplus (u_i \otimes \omega_i))$;
    // $\otimes = \min$, $\omega_i = w_i$, $\oplus = \max$ for disjunctive aggregation
    // $\otimes = \max$, $\omega_i = 1 - w_i$, $\oplus =$ *min* for conjunctive aggregation;
  if $N \in \mathcal{C}$ **then** // Chance Node: computes the qualitative utility
    foreach $Y \in Succ(N)$ **do** $u_N \leftarrow (u_N \oplus (\lambda_Y) \otimes DynProgPost(T, Y))$;
    // $\otimes = \min$, $\lambda_Y = \pi(Y)$, $\oplus = \max$ for optimistic utility
    // $\otimes = \max$, $\lambda_Y = 1 - \pi(Y)$, $\oplus = \min$ for pessimistic utility
  if $N \in \mathcal{D}$ **then** // Decision node: determines the best decision
    foreach $Y \in Succ(N)$ **do**
      $u_Y \leftarrow DynProgPost(T, Y)$;
      if $u_Y \geq u^*$ **then** $\delta(N) \leftarrow Y$ and $u^* \leftarrow u_Y$;
  return u^*;

Algorithm 2: DynProgAnte: *Ex-ante* Dynamic Programming.

Data: T: a decision tree, N: a node of T
Result: u^*: the value of the optimal strategy δ^* - δ^* is stored as a global variable

begin
  $u^* = 0$; // Initialization
  if $N \in \mathcal{LN}$ **then** // Leaf
    for $i \in \{1, \dots, p\}$ **do** $\vec{u_N}[i] \leftarrow u_i$;
  if $N \in \mathcal{C}$ **then** // Chance Node: computes the utility vectors
    for $i \in \{1, \dots, p\}$ **do** $\vec{u_N}[i] \leftarrow \epsilon$;
    foreach $Y \in Succ(N)$ **do**
      $\vec{u_Y} \leftarrow DynProgAnte(T, Y)$;
      for $i \in \{1, \dots, p\}$ **do** $\vec{u_N}[i] \leftarrow (\vec{u_N}[i] \oplus (\lambda_Y \otimes \vec{u_Y}[i]))$;
    // Optimistic utility $\otimes = \min$, $\lambda_Y = \pi(Y)$, $\oplus = \max$, $\epsilon \leftarrow 0$
    // Pessimistic utility $\otimes = \max$, $\lambda_Y = 1 - \pi(Y)$, $\oplus = \min$, $\epsilon \leftarrow 1$
  if $N \in \mathcal{D}$ **then** // Decision node
    foreach $Y \in Succ(N)$ **do**
      $v_Y \leftarrow \epsilon$; $\vec{u_Y} \leftarrow DynProgAnte(T, Y)$;
      for $i \in \{1, \dots, p\}$ **do** $v_Y \leftarrow v_Y \oplus (\vec{u_Y}[i] \otimes \omega_i)$;
      if $v_Y > u^*$ **then** $\delta(N) \leftarrow Y$, $\vec{u_N} \leftarrow \vec{u_Y}$ and $u^* \leftarrow v_Y$;
    // Disjunctive CDM: let $\otimes = \min$, $\omega_i = w_i$, $\oplus = \max$, $\epsilon \leftarrow 0$
    // Conjunctive CDM: let $\otimes = \max$, $\omega_i = 1 - w_i$, $\oplus =$ *min*, $\epsilon \leftarrow 1$
  return u^*;

As shown by Counter Example 1, the $U_{ante}^{-\max}$ and $U_{ante}^{+\min}$ decision rules do not satisfy the monotonicity principle [4]. Hence, Algorithm 2 may provide a good strategy but without any guarantee of optimality. It can nevertheless be considered as an approximation algorithm when used for optimizing any of these problematic criteria

Counter-Example 1. Consider the set of consequences $\mathcal{X} = \{x_1, x_2, x_3\}$ and consider two equally important agents 1 and 2 $(w_1 = w_2 = 1)$ with: $u_1(x_1) = 1$, $u_1(x_2) = 0.8$, $u_1(x_3) = 0.5$; $u_2(x_1) = 0.6$, $u_2(x_2) = 0.8$, $u_2(x_3) = 0.8$.

Consider the lotteries $L_1 = \langle 1/x_1, 0/x_2, 0/x_3 \rangle$, $L_2 = \langle 0/x_1, 1/x_2, 0/x_3 \rangle$ and $L_3 = \langle 0/x_1, 0/x_2, 1/x_3 \rangle$:

L_1 gives consequence x_1 for sure, L_2 gives consequence x_2 for sure and L_3 gives consequence x_3 for sure. It holds that:

$$\begin{aligned} U_{ante}^{-max}(L_1) &= \max_{i=1,2} U_i^-(L_1) &= \max(1, 0.6) &= 1. \\ U_{ante}^{-max}(L_2) &= \max_{i=1,2} U_i^-(L_2) &= \max(0.8, 0.8) &= 0.8. \end{aligned}$$

Hence $L_1 \succ L_2$ with respect to the U_{ante}^{-max} rule.

Consider now the compound lotteries $L = \langle 1/L_1, 1/L_3 \rangle$ and $L' = \langle 1/L_2, 1/L_3 \rangle$. If the weak monotonicity principle were satisfied, we would get: $U_{ante}^{-max}(L) > U_{ante}^{-max}(L')$.

Consider the reduction of compound lotteries L and L' respectively such as:

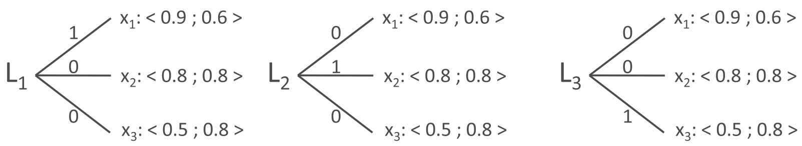

Fig. 3. Lotteries of Counter-Example 2.

$Reduction(\langle 1/L_1, 1/L_3\rangle) = \langle 1/x_1, 0/x_2, 1/x_3\rangle$ and
$Reduction(\langle 1/L_2, 1/L_3\rangle) = \langle 0/x_1, 1/x_2, 1/x_3\rangle$. It holds that:
$U_{ante}^{-max}(L) = U_{ante}^{-max}(Reduction(\langle 1/L_1, 1/L_3\rangle)) = 0.6$.
$U_{ante}^{-max}(L') = U_{ante}^{-max}(Reduction(\langle 1/L_2, 1/L_3\rangle)) = 0.8$.
$U_{ante}^{-max}(L) < U_{ante}^{-max}(L')$ while $U_{ante}^{-max}(L_1) > U_{ante}^{-max}(L_2)$. So, U_{ante}^{-max} is not monotonic.

Using the fact that $U_{ante}^{+min} = 1 - U_{ante}^{\tau-max}$ [4], this counter-example is modified to show that U_{ante}^{+min} does not satisfy the monotonicity principle either.

