A first order, four-valued, weakly paraconsistent logic and its relation with rough sets semantics

Alexis Tsoukiàs[1]

Abstract. A first order four-valued logic, called DDT, is presented in the paper as an extension of Belnap's logic using a weak negation and establishing an appropriate semantic for the predicate calculus. The logic uses a simple algebraic structure, that is the smallest non trivial interlaced bilattice on the four truth values, thus resulting in a boolean algebra on the set of truth values. The logic is a language for reasoning under uncertainty, enabling to capture hesitation due either to inconsistent or incomplete information, while keeping a clear distinction between these epistemic states. The logic was originally developed for preference modelling purposes (for which a brief account is given in the paper). The paper demonstrates and discusses the equivalence between the semantics of this logic and of rough sets semantics. On this basis, this papers presents the possibility of inducing rules from examples, that can be integrated in systems whose inference is expressed in the above logic. Such an approach enhances the potentialities of the use of rough sets in classification, reasoning and decision support.

1 Introduction

Contradictory information is a common situation in real life and in everyday human reasoning. Moreover humans are normally able to act both under such "contradictory" situations and in "absence" of information. From this perspective it is known that classical logic fails to be a good representation of human reasoning since any inconsistency allows the deduction of everything and absence of information simply is not considered. Classical logic enables to deduce all the possible theorems from a given set of sentences automatically. The introduction of new information (in the form of a new sentence) will change nothing (if the sentence is consistent with the already given set) or will destroy the conclusions (if it is inconsistent).

The problem of reasoning under inconsistency was faced in paraconsistent logics (see da Costa, 1974; Rescher and Brandom, 1980). The problem of inconsistent information is also present in Pawlak's (1982) seminal work on rough sets (see also Pawlak, 1991). Trying to describe objects under a set of attributes may result in ambiguous classification since identical descriptions may correspond to objects belonging to different categories.

The paper aims to present a first order extension of Belnap's four- valued logic (see Belnap 1976 and 1977) with strong connectives, which is weakly paraconsistent, as well as some applications. A specific claim we demonstrate in the paper is the equivalence between rough sets and the semantics of the logic presented. The paper is organised as follows. In section 1 the basic idea of the four-valued logic is presented and its underlying algebraic properties are discussed. In section 2 the first order extension of Belnap's logic with strong connectives is defined. In section 3 some relevant properties of the logic are presented. In section 4, a brief account of the application of the logic for preference modeling purposes is presented. In section 5, we demonstrate and discuss the equivalence between the logic's semantics and rough sets. Some open problems are discussed at the end of the paper.

2 Four truth values

The four values introduced by Belnap in his two seminal papers (Belnap 1976 and 1977) are of a clear epistemic nature. Actually these truth values represent different states where an agent (natural or artificial) may find himself/herself when asked to answer a query. Given a sentence ϕ, the agent may have been told that "ϕ holds", that "ϕ does not hold", both or nothing. The problem is how the agent should react in any of these cases, independently of the ontology of ϕ since (s)he is obliged to provide an answer. The logic presented tries to model the epistemic nature of reasoning without introducing epistemic operators such as the modal ones of knowledge and belief. The basic idea is to characterise some basic states in which an agent may find himself/herself through a four valued valuation of his/her language. The four values are:
- true (t): there is evidence that is true and there is no evidence that is false;
- false (f): there is evidence that is false and there is no evidence that is true;
- both (k): there is evidence that is true and false;
- unknown (u): there is no evidence that is either true or false;
and we define the four corresponding epistemic states as the "true" one, the "false" one, the "contradictory" one and the "unknown" one. The logic we develop will therefore be a calculus on epistemic states and not on the ontology of the language.

2.1 Lattices and Bilattices

Let us first introduce some basic definitions and notations (see also Ginsberg 1988, Fitting, 1991) limited to complete lattices.

