Possibilistic Influence Diagrams

Laurent GARCIA[1] and **Régis SABBADIN**[2]

Abstract. In this article we present the framework of *Possibilistic Influence Diagrams* (PID), which allow to model in a compact form problems of sequential decision making under uncertainty, when only ordinal data on transitions likelihood or preferences are available. The graphical part of a PID is exactly the same as that of usual influence diagrams, however the semantics differ. Transition likelihoods are expressed as possibility distributions and rewards are here considered as satisfaction degrees. Expected utility is then replaced by anyone of two possibilistic qualitative utility criteria for evaluating strategies in a PID. We describe a decision tree-based method for evaluating PID and computing optimal strategies. We then study the computational complexity of PID-related problems (computation of the value of a policy, computation of an optimal policy).

Keywords: decision, possibility theory, causal networks, influence diagrams.

1 INTRODUCTION

For several years, there has been a growing interest in the Artificial Intelligence community towards the foundations and computational methods of decision making under uncertainty. This is especially relevant for applications to planning, where a suitable sequence of decisions is to be found, starting from a description of the initial world, of the available decisions and their effects, and of the goals to reach. Several authors have thus proposed to integrate some parts of decision theory into the planning paradigm. A classical model for decision making under uncertainty is based on *Markov decision processes* (MDP), where actions effects are stochastic and the satisfaction of agents expressed by a numerical, additive utility function. However, classical dynamic programming (DP) algorithms [9] do not scale to the size of problems that can be compactly represented by the factored representations traditionally used in AI. Therefore several methods of aggregation/decomposition for large problems have been proposed in the AI community. Aggregation methods [2], for example, use a compact representation of probability and reward functions involved in the MDP description and propose algorithms dealing explicitly with this representation. These methods are often based on the use of *Influence Diagrams* [6] which form such a factored representation framework for problems of decision under uncertainty (sequential or non sequential).

However, transition probabilities are not always available for representing the effects of actions, especially in Artificial Intelligence applications where uncertainty is often ordinal, qualitative. The same remark applies to utilities: it is often more adequate to represent preference over states simply with an ordering relation rather than with additive utilities. Several authors have advocated this qualitative view of decision making and have proposed qualitative versions of decision theory. [4] propose a qualitative utility theory based on possibility theory, where preferences and uncertainty are both expressed qualitatively. [5, 10] have extended the use of these criteria to sequential decision, allowing the definition of (non-factored) possibilistic MDP and have proposed counterparts of the classical DP solution methods for MDP.

In this paper, after having presented possibilistic counterparts of expected utility (Section 2), we present (Section 3) the framework of *Possibilistic Influence Diagrams* (PID), which allow to model in a compact form problems of sequential decision making under uncertainty, where only ordinal data on transition likelihoods or preferences are available. The graphical part of PID is exactly the same as that of usual influence diagrams, however the semantics differ. Transition likelihoods are expressed by possibility distributions, and rewards are here considered as satisfaction degrees attached to partial goals. Expected utility is then replaced by anyone of the two possibilistic qualitative utility criteria proposed by [4] for evaluating strategies in a PID. Then (Section 4), we describe a decision tree-based method for evaluating PID and computing optimal strategies. We finally show some complexity results on solving PID-related problems (Section 5).

2 POSSIBILISTIC COUNTERPARTS OF EXPECTED UTILITY

[4] propose an ordinal counterpart of the expected utility theory based on possibility theory. The basic measure of possibility theory is the possibility distribution which describes knowledge about the unknown value taken by one or several attributes used to describe states of affairs. It can represent *uncertain knowledge* distinguishing what is plausible from what is less or it can represent *preferences* over desired or less desired states.