Consider two equally important agents, 1 and 2 with $w_1 = w_2 = 1$ and utilities

$$u_1^\tau(x_1) = 0,\ u_1^\tau(x_2) = 0.2,\ u_1^\tau(x_3) = 0.5;\ u_2^\tau(x_1) = 0.4,\ u_2^\tau(x_2) = 0.2,\ u_2^\tau(x_3) = 0.2.$$

Consider now the same lotteries L_1, L_2 and L_3 presented above. It holds that:

$$U_{ante}^{+min}(L_1) = \min_{i=1,2} U_i^+(L_1) = 0 < U_{ante}^{+min}(L_2) = \min_{i=1,2} U_i^+(L_2) = 0.2, \text{while}$$

$$U_{ante}^{+min}(Reduction(\langle 1/L_1, 1/L_3\rangle)) = 0.4 > U_{ante}^{+min}(Reduction(\langle 1/L_2, 1/L_3\rangle)) = 0.2,$$

which contradicts the weak monotonicity.

Likewise, the *ex-post* Dynamic Programming Algorithm (Algorithm 1) shall also be considered as an algorithm of approximation for $U_{ante}^{-\max}$ and $U_{ante}^{+\min}$ since they are correlated to their *ex-post* counter-parts as shown in [6]. Indeed, it holds that:

Proposition 2.

$$U_{ante}^{+\min}(L) \geq U_{post}^{+\min}(L).$$
$$U_{ante}^{-\max}(L) \leq U_{post}^{-\max}(L).$$

So, even if it is not always the case, it often happens that $U_{post}^{-\max} = U_{ante}^{-\max}$ (resp. $U_{post}^{+\min} = U_{ante}^{+\min}$); in these cases the solution provided by the *ex-post* Algorithm is optimal.

Algorithm 2 applies also for the optimization of the *ex-ante* generalization decision rules, namely the $U_{ante}^{\max}$ and $U_{ante}^{\min}$ utilities for heterogeneous agents. However, obtaining the optimal strategy using Dynamic Programming is guaranteed only for monotonic criteria. Counter-Example 2 shows that $U_{ante}^{\max}$ as well as $U_{ante}^{\min}$ do not satisfy this property.

Counter-Example 2. Consider two agents 1 and 2 having the same importance ($w_1 = w_2 = 1$), 1 being optimistic and 2 being pessimistic, and consider the three lotteries on $\mathcal{X} = \{x_1, x_2\}$ depicted in Fig. 3. Let L and L' be two compound lotteries defined by: $L = \langle 0.6/L_1, 1/L_3\rangle$ and $L' = \langle 0.6/L_2, 1/L_3\rangle$.

We can verify that L_1 is globally preferred to L_2 where:

$$U_{ante}^{\max}(L_1) = 0.9 > U_{ante}^{\max}(L_2) = 0.8 \text{ whereas } U_{ante}^{\max}(L) = 0.6 < U_{ante}^{\max}(L') = 0.8,$$

which proves that $U_{ante}^{\max}$ is not monotonic.

Since $U_{ante}^{min} = 1 - U_{ante}^{\tau max}$ (Proposition 1), this counter-example can be modified to show that U_{ante}^{+min} does not satisfy the monotonicity principle either. We consider the agents 1 and 2 with the same importance ($w_1 = w_2 = 1$) where 1 is pessimistic and 2 is optimistic and we replace utility functions relative to x_1 and x_2 for lotteries L_1, L_2, L_3, L and L' as follows: $u_1^\tau(x_1) = 0.1$, $u_2^\tau(x_1) = 0.4$, $u_1^\tau(x_2) = 0.2$, $u_2^\tau(x_2) = 0.2$, $u_1^\tau(x_3) = 0.5$ and $u_2^\tau(x_3) = 0.2$. We can check that L_2 is better than L_1 since $U_{ante}^{\min}(L_1) = 0.1 < U_{ante}^{\min}(L_2) = 0.2$ while $U_{ante}^{\min}(L) = 0.4 > U_{ante}^{\min}(L') = 0.2$, which proves that $U_{ante}^{\min}$ does not satisfy monotonicity property.

4.2.3. *Right optimization of $U_{ante}^{-\max}$ by Multi-Dynamic Programming*

The lack of monotonicity of $U_{ante}^{-\max}$ is not dramatic, even when optimality must be guaranteed. Indeed, with $U_{ante}^{-\max}$ we look for a strategy that has a good pessimistic utility U_i^- for *at least* one agent i. This means that if it is possible to get

Algorithm 3: MultiDynProg: right optimization of $U^{-\max}_{ante}$.

Data: T: a decision tree
Result: u^*: the value of the optimal strategy δ^*- δ^* is stored as a global value

begin
  $u = 0$; // Initialization
  $u^* = 0$; // Initialization
  for $i \in \{1, \dots, p\}$ **do**
    $\delta_i = \emptyset$; // Initialization
    $\delta_i \leftarrow PesDynProg(T, i)$ // Call to classical poss.Dyn.Prog. [24] - returns an optimal strategy and its value $U_i^-(\delta_i)$;
    $u \leftarrow \min\ (U_i^-(\delta_i),\ \omega_i)$;
    if $u > u^*$ **then** $\delta^* \leftarrow \delta_i$; $u^* \leftarrow u$;
return u^*;

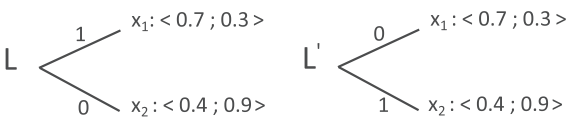

Fig. 4. Lotteries of Counter-Example 3.

for each i a strategy that optimizes U_i^- (and this can be done by the classical Dynamic Programming, since the pessimistic utility is monotonic), the one with the highest value for $U^{-\max}_{ante}$ is globally optimal. Formally, $U^{-\max}_{ante}$ can be expressed as follows:

$$U^{-\max}_{ante}(L) = \max_{i=1,p} \min(w_i, U_i^-(L)) \tag{18}$$

where $U_i^-(L)$ is the pessimistic utility of L according to agent i.

Corollary 1. *Let $\mathcal{L}$ be the set of possibilistic lotteries that can be built on $\mathcal{X}$, L be any possibilistic lottery and let:*
- $\mathcal{L}^* \subset \mathcal{L}$ s.t. $\mathcal{L}^* = \{L_1^*, \dots, L_p^*\}$ and $\forall L \in \mathcal{L}$, $U_i^-(L_i^*) \geq U_i^-(L)$;
- $L^* \in \mathcal{L}^*$, s.t. $\forall L_i^* \in \mathcal{L}^*$: $\max_{i=1,p} \min\ (w_i,\ U_i^-(L^*)) \geq \max_{i=1,p} \min\ (w_i,\ U_i^-(L_i^*))$.
It holds that $U^{-\max}_{ante}(L^*) \geq U^{-\max}_{ante}(L), \forall L \in \mathcal{L}$.

It follows that the optimization problem can be solved by a series of p calls to a classical (mono-agent) pessimistic optimization. This is the principle of the Multi-Dynamic Programming approach detailed by Algorithm 3. To reduce the execution time, this algorithm could be improved by considering only agents having importance weight w_i greater than $U^{-\max}_{ante}$ of the actual strategy.

4.2.4. Right optimization of $U^{+\min}_{ante}$: a Branch and Bound algorithm

As previously said, $U^{+\min}_{ante}$ utility does not satisfy monotonicity. So, *ex-ante* Dynamic Programming (Algorithm 2) can provide a good strategy, but without any guarantee of optimality. Besides, this criterion performs the egalitarian aggregation (use the min operator). Then unfortunately, as shown in Counter-Example 3 it is not possible to provide a result similar to Corollary 1 to circumvent the lack of monotonicity (as for $U^{-\max}_{ante}$).