Definition 2.1 *A complete lattice L is a triplet $\langle T, \sqcup, \sqcap \rangle$ where T is a partially ordered set, $\sqcup$ and $\sqcap$ being the joint and the meet operators, respectively.*

Therefore we have:

$$\exists \succeq \ \subseteq T \times T: \ \forall x, y \in T: x \sqcup y = glb_\succeq(x, y) \quad x \sqcap y = lub_\succeq(x, y)$$

Definition 2.2 *A complete bilattice B is a 5-tuple $\langle T, \sqcup, \sqcap, +, \bullet \rangle$ where T is a twice partially ordered set, $\sqcup$ and $\sqcap$ being the joint and the meet operators of the first partial order, respectively, and $+$ and $\bullet$ being the joint and meet operators of the second partial order, respectively.*

Therefore we have:

$$\exists \succeq_1, \succeq_2 \ \subseteq T \times T: \ \forall x, y \in T:$$

$$x \sqcup y = glb_{\succeq_1}(x, y) \quad x \sqcap y = lub_{\succeq_1}(x, y)$$

$$x + y = glb_{\succeq_2}(x, y) \quad x \bullet y = lub_{\succeq_2}(x, y)$$

We restrict the field by considering a particular class of bilattices known as "interlaced bilattices" (see Fitting, 1991).

Definition 2.3 *An interlaced bilattice is a complete bilattice such that meets and joints of one order are monotone with respect to the other order of the bilattice. That is:*

$$\forall x, y, z, w \in T, \ \ x \succeq_1 y \text{ and } z \succeq_1 w, \text{ then } x \bullet z \succeq_1 y \bullet w$$

$$\forall x, y, z, w \in T, \ \ x \succeq_2 y \text{ and } z \succeq_2 w, \text{ then } x \sqcap z \succeq_2 y \sqcap w$$

The concept of monotonicity is introduced as a basic condition for a bilattice to be interlaced. Interlacity is the minimum property of a bilattice to ensure it is not just two lattices stuck together. However, the concept of monotonicity will also be used in order to define basic transformations of a lattice (and a bilattice). In Scott's work (1972, 1982) on "approximation" lattices (mathematically equivalent to complete lattices) the concept of "continuity" is introduced as a necessary property of a function in order to be accepted as a transformation on the lattice. In the discrete case (as in this case) continuity reduces to monotonicity. Such a property is important as monotonic transformations are the only ones that preserve the order in a lattice. We can therefore define some properties of the basic unary transformations of an interlaced bilattice (keeping in mind monotonicity).

Definition 2.4 *Given an interlaced bilattice B:*
$\mathcal{N}_1 : T \mapsto T$ *is a monotone transformation on $\succeq_1$* **iff**

$$\forall x, y, \ \ x \succeq_1 y \ \Leftrightarrow \ \mathcal{N}_1(x) \succeq_1 \mathcal{N}_1(y)$$

$\mathcal{N}_2 : T \mapsto T$ *is a monotone transformation on $\succeq_2$* **iff**

$$\forall x, y, \ \ x \succeq_2 y \ \Leftrightarrow \ \mathcal{N}_2(x) \succeq_2 \mathcal{N}_2(y)$$

$\mathcal{I} : T \mapsto T$ *is an interlaced monotone transformation on $\succeq_1$ and $\succeq_2$* **iff**

$$\forall x, y, \ \ x \succeq_2 y \ \Leftrightarrow \ \mathcal{I}(x) \succeq_1 \mathcal{I}(y)$$

$$\forall x, y, \ \ x \succeq_1 y \ \Leftrightarrow \ \mathcal{I}(x) \succeq_2 \mathcal{I}(y)$$

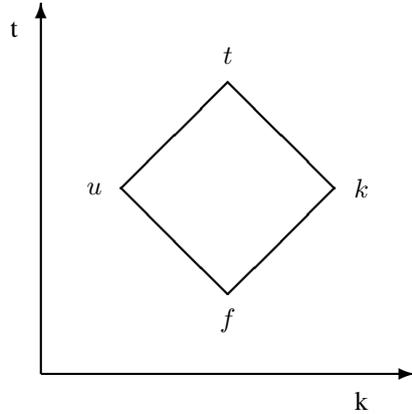## 2.2 Lattice representation of four truth values

Figure 1. The smallest non trivial interlaced bilattice Λ.

Using Scott's results on approximation lattices (see Scott 1972 and 1982) Belnap (1976 and 1977) ordered the four truth values on two lattices one named "information" lattice, the other "truth" lattice. Not surprisingly, these two lattices form the smallest non trivial interlaced bilattice (see Ginsberg 1988 and Fitting, 1991). Such a bilattice is shown in figure 1 and denoted as the bilattice Λ. Following the information order (the k one) we read $x \succeq_k y$ as "y approximates the information least as well as x". The glb_k is the value u and the lub_k is the value k. Following the t truth order we read $x \succeq_t y$ as "y is true at least as x is". The glb_t is the value f and the lub_t is the value t. In this context, negations are monotone transformations on a lattice with the duality property, that is $\mathcal{H}$ is a negation **iff** it is a monotone transformation on the bilattice (see definition 1.4) and, $\forall x, \in B \quad \mathcal{H}(\mathcal{H}(x)) = x$ (duality property). In fact, imposing the monotonicity of negation is the only way to preserve the structure of the bilattice and its interlaced property.