In the framework of decision under uncertainty, some state variables are *controlled* by a decision maker who can choose their value, depending on the observation of the values of (all or) some other state variables. The decision maker thus applies a *strategy* δ mapping the set of (observed) uncontrolled state variables values into controlled (decision) variables values. The uncertainty of the agent about the effect of strategy δ is represented by a possibility distribution π_δ mapping the set X of state variables values into a bounded, linearly ordered

[1] LERIA, University of Angers, France. Email: laurent.garcia@info.univ-angers.fr
[2] INRA-BIA Toulouse, France. Email: sabbadin@toulouse.inra.fr

(qualitative) valuation set L. Set L is assumed to be equipped with an order-reversing map n. Let 1_L and 0_L denote the top and bottom elements of L, respectively. Then $n(0_L) = 1_L$ and $n(1_L) = 0_L$. The quantity $\pi_\delta(x)$ thus represents the degree of possibility of the state x being the consequence of strategy δ. $\pi_\delta(x) = 1_L$ means that x is completely plausible, whereas $\pi_\delta(x) = 0_L$ means that it is completely impossible.

It makes sense, if information is qualitative, to represent not only the incomplete knowledge on the state by a possibility distribution π_δ on X but also the decision-maker's preferences on X by means of another possibility distribution μ_δ with values on a preference scale U. Here, we assume that uncertainty and preferences are commensurate, that is U can be identified to L. $\mu_\delta(x) = 1_L$ means that x is completely satisfactory, whereas if $\mu_\delta(x) = 0_L$, it is totally unsatisfactory. Notice that π_δ is normalised (there shall be at least one completely possible state of the world), but μ_δ may not be (it can be that no consequence is totally satisfactory).

[4] proposed the two following qualitative decision criteria for assessing the value of a strategy δ:

$$u^*(\delta) = \max_{x\in X} \min\{\pi_\delta(x), \mu_\delta(x)\}$$
$$u_*(\delta) = \min_{x\in X} \max\{n(\pi_\delta(x)), \mu_\delta(x)\}$$

u^* can be seen as an extension of the *maximax* criterion which assigns to an action the utility of its best possible consequence. On the other hand, u_* is an extension of the *maximin* criterion which corresponds to the utility of the worst possible consequence. u_* measures to what extent every plausible consequence is satisfactory. u^* corresponds to an adventurous (optimistic) attitude in front of uncertainty, whereas u_* is conservative (cautious).

3 POSSIBILISTIC INFLUENCE DIAGRAMS

Possibilistic influence diagrams are the possibilistic counterpart of *influence diagrams* (ID) [6, 11] which extend Bayesian networks to handle decision under uncertainty. A *possibilistic influence diagram* (PID, see Figure 1) is a *directed acyclic graph* DAG containing three different kinds of nodes :

- **chance nodes**, drawn as circles, represent state variables $X_i \in \mathcal{X} = \{X_1, \dots, X_n\}$, as in the *Bayesian Networks* (BN) framework [8]. A combination $x = \{x_1, \dots, x_n\}$ of state variables values represents a state.
- **decision nodes**, drawn as rectangles, represent decision variables $D_i \in \mathcal{D} = \{D_1, \dots, D_p\}$. A combination $d = \{d_1, \dots, d_p\}$ of values represents a decision.
- **utility nodes** $\{V_1, \dots, V_q\}$, drawn as diamonds, represent local "satisfaction degree" functions $r_i \in \{r_1, \dots, r_q\}$.

The directed edges in the PID represent either causal influences or information influences between variables.

Before we describe the underlying semantics of possibilistic influence diagrams (PID), let us give two structural assumptions on the DAG : i) The utility nodes have no children. ii) There exists a directed path comprising all decision nodes[3].

[3] While the first assumption is classical in usual ID, the second one is simply a convention which simplifies the resolution of problems described as ID (possibilistic or not). The directed path induces an ordering of decision variables which will be exploited by the decision tree solution methods.