Counter-Example 3. Consider two optimistic agents 1 and 2 having the same importance degree ($w_1{=}w_2{=}1$), and consider the two lotteries L and L' on $\mathcal{X} = \{x_1, x_2\}$ depicted in Fig. 4. L gives consequence x_1 for sure and L' gives consequence x_2 for sure. It holds that:

$$U^{+\min}_{ante}(L) = min\ (\ max\ ((1-1), max\ (\ min\ (1, 0.7),\ min\ (0, 0.4)))\ ,$$
$$max\ ((1-1), max\ (min\ (1, 0.3), min\ (0, 0.9)))) = 0.3.$$
$$U^{+\min}_{ante}(L') = min\ (\ max\ ((1-1), max\ (\ min\ (0, 0, 7),\ min\ (1, 0.4)))\ ,$$
$$max\ ((1-1), max\ (min\ (0, 0.3), min\ (1, 0.9)))) = 0.4.$$

Hence, $L' \succ L$ with respect to the $U^{+\min}_{ante}$ rule. Then, L' is the optimal lottery L^*.

Now, we look for the strategy that optimizes U_i^+ for each agent i. We get:

$U_1^+(L) = 0.7 \succ U_1^+(L') = 0.4$. So, $L_1^* = L$.

$U_2^+(L) = 0.3 \succ U_2^+(L') = 0.9$. So, $L_2^* = L'$.

Algorithm 4: B&B algorithm for the optimization of $U^{+\min}_{ante}$.

Data: T: a decision tree, δ: a (partial) strategy, u: an upper Bound of $U^{+\min}_{ante}(\delta)$
Result: u^*: the $U^{+\min}_{ante}$ value of the optimal strategy δ^* found so far

begin
  if $\delta(D_0) = \perp$ **then** $\mathcal{D}_{pend} \leftarrow \{D_0\}$;
  else $D_{pend} \leftarrow \{D_i \in \mathcal{D}$ s.t. $\exists D_j, \delta(D_j) \neq \perp$ and $D_i \in Succ(\delta(D_j))\}$;
  if $\mathcal{D}_{pend} = \emptyset$ **then** // δ is a complete strategy
    $\delta^* \leftarrow \delta$; $u^* \leftarrow u$;
  else
    $D_{next} \leftarrow arg\ min_{D_i \in D_{pend}}\ i$;
    foreach $C_i \in Succ(D_{next})$ **do**
      $\delta(D_{next}) \leftarrow C_i$;
      $u \leftarrow UpperBound(T, \delta)$;
      if $u > u^*$ **then** $u^* \leftarrow B\&B(u, \delta)$;
  return u^*;

We can check that $\max_{i\in\{1,2\}} \min\ (w_i, U_i(L_i^*)) = 0.7 \succ \max_{i\in\{1,2\}} \min\ (w_i, U_i(L^*)) = 0.4$, which proves that a result similar to Corollary 1 does not hold for $U^{+\min}_{ante}$.

To guarantee the optimality, we have to propose an exact algorithm and proceed by an implicit enumeration via a Branch and Bound (B&B) algorithm, as done for Rank Dependent Utility [29] and for Possibilistic Choquet integrals [8] (both in the mono agent case).

The Branch and Bound procedure described by Algorithm 4 takes as argument a partial strategy δ and an upper bound of the $U^{+\min}_{ante}$ value of its best extension. It returns u^* the $U^{+\min}_{ante}$ value of the best strategy δ^* found so far. To reduce the research time, we can initialize δ^* with any strategy, e.g. the one provided by Dynamic Programming (using Algorithm 2 or even Algorithm 1 proposed for the *ex-post* approach). At each step of the Branch and Bound algorithm, the current partial strategy δ is developed by the choice of an action for some unassigned decision node. When several decision nodes are candidate, the one with the minimal rank (i.e., the former one according to the temporal order) is developed. The recursive procedure backtracks when either the current strategy is complete (then δ^* and u^* are updated) or proves to be worse than the current δ^*.

Function $UpperBound(T, \delta)$ outlined by Algorithm 5 provides an upper bound of the best completion of δ: In practice, it builds for each agent i, a strategy δ_i that maximizes U_i^+ (using [41,43]'s algorithm, which is linear). It then selects, among these strategies, the one with the highest $U^{+\min}_{ante}$. Notice that $UpperBound(T, \delta) = U^{+\min}_{ante}(\delta)$ when δ is complete. Whenever the value returned by $UpperBound(T, \delta)$ is lower or equal to u^*, the value of the best current strategy, the algorithm backtracks yielding the choice of another action for the last considered decision node.

4.2.5. Agents with different attitudes: right optimization of U^{max}_{ante} and U^{min}_{ante}

Let us finally study the heterogeneous utilities U^{max}_{ante} and U^{min}_{ante} proposed in Section 3.2 to solve collective decision problems where the set of decision makers gathers pessimistic and optimistic agents. These decision rules are a generalization of *ex-ante* utilities so it is not surprising that they do not satisfy the weak monotonicity (see Counter Example 2).

4.2.6. Optimization for heterogeneous agents - the max-based rule

When optimizing U^{max}_{ante}, we are looking for a strategy that maximizes the qualitative utility (U_i^- or U_i^+) for at least one agent i: we get for each agent an optimal strategy with regards to $U^{\otimes}$ that is defined according to his / her attitude w.r.t. uncertainty i.e., $U_i^{\otimes} = U_i^-$ (resp. $U_i^{\otimes} = U_i^+$) if the agent is pessimistic (resp. optimistic). Then, we select the strategy that maximizes the global utility U^{max}_{ante}. Basically, the optimization of these criteria relies on the same idea than the one proposed for the optimization of U^{-max}_{ante}. Formally, we can write:

$$U^{\max}_{ante}(L) = \max_{i=1,p} \min(w_i, U_i^{\otimes}(L)); \tag{19}$$

where $U_i^{\otimes}(L)$ denotes either the pessimistic or the optimistic utility of L for agent i, his / her attitude w.r.t. uncertainty being captured by $\otimes$.

Corollary 2. *Let $\mathcal{L}$ be the set of possibilistic lotteries that can be built on $\mathcal{X}$, L be any possibilistic lottery and let:*
- $\mathcal{L}^* \subset \mathcal{L}$ s.t. $\mathcal{L}^* = \{L_1^*, \ldots, L_p^*\}$ and $\forall L \in \mathcal{L}$, $U_i^{\otimes}(L_i^*) \geq U_i^{\otimes}(L)$;
- $L^* \in \mathcal{L}^*$, s.t. $\forall L_i^* \in \mathcal{L}^*$: $\max_{i=1,p} \min\ (w_i,\ U_i^{\otimes}(L^*)) \geq \max_{i=1,p} \min\ (w_i,\ U_i^{\otimes}(L_i^*))$.
It holds that $U^{\otimes\max}_{ante}(L^*) \geq U^{\otimes\max}_{ante}(L), \forall L \in \mathcal{L}$.

Algorithm 5: UpperBound of B&B algorithm for the optimization of U_{ante}^{+min}.