Belnap developed his propositional logic using a monotone transformation on the k lattice as a negation and as basic binary connectives, the conjunction which corresponds to the meet on the t lattice and the disjunction which corresponds to the joint on the t lattice. He then defined implication as a two-valued binary connective such that "$x \rightarrow y$" is true iff $x \succeq_t y$ and false otherwise.

Such a logic however, lacks any specific semantics and is too weak to be used as calculus. Following the pioneering work of Dubarle (1963), we therefore tried to develop a stronger logic which could allow a first order calculus and connectives strong enough to represent both four valued and two valued sentences.

The basic extensions made to the propositional logic introduced by Belnap are the following.

1. Introduce a weak negation $\not\sim$ (to be read *"perhaps not"*, which is an interlaced monotone transformation of the Λ with duality. We therefore have the usual strong negation

$\neg$ as defined by Belnap as well as a weak negation. The truth tables of the two negations are shown in table 1 (on the use of two negations, see also Fages and Ruet, 1997).

α	t	k	u	f
$\neg\alpha$	f	k	u	t
$\not\sim\alpha$	k	t	f	u

Table 1. The truth tables of the two negations.

2. Define implication "$\rightarrow$" as follows:
   $\alpha \rightarrow \beta =_{def} \neg\not\sim\neg\not\sim\alpha \vee \beta$
   The reasons for such a definition will become clearer in the next section.

The resulting logic is a functionally complete propositional logic as has already been shown by Dubarle (1963) and it corresponds to a Boolean algebra on the bilattice Λ. Ruet (1996) also demonstrated the soundness and completeness of a practically equivalent logic (see also Fages and Ruet, 1997; the difference is only semantic, the weak negation in DDT is the complement of the weak negation introduced by Ruet, 1996). A similar functionally complete four valued logic was proposed by Bergstra et al., (1995), while extensions of Belnap's logic can be found in Font and Moussavi, (1993) and Kaluzhny and Muravitsky (1993).

3 A first order four valued logic

3.1 Syntax

An alphabet of the first order language $\mathcal{L}$, henceforth called DDT, consists of (for a preliminary version, see Doherty et al., 1992):
- a denumerable set of *individual variables* (possibly subscripted): $x_1, x_2 \cdots y_1, y_2 \cdots$
- the *logical connectives* "$\vee$" (or), "$\wedge$" (and), "$\rightarrow$" (implication), "$\sim$" (complementation), "$\not\sim$" (weak negation) and "$\neg$' (strong negation),
- the *unary operators* "**T**" (true), "**F**" (false), "**U**" (unknown), "**K**" (both), "$\triangle$" (presence of truth),
- the *quantifiers* "$\forall$" (for all) and "$\exists$" (exists),
- the constants $\mathcal{T}, \mathcal{K}, \mathcal{U}, \mathcal{F}$,
- the symbols "(" and ")" serving as punctuation,
- a countable set of *predicate constants* (i, p, q, r, ...) of positive arity.
We use greek letters $\alpha, \beta, \gamma, \cdots$ to represent general formula of the language. Well-formed formula are defined as usual. If α, β are wff, then $\neg\alpha$, $\not\sim\alpha$, $\mathbf{T}\alpha$, $\alpha\wedge\beta$, $\alpha\vee\beta$ etc. are wff.

In the following we give the truth tables of the principal connectives. In Table 2, we provide the truth tables of the negations and their combination. In table 3, we provide the truth tables of the three basic binary operations, the conjunction, the disjunction and the implication.

α	$\nmid\alpha$	$\sim\nmid\alpha$	$\sim\alpha$	$\neg\alpha$	$\neg\nmid\alpha$	$\neg\sim\nmid\alpha$	$\neg\sim\alpha$
t	k	u	f	f	k	u	t
k	t	f	u	k	f	t	u
u	f	t	k	u	t	f	k
f	u	k	t	t	u	k	f

Table 2. The truth tables of $\neg$, $\nmid$ and $\sim$ and their combinations.