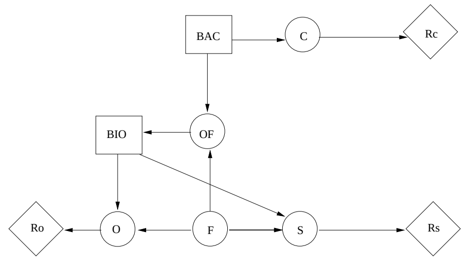

Figure 1. How to make a six-egg omelette from a five-egg one

Now, in addition to the graphical part, which is very similar to that of a classical ID, PID also comprise numerical (ordinal) specifications. Let us recall that, in our possibilistic setting, this numerical ordering is handled in a qualitative way. First, state and decision variables have a finite state space. Furthermore, utility nodes have no state space attached. Then, concerning local dependencies, each state variable X_i has a set $Par(X_i)$ of parents in the DAG. $Par(X_i)$ can comprise both state and decision variables. A conditional possibility table $\Pi(X_i|Par(X_i))$ is attached to every state variable X_i. It specifies every conditional possibility $\Pi(x_i|x_{Par(X_i)}, d_{Par(X_i)})$ where $x_i \in X_i$ [4], $x_{Par(X_i)} \in \mathcal{X}_{Par(X_i)} = \bigotimes_{j, X_j \in Par(X_i)\cap\mathcal{X}} X_j$ and $d_{Par(X_i)} \in \mathcal{D}_{Par(X_i)} = \bigotimes_{j, D_j \in Par(X_i)\cap\mathcal{D}} D_j$. $\mathcal{X}_{Par(X_i)}$ is the Cartesian product of the domains of the state variables belonging to $Par(X_i)$ and $\mathcal{D}_{Par(X_i)}$ is the Cartesian product of the domains of the decision variables belonging to $Par(X_i)$. If X_i is a root of the DAG ($Par(X_i) = \emptyset$) we specify the a priori possibility degrees $\Pi(x_i)$ associated with each instance x_i of X_i.

Now if D_j is a decision node, its value d_j will be fixed by the decision maker. However, incoming links will specify which information is available when the decision is taken, and how the domain of D_j is restricted by the values of the variables in $Par(D_j)$[5]. Furthermore, we adopt the generally accepted *no forgetting assumption*, which implies that all the values of the variables that have been instantiated before d_j is chosen are still known at the time of the choice of d_j.

Solving a PID amounts to compute a *strategy*, allowing to maximise one of the possibilistic qualitative utilities. A strategy for a PID specifies the value of every decision variables as a function of all (or part of) the known values of the variables at the time the decision is taken.

Formally, a strategy is defined as :

Definition 1 (Strategy) *In a PID, a strategy is a function $\delta : \mathcal{X} \to \mathcal{D}$ which can be represented as a set of local functions $\{\delta_j\}_{j\in 1\dots p}$, $\delta_j : \Omega_j \to D_j$.*

[4] X_i will denote both the variable name and its domain, the meaning should be clear from the context. In the same way, $\mathcal{X}_A$, with $A \subseteq \{1, \dots, n\}$ will represent both a set of variables and the Cartesian product of their domains. The same holds for decision variables.

[5] For sake of simplicity of exposition, we will assume in this paper that no restriction to the domain D_j will result from any instantiation of variables in $Par(D_j)$. However, this simplifying assumption is easy to relax, despite the fact that it induces cumbersome notations.

Ω_j denotes the cross product of the domains of all the variables (state and decision) instantiated before D_j. The variables instantiated before D_j (or before any state variable X_i) are called *ancestors* of D_j (of X_i). A strategy δ assigning an instantiation d to any global instantiation x of the state variables can thus be decomposed as a set of *local strategies* $\{\delta_j\}_{j\in 1\ldots p}$, δ_j specifying the value of decision variable D_j, for any possible instantiation of the variables in Ω_j. Notice that since Ω_j can contain decision variables, it is necessary to instantiate the decision variables in a suitable order (parents must be instantiated before their children). The fact that the diagram is a DAG guarantees that this order exists, and the above-mentioned assumption on the existence of a direct path comprising every decision variables guarantees its uniqueness.