Data: T: a decision tree, δ: a (partial or complete) strategy, N: a node of T
Result: $u(\delta)$: the $U_{ante}^{+\min}$ value of the current strategy δ

begin
  $u_N = 0$; `// Initialization`
  $u(\delta) = 0$; `// Initialization`
  for $i \in \{1, \ldots, p\}$ **do**
    if $N \in \mathcal{LN}$ **then** `// Leaf`
      $u_N \leftarrow u_i$;
    if $N \in \mathcal{C}$ **then** `// Chance Node: computes the optimistic utility`
      foreach $Y \in Succ(N)$ **do**
        $u_Y \leftarrow DynProgAnte(T, Y)$; `// Call to Ex-ante Dyn.Prog. (Algorithm 2)`
        $u_N \leftarrow \max(u_N, \min(\pi_Y, u_Y))$;
    if $N \in \mathcal{D}$ **then** `// Decision node`
      if $\delta(N) \neq \perp$ **then** `// Prefixed action`
        $\delta_i(N) \leftarrow \delta(N)$ and $u_N \leftarrow u_Y$
      else
        foreach $Y \in Succ(N)$ **do**
          $u_Y \leftarrow DynProgAnte(T, Y)$;
          if $u_Y > u_N$ **then**
            $\delta_i(N) \leftarrow Y$ and $u_N \leftarrow u_Y$;
    for $j \in \{1, \ldots, p\}$ **do**
      $u = 0$; `// Initialization`
      $U_j^+(\delta_i) \leftarrow OptUtil(\delta_i, j)$; `// Computes for each agent` j `the value of its optimistic utility` $U_j^+(\delta_i)$;
      $u \leftarrow \min\ (u,\ \max\ (U_j^+(\delta_i),\ 1 - \omega_i))$;
    if $u > u(\delta)$ **then** $u(\delta) \leftarrow u$;
  return $u(\delta)$;

Algorithm 6: ImproHetMultiDynProg: right optimization of $U_{ante}^{\max}$.

Data: T: a decision tree; D_0: root of T
Result: u^*: the value of the optimal strategy δ^*-δ^* is stored as a global variable

begin
  $u^* = 0$; `// Initialization`
  $\delta = DynProgPost(T, D_0, Opt)$; `//` Opt `is the subset of optimistic agents - returns an optimal strategy` δ^* `for` $U_{ante}^{+\max}$ `and its optimal value` u^*
  foreach $i \notin Opt$ **do**
    if $w_i > u^*$ **then**
      $\delta_i = PesDynProg(T, i)$; `// Call to classical poss. Dyn. Prog. [24] - returns an optimal strategy for the pessimistic utility` U_i^-;
      $u_i = \min\ (w_i, U_i^-(\delta_i))$;
      if $u_i > u^*$ **then** $\delta^* \leftarrow \delta_i$; $u^* \leftarrow u_i$;
  return u^*;

This result allows the use of Multi-Dynamic Programming for the optimization of $U_{ante}^{\max}$. A first idea consists on a direct adaptation of the MultiDynProg (Algorithm 3) by replacing line 6 and 7 respectively by:
line 6': $\delta_i \leftarrow DynProg(T, i)$; `// Call to poss. Dyn. Prog. w.r.t. the agent attitude (Pes or Opt) [24].`
line 7': $u \leftarrow \min(U_i^{\otimes}(\delta_i), w_i)$; `//` $U_i^{\otimes} = U_i^-$ `if` i `is pessimistic and` $U_i^{\otimes} = U_i^+$ `if` i `is optimistic.`

This algorithm (so called HetMultiDynProg) can be improved by considering first the optimistic decision makers and then the pessimistic ones instead of p calls to Dynamic Programming for each one of them. This gives rise to the second version of Multi-Dynamic Programming (so called ImproHetMultiDynProg) outlined by Algorithm 6. In short, this procedure can be described as follows: For optimistic agents the optimization of U_{ante}^{max} comes down to the optimization of U_{ante}^{+max} using Dynamic Programming (either Algorithm 2 or Algorithm 1, since U_{ante}^{+max} = U_{post}^{+max}). Then, we consider only pessimistic agents with an importance degree w_i higher than the current optimal value (obtained for optimistic agents), we compute their pessimistic utilities and select the strategy that maximizes U_{ante}^{max}. Obviously, HetMultiDynProg and ImproHetMultiDynProg provide the same optimal solutions since they are both exact algorithms. However, ImproHetMultiDynProg needs less iterations - this reduces the execution time as it will be shown by the experimental study.

4.2.7. Optimization for heterogeneous agents - the min-based rule

Let us consider the latest awkward criterion $U^{\min}_{ante}$. Since it is not monotonic, Dynamic Programming comes without guarantee of optimality. Thus, to obtain optimal strategy we adapt the Branch and Bound procedure (Algorithm 4) proposed for the optimization of $U^{+\min}_{ante}$ by adjusting the computing of $UpperBound(T, \delta)$. We retain the same principle but here we compute the $U^{\min}_{ante}$ of the best completion of δ: for each agent, $UpperBound(T, \delta)$ builds a strategy δ_i that maximizes $U^{\otimes}_i$ taking into account the agent's attitude w.r.t. uncertainty ($\otimes = \min$ if the agent is pessimistic and $\otimes = \max$ if he is optimistic). Then, it selects among these strategies the one with the highest $U^{\min}_{ante}$. More specifically, we extend Algorithm 5 to deal with heterogeneous decision makers rather than only optimistic ones. Instead of computing U^+_i the optimistic utility for all agents we compute $U^{\otimes}_i$ for each one of them depending on the decision maker attitude ($U^{\otimes}_i = U^+_i$ (resp. U^-_i) if the agent is optimistic (resp. pessimistic)). The modifications concern lines 9 and 21 of the UpperBound function (Algorithm 5) that are respectively replaced by:

line 9': $u_N \leftarrow (u_N \otimes (\lambda_Y \oplus u_Y))$; `// if` i `is Optimistic:` $\otimes = \max$, $\lambda_Y = \pi(Y)$, $\oplus = \min$, `if` i `is Pessimistic:` $\otimes = \min$, $\lambda_Y = 1 - \pi(Y)$, $\oplus = \max$.

line 21': $U_j(\delta_i) \leftarrow Util(\delta_i, j)$; `// Computes for each agent` j `the value of its optimistic or pessimistic utility w.r.t. his/her attitude.`

5. Experiments

This last Section aims at experimenting the feasibility of the exact algorithms proposed, namely (i) Dynamic Programming for $U^{+\max}_{post}$ and $U^{-\min}_{post}$, and also for $U^{+\max}_{ante}$ and $U^{-\min}_{ante}$ because the latters coincide with formers, (ii) Multi-Dynamic Programming for $U^{-\max}_{ante}$ (pure pessimistic agents) and $U^{\max}_{ante}$ (heterogeneous agents), and (iii) Branch and Bound for $U^{+\min}_{ante}$ (pure optimistic agents) and $U^{\min}_{ante}$ (heterogeneous agents).

Beyond a proof of feasibility of these algorithms, our experiments aim at evaluating to what extent the optimization of the problematic (non monotonic) utilities, can be approximated by Dynamic Programming. For homogeneous agents, *ex-post* and *ex-ante* Dynamic Programming algorithms can indeed be used but come without guarantees of optimality - they can be considered as approximation algorithms. However, for heterogeneous agents, the post approach is meaningless and only *ex-ante* Dynamic Programming shall be considered for approximation purposes.