From an algebraic point of view these eight combinations represent one of the Sylow subgroups of the group of all permutations of four elements and precisely the one preserving complementarity between t and f and between k and u. Under such a property it is easy to observe that the "complementation" can be defined through the other two negations. The following identities are true (the proofs are trivial from the truth table).

$$\sim\alpha \equiv \neg\nmid\neg\nmid\alpha \equiv \nmid\neg\nmid\neg\alpha$$
$$\neg\neg\alpha \equiv \alpha$$
$$\sim\sim\alpha \equiv \alpha$$
$$\nmid\nmid\alpha \equiv \alpha.$$

This is not a surprising result. Actually, only $\neg$ and $\nmid$ are negations (fulfilling monotonicity and duality) while $\sim$ (which is not a monotone transformation) should be viewed as an abbreviation of $\neg\nmid\neg\nmid$ which represents in turn the complement on the bilattice. It is easy to observe that the negation corresponding to the monotone transformation on the t lattice can be defined as the sequence $\nmid\neg\nmid$. Moreover, the implication introduced corresponds to the conventional strong monotonic implication. In fact $\alpha\rightarrow\beta$ should be read as *"either the complement of α or β"*.

We now introduce the truth tables for the basic binary operators.

$\wedge$	u	f	t	k
u	u	f	u	f
f	f	f	f	f
t	u	f	t	k
k	f	f	k	k

$\vee$	u	f	t	k
u	u	u	t	t
f	u	f	t	k
t	t	t	t	t
k	t	k	t	k

$\rightarrow$	t	k	u	f
t	t	k	u	f
k	t	t	u	u
u	t	k	t	k
f	t	t	t	t

Table 3. The truth tables of $\wedge$, $\vee$ and $\rightarrow$.

A two-valued fragment of the language, called DDT^2, can be created by introducing some strong unary operators. They are defined as follows:
- $\mathbf{T}\alpha =_{def} \alpha\wedge\sim\neg\alpha$.
- $\mathbf{F}\alpha =_{def} \sim\alpha\wedge\neg\alpha$.
- $\mathbf{U}\alpha =_{def} \sim\nmid\alpha\wedge\neg\nmid\alpha$.
- $\mathbf{K}\alpha =_{def} \nmid\alpha\wedge\nmid\neg\alpha$.
- $\triangle\alpha =_{def} \mathbf{T}\alpha\vee\mathbf{K}\alpha$.
- $\triangle\neg\alpha =_{def} \mathbf{F}\alpha\vee\mathbf{K}\alpha$.

The truth tables for the defined operators are presented in table 4. It is easy to verify that $\triangle\alpha \equiv \mathbf{T}(\alpha\vee\not\sim\alpha)$ and $\triangle\neg\alpha \equiv \mathbf{T}(\neg\alpha\vee\not\sim\neg\alpha)$.

α	$\mathbf{T}\alpha$	$\mathbf{K}\alpha$	$\mathbf{U}\alpha$	$\mathbf{F}\alpha$	$\triangle\alpha$	$\triangle\neg\alpha$
t	t	f	f	f	t	f
k	f	t	f	f	t	t
u	f	f	t	f	f	f
f	f	f	f	t	f	t

Table 4. Truth Tables for the strong unary operators.

3.2 Semantics

The logic introduced deals with uncertainty. A set A may be defined, but the membership of an object a to the set may not be certain either because the information is not sufficient or because the information is contradictory.

In order to distinguish these two principal sources of uncertainty, the knowledge about the *"membership"* of a to A and the *"non-membership"* of a to A are evaluated independently since they are not necessarily complementary. From this point of view, from a given knowledge, we have two possible entailments, one positive, about membership and one negative, about non-membership. Therefore, any predicate is defined by two sets, its positive and its negative extension in the universe of discourse. Since the negative extension does not necessarily correspond to the complement of the positive extension of the predicate we can expect that the two extensions possibly overlap (due to the independent evaluation) and that there exist parts of the universe of discourse that do not belong to either of the two extensions. The four truth values capture these situations.

More formally we have:

A *similarity type* ρ is a finite set of predicate constants R, where each R has an arity $n_R \leq \omega$. Every alphabet uniquely determines a class of *formulas*. Relative to a given similarity type ρ, $R(x_1, \ldots, x_m)$ is an atomic formula iff $x_1, \ldots, x_m$ are individual variables, $R \in \rho$, and $n_R = m$. Similarly, $(x = y)$ is an atomic formula iff x and y are variables. The definitions of $\mathcal{L}[\rho]$ formulas, free variables, etc. are defined in the usual way. In this paper, formulas are denoted by the letters α, β, γ, $\cdots$, possibly subscripted.