Finally, for each utility node V_k, we prescribe ordinal values $\mu_k(x_{Par(V_k)}, d_{Par(V_k)})$ to every possible instantiations $(x_{Par(V_k)}, d_{Par(V_k)})$ of the parent variables of V_k. These values represent satisfaction degrees attached to the local instantiations of the parent variables. It is assumed, analogously to *Flexible Constraint Satisfaction Problems* [3], that the global satisfaction degree $\mu(x, d)$ associated with a global instantiation (x, d) of all variables (state and decision) can be computed as the *minimum* of the local satisfaction degrees : $\mu(x, d) = \min_{k=1\ldots q} \mu_k(x_{Par(V_k)}, d_{Par(V_k)})$.

Now, we are able to compute the optimistic and pessimistic utilities of a strategy δ. Once a strategy δ is chosen, the chance nodes form a possibilistic causal network and thus determine a unique *least specific* joint possibility distribution π_δ over chance nodes interpretations $x = \{x_1, \ldots, x_n\}$. This joint possibility distribution can be computed by applying the *chain rule* [1] :

$$\pi_\delta(x) = \min_{k=1\ldots n} \Pi(x_k|x_{Par(X_k)}, d_{Par(X_k)}),$$

where $d = \delta(x)$. A global satisfaction degree $\mu_\delta(x)$ on state variables instantiations induced by a strategy can also be computed :

$$\mu_\delta(x) = \min_{l=1\ldots q} \mu_l(x_{Par(V_l)}, d_{Par(V_l)}).$$

Notice that $\pi_\delta(x)$ and $\mu_\delta(x)$ depend on the value of state variables only since a strategy δ uniquely defines the value $d = \delta(x)$ from x. It is then possible to compute the possibilistic optimistic and pessimistic qualitative expected utilities of a strategy δ :

$$\begin{aligned} u^*(\delta) &= \max_x \min\{\pi_\delta(x), \mu_\delta(x)\} \\ u_*(\delta) &= \min_x \max\{n(\pi_\delta(x)), \mu_\delta(x)\} \end{aligned} \qquad (1)$$

Consider the example of Figure 1. The problem is to make a six-egg omelette from a five-egg one. The new egg can be fresh or rotten. There are two binary decision variable: break the egg in the omelette BIO ($BIO = yes$ if the egg is put in the omelette and $BIO = false$ if the egg is thrown away) ; break it apart in a cup BAC.

The state variables are the following. C : whether we have a cup to wash ($C = t$) or not ($C = f$). OF : whether we observe that the egg is fresh ($OF = t$) or rotten ($OF = f$) or we do not observe the egg ($OF = u$). F : the “real” state of the egg, fresh ($F = t$) or rotten ($F = f$). O : whether we get a six-egg omelette ($O = 6O$), a five-egg omelette ($O = 5O$) or no omelette at all ($O = nO$). S : whether an egg is spoiled ($S = t$) or not ($S = f$).

Let us take L={0,...,5}. The only a priori possibilities concern F, let us stand for $\Pi(F = t) = 5$ and $\Pi(F = f) = 3$. The conditional possibilities are defined w.r.t. the natural meaning of the problem (here each possibility value is 0 or 5 but it is not always the case) :

$\Pi(C\|BAC)$	t	f
t	5	0
f	0	5

$\Pi(S\|F, BIO)$	(t, f)	(t, t)	$(f, \cdot)$
t	5	0	0
f	0	5	5

$\Pi(OF\|F, BAC)$	(t, t)	(f, t)	$(\cdot, f)$
t	5	0	0
f	0	5	0
u	0	0	5

$\Pi(O\|F, BIO)$	$(\cdot, f)$	(t, t)	(f, t)
$6O$	0	5	0
$5O$	5	0	0
nO	0	0	5

Utility functions μ_O, μ_S and μ_R are defined as follows, and for any global instanciation (x, d), $\mu(x, d) = \min\{\mu_O(o), \mu_S(s), \mu_C(c)\}$.