The implementation has been done in Java, on a processor Intel Core $i7$ 2670 QMCPU, 2.2Ghz, 6Gb of RAM. The experiments were performed on complete binary decision trees. We have considered five sets of problems, the number of decisions to be made in sequence (denoted *seq*) varying from 2 to 6, with an alternation of decision and chance nodes: at each decision level l (i.e., odd level), the tree contains 2^{l-1} decision nodes followed by 2^l chance nodes.[3] In the present experiments, the number of agents is set equal to 6 (for heterogeneous agents cases, we set 3 optimistic and 3 pessimistic agents). The utility values as well as the weights degrees are uniformly fired in the set $\{0, 0.1, 0.2, \ldots, 0.9, 1\}$. Conditional possibilities are chosen randomly in $[0, 1]$ and normalized. Each of the five samples of problems contains 1000 randomly generated trees.

5.1. Feasibility analysis and temporal performances

Table 1 presents, for each criterion, the execution time of each possible algorithm. Obviously, whatever the algorithm the CPU time increases with the size of the tree. Dynamic Programming is always below to the threshold of 1 ms, while the Branch and Bound algorithms are more expensive (up to 16 ms) but it remains affordable even for big trees (1365 decision nodes).

For tricky (non monotonic) decision rules, both the exact algorithm(s) and the approximation algorithm(s) are presented. It can be checked that for these rules the approximation Dynamic Programming is always faster than exact algorithms. Unsurprisingly, and whatever the rule tested, the *ex-ante* Dynamic Programming is slightly slower than the *ex-post* Dynamic Programming - both remaining far below the millisecond, in any case. Finally, as to the optimization of U^{max}_{ante}, the experimental results verify that ImproHetMultiDynProg is quicker than HetMultiDynProg - both being exact algorithms.

Furthermore, to study the effects of varying the number of agents, we consider the optimization of $U^{-\max}_{ante}$ and $U^{\max}_{ante}$, for reasonable trees (341 decision nodes) with p agents from 3 to 10, using the more time-consuming algorithm (Branch and Bound). Clearly, as shown in Table 2, the average CPU time with 3 agents, is about 3 milliseconds for $U^{-\max}_{ante}$ and about 4 milliseconds for $U^{\max}_{ante}$. The maximal CPU time for decision trees with 10 agents is less than 11 milliseconds in both cases. Thus, we can say that the results are good enough to allow the handling of real-size problems.

5.2. Quality of approximation of exact algorithms by Dynamic Programming

As previously said, $U^{-\max}_{ante}$ and $U^{+\min}_{ante}$, relative to homogeneous agents, and $U^{\max}_{ante}$ and $U^{\min}_{ante}$, for heterogeneous ones, are not monotonic. For both cases, right optimization is performed using Multi-Dynamic Programming for max-oriented

[3] Hence, for a sequence length $seq = 2$ (resp. 3, 4, 5, 6), the number of decision nodes in each tree of the sample is equal to 5 (resp. 21, 85, 341, 1365).

Table 1
Average CPU time, in milliseconds, according to the size of the tree (in number of decision nodes).

Algorithm		# of decision nodes				
		5	21	85	341	1365
$U_{post}^{-\min}$ $U_{ante}^{-\min}$	Post Dyn. Prog.	0.022	0.026	0.038	0.052	0.106
$U_{post}^{+\max}$ $U_{ante}^{+\max}$	Post Dyn. Prog.	0.024	0.030	0.043	0.060	0.117
$U_{post}^{-\max}$	Post Dyn. Prog.	0.025	0.027	0.039	0.053	0.112
$U_{post}^{+\min}$	Post Dyn. Prog.	0.026	0.028	0.041	0.059	0.110
$U_{ante}^{-\max}$	Multi Dyn. Prog.	0.063	0.074	0.102	0.129	0.605
$U_{ante}^{-\max}$	Ante Dyn. Prog.	0.049	0.065	0.93	0.102	0.446
$U_{ante}^{+\min}$	Branch & Bound	0.359	0.794	2.044	6.095	14.198
$U_{ante}^{+\min}$	Ante Dyn. Prog.	0.032	0.063	0.090	0.114	0.534
$U_{ante}^{\max}$	Het. Multi. Dyn. Prog.	0.068	0.073	0.114	0.136	0.319
$U_{ante}^{\max}$	Impro. Het. Multi. Dyn. Prog.	0.047	0.058	0.079	0.124	0.187
$U_{ante}^{\max}$	Ante Dyn. Prog.	0.053	0.065	0.096	0.149	0.217
$U_{ante}^{\min}$	Het. Branch & Bound	0.420	0.972	2.708	8.483	16.356
$U_{ante}^{\min}$	Ante Dyn. Prog.	0.051	0.071	0.109	0.131	0.206

Table 2
Average CPU time (in milliseconds) for $U_{ante}^{-\max}$ and $U_{ante}^{\max}$ using Branch and Bound algorithms (B&B and Het. B&B) for trees with 341 decision nodes.

	# of agents							
	3	4	5	6	7	8	9	10
$U_{ante}^{-\max}$	3.596	4.394	5.344	6.056	7.137	7.840	8.534	9.204
$U_{ante}^{\max}$	4.520	5.444	7.023	7.824	8.785	9.521	10.457	10.987

Table 3
Quality of approximation of exact algorithms Multi Dyn. Prog. (for $U_{ante}^{-\max}$) and *B&B* (for $U_{ante}^{+\min}$) by *ax-ante* and *ax-post* Dyn. Prog.

Algorithm		# of decision nodes				
		5	21	85	341	1365
% of success						
$U_{ante}^{-\max}$	Ante Dyn. Prog	16.1%	19.8%	23.7%	27.1%	31.9%
$U_{ante}^{-\max}$	Post. Dyn. Prog	17%	24.2%	28.9%	33.7%	39%
$U_{ante}^{+\min}$	Ante Dyn. Prog.	82%	78.6%	71%	65.4%	60.2%
$U_{ante}^{+\min}$	Post Dyn. Prog.	93.2%	91%	89.3%	87.5%	84.7%
Closeness Value						
$U_{ante}^{-\max}$	Ante Dyn. Prog.	0.49	0.54	0.62	0.71	0.80
$U_{ante}^{-\max}$	Post Dyn. Prog.	0.47	0.51	0.59	0.69	0.73
$U_{ante}^{+\min}$	Ante Dyn. Prog.	0.96	0.94	0.92	0.91	0.90
$U_{ante}^{+\min}$	Post Dyn. Prog.	0.97	0.96	0.95	0.94	0.93

aggregation utilities and Branch and Bound for min-oriented ones. For these criteria, Dynamic Programming algorithms can nevertheless be considered as approximation algorithms. The following experiments estimate the quality of these approximations. To this extent, we compute for each sample the success rate of the considered approximation algorithm, i.e., the number of trees for which the value provided by the approximation algorithm is actually optimal (i.e., equals to the one computed by the exact algorithm); then for the trees for which the approximation algorithm fails to reach optimality, we report the average closeness value to $\frac{U_{Approx}}{U_{Exact}}$ where U_{Approx} is the utility of the strategy provided by the approximation algorithm and U_{Exact} is the optimal utility - the one of the solution by the exact algorithm: Namely, Branch and Bound algorithm for $U_{ante}^{+\min}$ and its adaptation for the generalized criterion $U_{ante}^{\min}$ and Multi-Dynamic Programming for $U_{ante}^{-\max}$ and its generalization for $U_{ante}^{\max}$. The results are given in Tables 3 and 4.