A *structure* or *model* M for similarity type ρ consists of a non-empty domain $|M|$ and, for each predicate symbol $R \in \rho$, an ordered pair $R^M = \langle R^{M^+}, R^{M^-}\rangle$ of sets (not necessarily a partition) of n_R-tuples from $|M|$. In fact, an individual can be in the two sets or in neither of them.

A *variable assignment* is a mapping from the set of variables to objects in the domain of the model. Capital letters from the beginning of the alphabet are used to represent variable assignments.

The truth definition for DDT is defined via two semantic relations, $\models_t$ (true entailment) and $\models_f$ (false entailment), by simultaneous recursion as in the following definition (due to the structure introduced in the case of "not true entailment" $\not\models_t$ does not coincide with the

false entailment and in the case of "not false entailment" $\not\models_f$ does not coincide with the true entailment). Each formula is univocally defined through its model whichis however, a couple of sets, the "positive" and "negative" extensions of the formula.

Definition 3.1 *Let M be a model structure and A a variable assignment.*

1. $M\models_t R(x_1,\ldots,x_n)[A]$ *iff* $\langle A(x_1),\ldots,A(x_n)\rangle \in R^{M^+}$.

2. $M\models_f R(x_1,\ldots,x_n)[A]$ *iff* $\langle A(x_1),\ldots,A(x_n)\rangle \in R^{M^-}$.

3. $M\not\models_t R(x_1,\ldots,x_n)[A]$ *iff* $\langle A(x_1),\ldots,A(x_n)\rangle \in |M| \setminus R^{M^+}$.

4. $M\not\models_f R(x_1,\ldots,x_n)[A]$ *iff* $\langle A(x_1),\ldots,A(x_n)\rangle \in |M| \setminus R^{M^-}$.

5. $M\models_t \neg\alpha[A]$ *iff* $M\models_f \alpha[A]$.

6. $M\models_f \neg\alpha[A]$ *iff* $M\models_t \alpha[A]$.

7. $M\not\models_t \neg\alpha[A]$ *iff* $M\not\models_f \alpha[A]$.

8. $M\not\models_f \neg\alpha[A]$ *iff* $M\not\models_t \alpha[A]$.

9. $M\models_t \not\prec\alpha[A]$ *iff* $M\models_t \alpha[A]$.

10. $M\models_f \not\prec\alpha[A]$ *iff* $M\not\models_f \alpha[A]$.

11. $M\not\models_t \not\prec\alpha[A]$ *iff* $M\not\models_t \alpha[A]$.

12. $M\not\models_f \not\prec\alpha[A]$ *iff* $M\models_f \alpha[A]$.

13. $M\models_t (\alpha\vee\beta)[A]$ *iff* $M\models_t \alpha[A]$ *or* $M\models_t \beta[A]$.

14. $M\models_f (\alpha\vee\beta)[A]$ *iff* $M\models_f \alpha[A]$ *and* $M\models_f \beta[A]$.

15. $M\not\models_t (\alpha\vee\beta)[A]$ *iff* $M\not\models_t \alpha[A]$ *and* $M\not\models_t \beta[A]$.

16. $M\not\models_f (\alpha\vee\beta)[A]$ *iff* $M\not\models_f \alpha[A]$ *or* $M\not\models_f \beta[A]$.

17. $M\models_t (\alpha\wedge\beta)[A]$ *iff* $M\models_t \alpha[A]$ *and* $M\models_t \beta[A]$.

18. $M\models_f (\alpha\wedge\beta)[A]$ *iff* $M\models_f \alpha[A]$ *or* $M\models_f \beta[A]$.

19. $M\not\models_t (\alpha\wedge\beta)[A]$ *iff* $M\not\models_t \alpha[A]$ *or* $M\not\models_t \beta[A]$.

20. $M\not\models_f (\alpha\wedge\beta)[A]$ *iff* $M\not\models_f \alpha[A]$ *and* $M\not\models_f \beta[A]$.

21. $M\models_t \forall x\alpha[A]$ iff $M\models_t \alpha[A']$ for all A' differing with A at most at x.

22. $M\not\models_t \forall x\alpha[A]$ iff $M\not\models_t \alpha[A']$ for all A' differing with A at most at x.