O	$\mu_O(O)$
$6O$	5
$5O$	3
nO	0

S	$\mu_S(S)$
f	5
t	2

C	$\mu_C(C)$
f	5
t	4

4 SOLUTION METHOD: DECISION TREE

Solving a PID amounts to finding an *optimal* strategy δ^*, i.e. one which maximises the (either optimistic or pessimistic) possibilistic utility. A PID can be unfolded into a *possibilistic decision tree*, using a method very similar to that of [6] for usual ID. A decision tree represents every possible scenarios of successive actions and observations, for *a given ordering* of decision variables[6]. In a possibilistic decision tree, branches issuing from a decision node are labelled with possible instantiations of the decision variable. Branches issuing from a chance node X_i are labelled with an instantiation of the variable and with its possibility degree computed from local transition possibilities together with the instantiations of the ancestors of X_i in the branch.

In order to build the decision tree, an ordering of the variables, compatible with the precedence constraints expressed by the DAG in Figure 1, must be chosen. Figure 2 shows a decision tree obtained by unfolding the PID in Figure 1 with the ordering $\{BAC, F, OF, BIO, O, S, C\}$, which is obviously compatible with the order induced by the DAG.

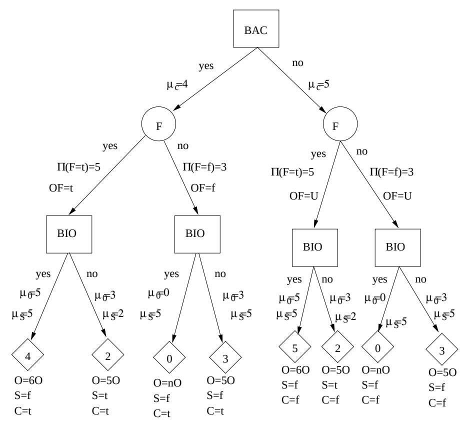

Figure 2. Decision tree associated with the example PID.

---

[6] This is the reason why we choose to fix an ordering of decision variables in a PID.

The tree is then built by successively adding nodes, following the initial ordering. Branches issuing from the node are then added and labelled with the different possible values of the corresponding variable (either states or decisions). Note that in the decision tree of figure 2, after nodes BAC and F have been added, node OF is omitted since it has only one possible value, knowing the values of BAC and F (for example, in the leftmost branch OF is *true* since F is known to be *true* and $BAC = yes$). For the same reasons, nodes C, O and S are not displayed. Then, to each leaf of the decision tree is associated one particular interpretation of the variables. In figure 2, the leftmost branch corresponds to the instantiations $F = t, OF = t, O = 6O, S = f, C = t$ and $BAC = yes, BIO = yes$.

Then, to each leaf are attached both a possibility degree $\pi(x)$ and an aggregated satisfaction degree $\mu(x)$. Following the chain rule, $\pi(x)$ (resp. $\mu(x)$) is the minimum of the transition possibilities (resp. satisfaction degrees) involved in the branch from the root to the leaf. For example, the possibility degree of the leftmost branch of figure 2 is $\pi = \Pi(F = t) = 5$ and its utility is $\mu = \min(\mu_O(6O), \mu_C(t), \mu_S(f)) = 4$.

Now, in order to compute a possibilistic optimal strategy from the decision tree, we apply a *backwards induction* method similar to the one designed in [5]. First, note that choosing a strategy amounts to cut all branches issuing from each decision node, but one (corresponding to the action prescribed by the strategy, the values of the ancestor variables of the decision node being fixed in the branch). Figure 3 shows the pruned tree when strategy ($BAC = no$, $BIO = yes$) is chosen (note that this strategy is unconditional). In the optimistic case, the value of the leaf is $\min(\pi(x), \mu(x))$, hence the value of the leftmost leaf is 4. The utility of a strategy δ is then simply the maximum of the leaves values, corresponding to $u^*(\delta) = \max_{x \in \mathcal{X}} \min\{\pi_\delta(x), \mu_\delta(x)\}$. In Figure 3, the values of the two remaining leaves of the pruned tree are 5 and 0, thus the utility of the chosen strategy is 5. The pessimistic value of the same strategy corresponds to $u_*(\delta) = \min_{x \in \mathcal{X}} \max\{n(\pi_\delta(x)), \mu_\delta(x)\}$. The remaining leaves take values $\max(0, 5) = 5$ and $\max(n(3), 0) = 2$ and the pessimistic utility is the minimum of these leaves values : $\min(5, 2) = 2$.