Table 4
Quality of approximation of exact algorithms Het. Multi Dyn. Prog. (for $U_{ante}^{\max}$) and Het. B&B (for $U_{ante}^{\min}$) by *ex-ante* Dyn. Prog.

Algorithm		# of decision nodes				
		5	21	85	341	1365
% of success						
$U_{ante}^{\max}$	Ante Dyn. Prog	19%	25.9%	28%	34%	39.6%
$U_{ante}^{\min}$	Ante Dyn. Prog.	85%	81.6%	74%	68.9%	63.4%
Closeness Value						
$U_{ante}^{\max}$	Ante Dyn. Prog	0.44	0.49	0.53	0.58	0.67
$U_{ante}^{\min}$	Ante Dyn. Prog.	0.95	0.95	0.93	0.92	0.91

Clearly, *Ex-Post* Dynamic Programming provides a good approximation for $U_{ante}^{+\min}$ - its success rate decreases with the number of nodes but stay higher than 70%, and above all it has a very high closeness value (above 0.9). Notice that it is always better than its *ex-ante* counterpart, in terms of success rate, of closeness and of CPU time. This is good news since it is polynomial while Branch and Bound, the exact algorithm, is exponential in the number of nodes. As for $U_{ante}^{-\max}$, none of the approximation algorithms is good. However, this is not so bad news since Multi-Dynamic Programming, the exact algorithm is polynomial and has very affordable CPU time.

Finally, regarding the optimization of exact algorithms for $U_{ante}^{\min}$ and $U_{ante}^{\max}$ by Dynamic Programming, the results are quite similar to $U_{ante}^{+\min}$ and $U_{ante}^{-\max}$: Dynamic Programming is a good approximation when competing with the Branch and Bound algorithm, but does not help a lot when Multi-Dynamic Programming can be applied - because the latter is polynomial, even in the case of agents having different attitudes w.r.t. uncertainty.

6. Conclusion

This paper follows a recent work presented in [4–6] for possibilistic collective decision making. We consider more general cases where each decision maker is free to be optimistic or pessimistic and we propose new decision rules for this specific situation that can be seen as a generalization of the *ex-ante* rules. We then consider sequential collective decision problems and complete the algorithmic study introduced in [7] by proposing an adaptation of Multi-Dynamic Programming and Branch and Bound to optimize criteria relative to heterogeneous agents.

This work is a first step in the handling of multi agent (sequential) decision problems. It opens several future directions of research. The first one, which comes along the use of lotteries, is the characterization, in the style of Von Neumann and Morgenstern, of the decision rules considered in this paper. This would complete the axiomatization made in [6] in the style of Savage. The second one is to extend this work, especially for heterogeneous agents, to Hurwicz-like decision criterion that may offer more general results. Besides, it is interesting to focus on decision problems where agents have different knowledge or even cases where agents have simultaneously different knowledge and different attitude w.r.t. uncertainty. Finally, from a more practical point of view, we shall extend this work to more sophisticated (and more compact) qualitative decision models such as possibilistic influence diagrams [24] or possibilistic Markov decision models [41].

Conflict of interest statement

None declared.

Appendix A. Proofs

Proof of Proposition 1. In the following we prove that $U_{ante}^{\min}(L) = 1 - U_{ante}^{\tau\max}(L)$. We can apply the same reasoning to prove that $U_{ante}^{\max}(L) = 1 - U_{ante}^{\tau\min}(L)$.

$$
\begin{aligned}
1 - U_{ante}^{\min}(L) &= 1 - [\min_{i\in\mathcal{A}} \max((1-w_i), \bigotimes_{x_j\in\mathcal{X}} \oplus (u_i(x_j), \Lambda[x_j])))]. \\
&= \max_{i\in\mathcal{A}} 1 - [\max((1-w_i), \bigotimes_{x_j\in\mathcal{X}} \oplus (u_i(x_j), \Lambda[x_j])))]. \\
&= \max_{i\in\mathcal{A}} \min 1 - [((1-w_i), \bigotimes_{x_j\in\mathcal{X}} \oplus (u_i(x_j), \Lambda[x_j])))]. \\
&= \max_{i\in\mathcal{A}} \min(w_i, 1 - [\bigotimes_{x_j\in\mathcal{X}} \oplus (u_i(x_j), \Lambda[x_j]))]. \\
&= \max_{i\in\mathcal{A}} \min(w_i, \overline{\bigotimes}_{x_j\in\mathcal{X}} 1 - [\oplus(u_i(x_j), \Lambda[x_j])]).
\end{aligned}
$$

$$= \max_{i \in \mathcal{A}} \min(w_i, \overline{\bigotimes_{x_j \in \mathcal{X}}} \overline{\oplus} 1 - [(u_i(x_j), \Lambda[x_j])]).$$

$$= \max_{i \in \mathcal{A}} \min(w_i, \overline{\bigotimes_{x_j \in \mathcal{X}}} \overline{\oplus} (1 - u_i(x_j), (1 - \Lambda[x_j]).$$

$$= \max_{i \in \mathcal{A}} \min(w_i, \overline{\bigotimes_{x_j \in \mathcal{X}}} \overline{\oplus} (u_i^{\tau}(x_j), (1 - \overline{\Lambda}[x_j]).$$

$$= U_{ante}^{\tau \max}(L),$$

where if $\otimes$ (resp. $\oplus$) = min then $\overline{\otimes}$ (resp. $\overline{\oplus}$) = max and if $\Lambda[x_j] = L[x_j]$ then $\overline{\Lambda} = 1 - L[x_j]$ and conversely. □

Proof of Proposition 2. This proof shows that $U_{ante}^{+\min}(L) \geq U_{post}^{+\min}(L)$, the one relative to $U_{ante}^{-\max}(L)$ can be obtained in the same way.

Let $u'_i(x) = \max(u_i(x), 1 - w_i)$.

$$U_{post}^{+\min}(L) = \max_{x \in \mathcal{X}} \min(L[x], \min_{i \in \mathcal{A}} \max(1 - w_i, u_i(x))).$$
$$= \max_{x \in \mathcal{X}} \min(L[x], \min_{i \in \mathcal{A}} u'_i(x)).$$
$$= \max_{x \in \mathcal{X}} \min_{i \in \mathcal{A}} \min(L[x], u'_i(x)).$$
$$U_{ante}^{+\min}(L) = \min_{i \in \mathcal{A}} \max(1 - w_i, \max_{x \in \mathcal{X}} \min(u_i(x), L[x])).$$
$$= \min_{i \in \mathcal{A}} \max_{x \in \mathcal{X}} \max(1 - w_i, \min(u_i(x), L[x])).$$
$$= \min_{i \in \mathcal{A}} \max_{x \in \mathcal{X}} \min(\max(1 - w_i, u_i(x)), \ \max(1 - w_i, L[x])).$$
$$= \min_{i \in \mathcal{A}} \max_{x \in \mathcal{X}} \min(u'_i(x), \max(1 - w_i, L[x])).$$

Besides since, $\forall x \in \mathcal{X}$, $\forall i \in \mathcal{A}$, $\max(1 - w_i, L[x]) \geq L[x]$; we have:
(i) $\min_{i \in \mathcal{A}} \max_{x \in \mathcal{X}} \min(u'_i(x), \max(1 - w_i, L[x])) \geq \min_{i \in \mathcal{A}} \max_{x \in \mathcal{X}} \min(u'_i(x), L[x])$.
Let f (x, i) = $\min(u'_i(x), L[x])$, then we have:
$\forall x \in \mathcal{X}$, $\forall i \in \mathcal{A}$, $\max_{x \in \mathcal{X}} f(x, i) \geq f(x, i)$.

$$\min_{i \in \mathcal{A}} \max_{x \in \mathcal{X}} f(x, i) \geq \min_{i \in \mathcal{A}} f(x, i); \forall x \in \mathcal{X}.$$
$$\min_{i \in \mathcal{A}} \max_{x \in \mathcal{X}} f(x, i) \geq \max_{x \in \mathcal{X}} \min_{i \in \mathcal{A}} f(x, i).$$

Then we obtain (ii) $\min_{i \in \mathcal{A}} \max_{x \in \mathcal{X}} \min(u'_i(x), L[x]) \geq \max_{x \in \mathcal{X}} \min_{i \in \mathcal{A}} \min(u'_i(x), L[x])$.