23. $M\models_f \forall x\alpha[A]$ iff $M\models_t \alpha[A']$ for an A' differing with A at most at x.

24. $M\not\models_f \forall x\alpha[A]$ iff $M\not\models_t \alpha[A']$ for an A' differing with A at most at x.

It is now possible to introduce an evaluation function $v(\alpha)$ mapping $\mathcal{L}$ in to the set of truth values $\{t, k, u, f\}$ as follows:

- $v(\alpha) = t$ iff $M\models_t \alpha[A]$ and $M\not\models_f \alpha[A]$
- $v(\alpha) = k$ iff $M\models_t \alpha[A]$ and $M\models_f \alpha[A]$
- $v(\alpha) = u$ iff $M\not\models_t \alpha[A]$ and $M\not\models_f \alpha[A]$
- $v(\alpha) = f$ iff $M\not\models_t \alpha[A]$ and $M\models_f \alpha[A]$

Recalling that the truth values are ordered on the bilattice Λ, it is easy to verify that the evaluation function previously defined fulfills the following properties:

- $v(\alpha \wedge \beta) = \min_t(v(\alpha), v(\beta))$
- $v(\alpha \vee \beta) = \max_t(v(\alpha), v(\beta))$
- $v(\alpha \rightarrow \beta) = t$ **iff** $v(\alpha) \preceq_t v(\beta)$
- $v(\alpha \equiv \beta) = t$ **iff** $v(\alpha) = v(\beta)$

where subscript t indicates the "truth" dimension of the bilattice Λ.

From the above definitions, it is easy to see that when $M\models_t \alpha[A]$, formula α *can be "true" or "contradictory"* which in any case implies that there is presence of truth in α. Such a consequence relation introduces a kind of "ambiguity" since it does not allow to assign a truth value univocally (actually we need the "false consequence relation"). We can therefore define a *"strong consequence"* relation which may correspond to the case where formula α, in a variable assignment A, has exactly the truth value "true". This is typical of two-valued valuations.

Definition 3.2 (Strong Consequence.) *A formula α is true in a model M iff $M\models_t \alpha[A]$ and $M\not\models_f \alpha[A]$ for all variable assignments A and we write $M\equiv\alpha[A]$. A formula α is satisfiable iff α is true in a model M for some M. A set of formulas Γ is said to be a strong consequence or strongly entails a formula α (written $\Gamma\equiv\alpha$) when for all models M and variable assignments A, if $M\equiv\beta_i[A]$, for all $\beta_i \in \Gamma$, then $M\equiv\alpha[A]$.*

See Thomason and Horty (1988) and Fenstad et. al. (1987) for an account of related logics and their applications.

4   Some properties of DDT

4.1   Theorems of DDT

Proposition 4.1 *The following formulas hold in the DDT logic.*

1. $\not\sim\alpha \equiv (\alpha \vee \mathcal{K}) \rightarrow (\alpha \wedge \mathcal{K})$.

2. $\mathbf{T} \not\sim \alpha \equiv \mathbf{K}(\alpha \wedge \neg \alpha)$.

3. $\not\sim \alpha \wedge \not\sim \beta \rightarrow \not\sim (\alpha \wedge \beta)$.

4. $\alpha \wedge \not\sim \beta \rightarrow \not\sim (\alpha \wedge \beta)$.

5. $\not\sim \alpha \wedge \beta \rightarrow \not\sim (\alpha \wedge \beta)$.

6. $(\alpha \rightarrow \beta) \rightarrow ((\alpha \rightarrow (\beta \rightarrow \gamma)) \rightarrow (\alpha \rightarrow \gamma))$.

7. $\alpha \rightarrow ((\alpha \rightarrow \beta) \rightarrow \beta))$.

8. $\alpha \wedge \beta \rightarrow \beta$.

9. $\alpha \wedge \beta \rightarrow \alpha$.

10. $\alpha \rightarrow (\beta \rightarrow (\alpha \wedge \beta))$.

11. $\alpha \rightarrow \beta \vee \alpha$.

12. $\beta \rightarrow \beta \vee \alpha$.

13. $\alpha \vee \sim \alpha$.

14. $\neg \forall x \phi(x) \equiv \exists x \neg \phi(x)$.

15. $\neg \exists x \phi(x) \equiv \forall x \neg \phi(x)$.

16. $\not\sim \forall x \phi(x) \equiv \forall x \neg \not\sim \neg \phi(x)$.