An optimal strategy can be computed by rolling the full decision tree backwards, keeping only one issuing branch for each decision node, and eliminating unobserved chance nodes:

- If all the children of a decision node already have an attached possibilistic utility, only its issuing branch with the highest value is kept. This highest value is attached to the decision node.
- If all the children of an unobserved chance node already have an attached possibilistic utility, it is replaced with a utility node containing its possibilistic (either optimistic or pessimistic) expected value.

These two steps are iterated until a possibilistic utility is attached to the root node. The remaining tree represents the optimal optimistic strategy, together with its utility. Figure 3 shows the optimal optimistic strategy computed with the omelette example. Note that the computed strategy (which is to break the egg directly in the omelette without using the cup) agrees with the intuition that since the a priori distribution on F makes F more likely than noF and since the decision maker is optimistic, he should not bother testing the freshness of the egg before using it.

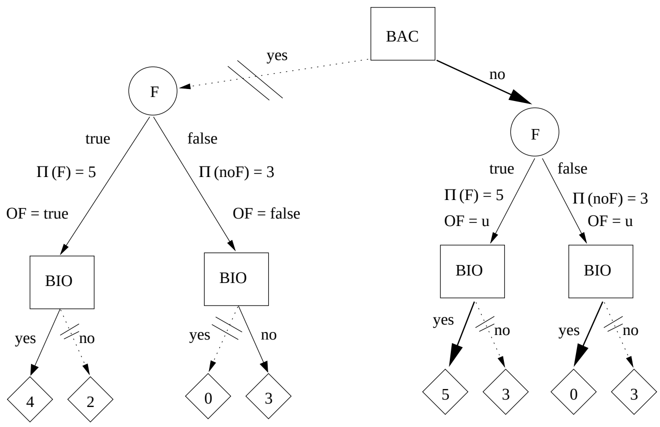

Figure 3. Optimal optimistic decision strategy computation.

5 COMPLEXITY RESULTS

The two basic questions which can be answered within the PID framework are :

1. Assessment of the possibilistic utility of a strategy δ.
2. Computation of an optimal strategy δ^*.

In this section we give the complexity of these problems, both for optimistic and pessimistic utilities. Proofs of Prop. 2, 3, 4 will only be sketched for sake of brevity since they are largely similar to that of Prop. 1. Note that NPO represent the class of optimisation problems "find $\alpha^* = \max_\alpha \ldots$" for which the associated decision problem "is $\alpha^* = (\max_\alpha \ldots) \geq \beta$?" is in NP.

Proposition 1 (Optimistic strategy evaluation (OSE)) *A PID and a strategy δ being given, the problem of computing $u^*(\delta)$ is in NPO and is NP-hard.*

Proof : OSE is a maximisation problem, since $u^*(\delta) = \max_x \min\{\pi_\delta(x), \mu_\delta(x)\}$. OSE is in NPO since it can be solved in polynomial time by the following non-deterministic algorithm, where step 2 can obviously be performed in polynomial time:
$\forall \alpha \in L$,

1. guess x,
2. if $\min\{\pi_\delta(x), \mu_\delta(x)\} \geq \alpha$, return $YES(\alpha)$, else $NO(\alpha)$.
3. $u^*(\delta) = \max\{\alpha \text{ s.t. } YES(\alpha) \text{ holds}\}$.