From (i) and (ii) we can deduce that $U_{ante}^{+\min}(L) \geq U_{post}^{+\min}(L)$. □

Proof of Corollary 1. Let L be a possibilistic lottery, $\mathcal{L}$ be the set of possibilistic lotteries, $\mathcal{L}^*$ a subset of $\mathcal{L}$ s.t. $\mathcal{L}^* = \{L_1^*, \ldots, L_p^*\}$ and $L^* \in \mathcal{L}^*$ s.t. $\forall L_i^* \in \mathcal{L}^*$: $\max_{i=1,p} \min(w_i, U_i^-(L^*)) \geq \max_{i=1,p} \min(w_i, U_i^-(L_i^*))$.

Let $L_k \in \mathcal{L}$, $L_k^* \in \mathcal{L}^*$ and we suppose that for any $L_k \in \mathcal{L}$, for any $L_k^* \in \mathcal{L}^* : U_i^-(L_k^*) \geq U_i^-(L_k)$. We have to prove that: $\forall L \in \mathcal{L}, U_{ante}^{-max}(L^*) \geq U_{ante}^{-max}(L)$.

We start by verifying if $\min(w_i, U_i^-(L_k^*)) \geq \min(w_i, U_i^-(L_k))$:

- If $(w_i \leq U_i^-(L_k))$ then: $\min(w_i, U_i^-(L_k^*)) = \min(w_i, U_i^-(L_k)) = w_i$.
- Else if $(w_i \geq U_i^-(L_k))$ then:
  - If $(w_i \leq U_i^-(L_k^*))$, then:
    $(\min(w_i, U_i^-(L_k^*)) = (w_i)) \geq (\min(w_i, U_i^-(L_k)) = (U_i^-(L_k)))$.
  - If $(w_i \geq U_i^-(L_k^*))$ then:
    $(\min(w_i, U_i^-(L_k^*)) = (U_i^-(L_k^*))) \geq (\min(w_i, U_i^-(L_k)) = (U_i^-(L_k)))$.

Hence, $\min(w_i, U_i^-(L_k^*)) \geq \min(w_i, U_i^-(L_k))$.

So, $\max_{i \in \mathcal{A}} \min(w_i, U_i^-(L_k^*)) \geq \max_{i \in \mathcal{A}} \min(w_i, U_i^-(L_k))$.

Since, $\max_{i=1,p} \min(w_i, U_i^-(L^*)) \geq \max_{i=1,p} \min(w_i, U_i^-(L_i^*))$ $\forall L_i^* \in \mathcal{L}^*$ then $U_{ante}^{-\max}(L^*) \geq U_{ante}^{-\max}(L), \forall L$. □

Proof of Corollary 2. Let L be a possibilistic lottery, $\mathcal{L}$ be the set of possibilistic lotteries, $\mathcal{L}^*$ a subset of $\mathcal{L}$ s.t. $\mathcal{L}^* = \{L_1^*, \ldots, L_p^*\}$ and $L^* \in \mathcal{L}^*$ s.t. $\forall L_i^* \in \mathcal{L}^*$: $\max_{i=1,p} \min(w_i, U_i^-(L^*)) \geq \max_{i=1,p} \min(w_i, U_i^-(L_i^*))$.

Let $L_k \in \mathcal{L}$, $L_k^* \in \mathcal{L}^*$ and we suppose that for any $L_k \in \mathcal{L}$, for any $L_k^* \in \mathcal{L}^*$: $U_i^{\otimes}(L_k^*) \geq U_i^{\otimes}(L_k)$. We have to prove that: for any $L \in \mathcal{L}$, $U_{ante}^{\otimes \max}(L^*) \geq U_{ante}^{\otimes \max}(L)$.

We start by verifying if $\min(w_i, U_i^{\otimes}(L_k^*)) \geq \min(w_i, U_i^{\otimes}(L_k))$ where $U_i^{\otimes} = U_i^-$ if the agent i is pessimistic and $U_i^{\otimes} = U_i^+$ if he is optimistic.

- If $(w_i \leq U_j^{\otimes}(L_k)) : \min(w_i, U_i^{\otimes}(L_k^*)) = \min(w_i, U_i^{\otimes}(L_k)) = w_i$.
- If $(w_i \geq U_i^{\otimes}(L_k))$:
  - If $(w_i \leq U_i^{\otimes}(L_k^*)) : \min(w_i, U_i^{\otimes}(L_k^*)) = w_i \geq \min(w_i, U_i^{\otimes}(L_k)) = U_i^{\otimes}(L_k)$.
  - If $(w_i \geq U_i^{\otimes}(L_k^*)) : \min(w_i, U_i^{\otimes}(L_k^*)) = U_i^{\otimes}(L_k^*) \geq \min(w_i, U_i^{\otimes}(L_k)) = U_i^{\otimes}(L_k)$.

Hence, $\min(w_i, U_i^{\otimes}(L_k^*)) \geq \min(w_i, U_i^{\otimes}(L_k))$.

So, $\max_{i \in \mathcal{A}} \min(w_i, U_i^{\otimes}(L_k^*)) \geq \max_{i \in \mathcal{A}} \min(w_i, U_i^{\otimes}(L_k))$.

Since, $\max_{i=1,p} \min(w_i, U_i^{\otimes}(L^*)) \geq \max_{i=1,p} \min(w_i, U_i^{\otimes}(L_i^*))$ $\forall L_i^* \in \mathcal{L}^*$ then $U_{ante}^{\otimes \max}(L^*) \geq U_{ante}^{\otimes \max}(L), \forall L$. □