Now, in order to show that OSE is NP-hard, we exhibit a polynomial transformation from 3-SAT to OSE. Let 3-SAT be defined as a set $\{v_1, \ldots, v_n\}$ of variables and $\{\psi_1, \ldots, \psi_m\}$ be a set of 3-clauses. Let us define the following PID, where $\mathcal{X} = \{X_1, \ldots, X_n\}$ are binary variables, and the set of utility nodes is $\{r_0, r_1, \ldots, r_m\}$. $\mathcal{D} = \{D_0\}$, where D_0 has domain $\{t, f\}$. Variables X_i have no parents and their unconditional possibility distributions are defined as $\Pi(X_i = t) = \Pi(X_i = f) = 1_L, \forall i$. There are arcs from X_i to r_j if and only if $v_i \in Scope(\psi_j)$. The local utility functions are the following : $\forall j > 0$, $\mu_j(x_{Par(r_j)}) = 1_L$ if $x_{Par(r_j)}$ satisfies ψ_j and 0_L else. D_0 is not connected to any other node. Then, if we fix δ

to $d_0 = t$ (δ is unconditional) and if we fix $\alpha = 1_L$, then obviously the utility of any δ is 1_L if and only if $\{\psi_1, \ldots, \psi_m\}$ is satisfiable. So, this OSE solves 3-SAT for any strategy δ. Furthermore, the transformation is polynomial in time and space since utility tables have at most 2^3 elements (the ψ_j are 3-clauses). □

Proposition 2 (Optimistic strategy optimisation (OSO)) *A PID being given, the problem of computing an optimal optimistic policy δ^* and its utility $u^*(\delta)$ is in NPO and is NP-hard.*

Sketch of proof : Just notice that the OSO problem can be stated equivalently as the following maximisation problem : Find $U^* = \max_d \max_x \min\{\Pi(x|d), \mu(x,d)\}$. So, similarly to the OSE case, the problem will be here to guess interpretations (x,d) (instead of x only) for all fixed $\alpha \in L$, and to keep the "best one" : (x^*, d^*). Thus, OSO is NPO. OSO is of course NP-hard, since OSE is. Note that δ^* is only defined in x^* ($\delta^*(x^*) = d^*$) and can take any value elsewhere. □

Remark that it is obvious, from the proof of Prop. 2 that finding an optimal policy in an optimistic decision problem amounts to finding the branch with maximal leaf utility in the corresponding decision tree.

Proposition 3 (Pessimistic strategy evaluation (PSE)) *A PID and a strategy δ being given, the problem of computing $u_*(\delta)$ is in NPO and is NP-hard.*

Sketch of proof : PSE is a *minimisation* problem. $u_*(\delta) = \min_x \max\{n(\pi_\delta(x)), \mu_\delta(x)\}$. The same reasoning as for OSE can be used to show that PSE is NPO:
$\forall \alpha \in L$,

1. guess x,
2. if $\max\{n(\pi_\delta(x)), \mu_\delta(x)\} \leq \alpha$, return $YES(\alpha)$, else $NO(\alpha)$.
3. $u^*(\delta) = \min\{\alpha \text{ s.t. } YES(\alpha) \text{ holds}\}$.

To show NP-hardness, we use the same transformation as for OSE, except that $\forall j > 0$, $\mu_j(x_{Par(r_j)}) = 0_L$ if $x_{Par(r_j)}$ satisfies ψ_j and 1_L else. Then the logic base will be satisfiable if and only if $u_* = 0_L$ (and not 1_L). □

The problem, when solving a PSE, amounts to finding the branch in the pruned decision tree (a strategy being fixed) with *minimum* leaf utility.

The pessimistic strategy optimisation problem is more complex since the corresponding exploration of the (complete) decision tree alternates maximisation at decision nodes and minimisation at state nodes. The following result can be shown:

Proposition 4 (Pessimistic strategy optimisation (PSO)) *A PID being given, the problem of computing an optimal pessimistic policy δ^* and its utility $u_*(\delta)$ is $\sum_2^p$-complete.*

Sketch of proof : We want to compute: $\max_\delta u_*(\delta) = \max_\delta \min_x \max\{n(\pi_\delta(x), \mu_\delta(x))\}$, which we can write $\max_\delta u_*(\delta) = \max_d \min_x \max\{n(\Pi(x|d)), \mu(x,d)\}$. Let $\alpha \in L$ be given, " is $\max_\delta u_*(\delta) \leq \alpha$?" is equivalent to "is it true that $\exists x, \forall d, \max\{n(\Pi(x|d), \mu(x,d))\} \leq \alpha$?". This question, of the form "is it true that $\exists x, \forall y, r(x,y)$?" is known to be $\sum_2^p$-complete. So, PSO can be solved by solving at most $|L|$ $\sum_2^p$-complete decision problems and is thus $\sum_2^p$-complete itself. □

6 CONCLUSION

In this article we have described *Possibilistic Influence Diagrams* (PID), which allow to model in a compact form problems of sequential decision making under qualitative uncertainty in the framework of possibility theory. We have described a decision-tree based solution method for PID and we have given computational complexity results for several questions related to PID. Even though our complexity results seem negative in a sense, PID-related problems being at least NP-hard, these results have to be mitigated by comparing this complexity to that of structured stochastic MDP, shown to be EXP-hard [7], i.e. provably not in the polynomial hierarchy!

We now plan to work in the two following directions. First, even if we know that PID solving is at best NP-hard, we may look for more efficient solution methods than decision tree "brute force" exploration. We can either explore methods derived from stochastic ID [11, 12], such as *variable elimination* and to exploit specific features of PID. We can also try to exploit methods of exploration of *minimax* trees, since the decision tree obtained in the pessimistic case is a minimax tree. Another possible direction is to study more particularly structured *possibilistic Markov decision processes* [10] for which specific PID representations should be defined, representing explicitly the time-structure of the process (in the way 2-DBN are used to model structured MDP in [2]). This would lead to define structured versions of the possibilistic *value iteration* and *policy iteration* algorithms proposed in [10].

REFERENCES

[1] S. Benferhat, D. Dubois, L. Garcia and H. Prade (2002). On the transformation between possibilistic logic bases and possibilistic causal networks. *Int. J. of Approximate Reasoning*, 29, 135-173.

[2] C. Boutilier, R. Dearden and M. Goldszmidt (2000). Stochastic Dynamic Programming with Factored Representations. *Artificial Intelligence*, 121:1, 49-107.

[3] D. Dubois, H. Fargier and H. Prade (1996). Possibility Theory in Constraint Satisfaction Problems: Handling Priority, Preference and Uncertainty, *Applied Intelligence* 6, 287-309.

[4] D. Dubois and H. Prade (1995). Possibility theory as a basis for qualitative decision theory. *Proc. of IJCAI'95*, 1925-1930.

[5] H. Fargier, J. Lang and R. Sabbadin (1998). Towards qualitative approaches to multi-stage decision making. *Int. J. of Approximate Reasoning*, 19, 441-471.

[6] R.A. Howard and J.E. Matheson (1984). Influence Diagrams. *Readings on the Principles and Applications of Decision Analysis*, 2, 719-762. Menlo Park, CA.

[7] C. Lusena, J. Goldsmith and M. Mundhenk (2001). Nonapproximability. Results for POMDP. *J. of Art. Int. Research*, 14, 83-103.

[8] J. Pearl (1988). *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufmann.

[9] M.L. Puterman (1994). *Markov Decision Processes*. John Wiley and Sons.

[10] R. Sabbadin (2001). Possibilistic Markov decision processes. *Engineering Appl. of Artificial Intelligence*, 14, 287-300.

[11] R.D. Schachter (1986). Evaluating influence diagrams. *Operations research*, 34(6), 871-882.

[12] N. L. Zhang (1998). Probabilistic inference in influence diagrams. *Computational Intelligence*, 14:475-497/