References

[1] M. Allais, Le Comportement de l'Homme Rationnel devant le Risque: Critique des Postulats et Axiomes de l'Ecole Americaine, Econometrica 21 (4) (1953) 503–546.
[2] R. Bellman, Dynamic Programming, Princeton University Press, 1957.
[3] R. Bellman, A Markovian decision process, Indiana Univ. Math. J. 6 (1957) 679–684.
[4] N. Ben Amor, F. Essghaier, H. Fargier, Solving multi-criteria decision problems under possibilistic uncertainty using optimistic and pessimistic utilities, in: Proceedings of the 15th Conference on Information Processing and Management of Uncertainty, 2014, pp. 269–279.
[5] N. Ben Amor, F. Essghaier, H. Fargier, Décision collective sous incertitude possibiliste. Principes et axiomatisation, Rev. Intell. Artif., Log. Floue Appl. 29 (5) (2015) 515–542.
[6] N. Ben Amor, F. Essghaier, H. Fargier, Egalitarian collective decision making under qualitative possibilistic uncertainty: principles and characterization, in: Proceedings of the 29th AAAI Conference on Artificial Intelligence, 2015, pp. 3482–3488.
[7] N. Ben Amor, F. Essghaier, H. Fargier, Algorithms for multi-criteria optimization in possibilistic decision trees, in: European Conference on Symbolic and Quantitative Approaches to Reasoning with Uncertainty, 2017, pp. 295–305.
[8] N. Ben Amor, H. Fargier, W. Guezguez, Possibilistic sequential decision making, Int. J. Approx. Reason. 55 (2014) 1269–1300.
[9] K. Chatterjee, R. Majumdar, T.A. Henzinger, Markov decision processes with multiple objectives, in: Proceedings of the 26th Symposium on Theoretical Aspects of Computer Science 2006, Springer, Berlin, Heidelberg, 2006, pp. 325–336.
[10] Z. Covaliu, R. Oliver, Representation and solution of decision problems using sequential decision diagrams, Manag. Sci. 41 (12) (1995) 1860–1881.
[11] P. Diamond, Cardinal welfare, individualistic ethics, and interpersonal comparison of utility: comment, J. Polit. Econ. 75 (1967).
[12] D. Dubois, H. Fargier, P. Perny, Qualitative decision theory with preference relations and comparative uncertainty: an axiomatic approach, Artif. Intell. 148 (2003) 219–260.
[13] D. Dubois, H. Fargier, H. Prade, P. Perny, Qualitative decision theory: from Savage's axioms to nonmonotonic reasoning, J. ACM 49 (2002) 455–495.
[14] D. Dubois, L. Godo, H. Prade, A. Zapico, Making decision in a qualitative setting: from decision under uncertainty to case-based decision, in: Proceedings of KR, 1998, pp. 594–607.
[15] D. Dubois, H. Prade, Weighted minimum and maximum operations in fuzzy set theory, J. Inf. Sci. 39 (1986) 205–210.
[16] D. Dubois, H. Prade, Possibility Theory: An Approach to Computerized Processing of Uncertainty (traduction revue et augmentée de Théorie des Possibilités), Plenum Press, 1988.
[17] D. Dubois, H. Prade, The treatment of uncertainty in knowledge-based systems using fuzzy sets and possibility theory, Int. J. Intell. Syst. 3 (1988) 141–165.
[18] D. Dubois, H. Prade, Possibility theory as a basis for qualitative decision theory, in: Proceedings of the International Joint Conference on Artificial Intelligence, 1995, 1995, pp. 1924–1930.
[19] D. Dubois, H. Prade, An introductory survey of possibility theory and its recent developments, J. Jpn. Soc. Fuzzy Theory Syst. 10 (1) (1998) 21–42.
[20] D. Dubois, H. Prade, R. Sabbadin, Qualitative decision theory with Sugeno integrals, in: G.F. Cooper, S. Moral (Eds.), Proceedings of the 13th Conference on Uncertainty in Artificial Intelligence, Morgan Kaufmann, 1998, pp. 121–128.
[21] D. Dubois, H. Prade, R. Sabbadin, Decision theoretic foundations of qualitative possibility theory, Eur. J. Oper. Res. 128 (2001) 459–478.
[22] D. Ellsberg, Risk, ambiguity, and the Savage axioms, Q. J. Econ. 75 (4) (1961) 643–669.
[23] M. Fleming, A cardinal concept of welfare, Q. J. Econ. LXVI (1952) 366–384.
[24] L. Garcia, R. Sabbadin, Possibilistic influence diagrams, in: Proceedings of the 17th European Conference on Artificial Intelligence, 2006, pp. 372–376.
[25] P.H. Giang, P.P. Shenoy, A qualitative linear utility theory for Spohn's theory of epistemic beliefs, in: Proceedings for Uncertainty in Artificial Intelligence, 2000, 2000, pp. 220–229.
[26] Y.Y. Haimes, D. Li, V. Tulsiani, Multiobjective decision-tree analysis, Risk Anal. 10 (1) (1990) 111–127.
[27] J.C. Harsanyi, Cardinal welfare, individualistic ethics, and interpersonal comparisons of utility, J. Polit. Econ. 63 (1955) 309–321.
[28] R.A. Howard, J.E. Matheson, Influence diagrams, Decis. Anal. 2 (3) (2005) 127–143.
[29] G. Jeantet, O. Spanjaard, Rank-dependent probability weighting in sequential decision problems under uncertainty, in: Proceedings of the 18th International Conference on Automated Planning and Scheduling, 2008, pp. 148–155.
[30] R.M. Karp, M. Held, Finite-state processes and dynamic programming, SIAM J. Appl. Math. 15 (3) (1967) 693–718.
[31] D. Kikuti, F.G. Cozman, R.S. Filho, Sequential decision making with partially ordered preferences, Artif. Intell. 175 (7) (2011) 1346–1365.
[32] F.A. Lootsma, Multicriteria decision analysis in a decision tree, Eur. J. Oper. Res. 101 (3) (1997) 442–451.
[33] R. Marinescu, A. Razak, N. Wilson, Multi-objective influence diagrams, CoRR arXiv:1210.4911, 2012.
[34] R. Marinescu, N. Wilson, Order-of-magnitude influence diagrams, CoRR arXiv:1202.3745, 2012.
[35] O. Morgenstern, J. Von Neumann, Theory of Games and Economic Behavior, Princeton University Press, 1944.

[36] T.L. Morin, Monotonicity and the principle of optimality, J. Math. Anal. Appl. 88 (2) (1982) 665–674.
[37] R.B. Myerson, Utilitarianism, egalitarianism, and the timing effect in social choice problems, Econometrica 49 (1981) 883–897.
[38] G. Nemhauser, Introduction to dynamic programming, 01 1967.
[39] Z. Pawlak, Rough sets, Int. J. Comput. Inf. Sci. 11 (1982) 341–356.
[40] H. Raiffa, Decision Analysis-Introductory Lectures on Choices Under Uncertainty, Addison Wesley, Reading, MA, 1970.
[41] R. Sabbadin, Empirical comparison of probabilistic and possibilistic Markov decision processes algorithms, in: Proceedings of European Conference on Artificial Intelligence, 2000, 2000, pp. 586–590.
[42] R. Sabbadin, Possibilistic Markov decision processes, Eng. Appl. Artif. Intell. 14 (3) (2001) 287–300.
[43] R. Sabbadin, H. Fargier, J. Lang, Towards qualitative approaches to multi-stage decision making, Int. J. Approx. Reason. 19 (1998) 441–471.
[44] L. Savage, The Foundations of Statistics, Wiley, 1954.
[45] G. Shafer, A Mathematical Theory of Evidence, Princeton University Press, 1976.
[46] P. Walley, Statistical Reasoning with Imprecise Probabilities, Chapman & Hall, 1991.
[47] L.A. Zadeh, Fuzzy sets as a basis for theory of possibility, Fuzzy Sets Syst. 1 (1978) 3–28.
[48] L.A. Zadeh, A theory of approximate reasoning, in: J.E. Hayes, D. Michie, L.I. Mikulich (Eds.), Machine Intelligence, Elsevier, Amsterdam, 1979, pp. 149–194.