17. $\not\sim \exists x \phi(x) \equiv \exists x \neg \not\sim \neg \phi(x)$.

Proof. Trivial from truth tables and semantics.

The following formulas do not hold in DDT.

1. $\alpha \wedge \neg \alpha \rightarrow \beta$.

2. $\beta \rightarrow \alpha \vee \neg \alpha$.

4.2 Paraconsistency

DDT is a paraconsistent logic. From the previous section we see that the *"reduction ad absurdum"* law does not hold in this logic and that this is sufficient to characterise it. However, two observations should be made.

1. The same law may be valid if we substitute the "strong negation" by the "complementation". The following therefore holds:
   - $\alpha \wedge \sim \alpha \rightarrow \beta$.
   - $\beta \rightarrow \alpha \vee \sim \alpha$.

2. In the two-valued fragment of the DDT logic, using formulas containing the strong unary operators $\mathbf{T}, \mathbf{K}, \mathbf{U}, \mathbf{F}, \triangle$, the law is again valid. The following therefore holds:
   - $\mathbf{T}\alpha \wedge \mathbf{T}\neg\alpha \rightarrow \beta$.
   - $\mathbf{T}\alpha \wedge \neg\mathbf{T}\alpha \rightarrow \beta$.
   - $\mathbf{T}(\alpha \wedge \neg\alpha) \rightarrow \beta$.
   - $\beta \rightarrow \mathbf{T}\alpha \vee \neg\mathbf{T}\alpha$.

Since the logic contains non-paraconsistent fragments we will call it a *"weakly paraconsistent logic"*.

5   Preference modeling applications

Preference modeling problems have been the original stimulus for the development of DDT logic. Preference modeling is a fundamental activity in decision aiding processes where a decision- maker's preferences have to be represented in order to identify a choice or a ranking among a set of alternatives (for more details see Roubens and Vincke, 1985 or Vincke, 1992). In such situations, uncertainty and/or ambiguity are very common, either because the available information is uncertain and/or imprecise or because the decision-maker is not sure and/or inconsistent. Decision aid calls for more or less *"action here and now"*. In other words decision-makers have to make a decision (whatever that means) in a precise time horizon, provided a specific amount of resources and information - knowledge is available. Therefore, no one cares what the definitely optimal choices may be, while a locally satisfactory solution is searched for (see Simon, 1979). Under such a perspective, uncertainty may not always be always reducible to more stable and sure situations, hence the necessity to study alternative formalisms for preference modelling (see also Kacprzyk and Roubens, 1988).

From a decision point of view, the distinction between uncertainty due to lack of information and uncertainty due to contradictory information is of capital importance since it generates different operational attitudes. In the first case, uncertainty may be reduced (if possible) to gathering more information (or the relevant information), while in the second case some conflicts, inconsistencies or contradictions have to be solved. When decision-makers are not sure it is always useful to know why.

The DDT language captures this kind of reasoning in a very clear and intuitive way. It has therefore been used as the basic formalism under which a non-conventional theory about preference modelling could be developed. In Tsoukiàs and Vincke (1995) a new preference structure, named **PC**, is introduced and axiomatised, while in Tsoukiàs and Vincke (1997) a semantical investigation, from a decision point of view, of the **PC** preference structure is conducted.

The basic ideas of this approach are the following. Preferential information is usually captured under a binary relation S ($s(x,y)$: x is at least as good as y). Given any two objects x and y, the relation S may be established between them evaluating both positive and negative reasons.

Conventional preference models use the well-known $\langle P, I, R \rangle$ preference structure where (see also Roubens and Vincke, 1985):
- P is strict preference; $\forall x, y \; p(x,y) \Leftrightarrow s(x,y) \wedge \neg s(y,x)$
- I is indifference; $\forall x, y \; i(x,y) \Leftrightarrow s(x,y) \wedge s(y,x)$

- R is incomparability; $\forall x, y\ p(x,y) \Leftrightarrow \neg s(x,y) \wedge \neg s(y,x)$

under the implicit hypothesis that negative reasons against S are the complement of positive reasons. However, if we keep the evaluation of positive and negative reasons between such crisp and sure situations independent, some hesitation may occur either due to lack of relevant information when an object x is compared to an object y or vice versa and/or contradictory information when an object x is compared to an object y or vice-versa (for examples see Tsoukiàs, 1994 or Tsoukiàs and Vincke, 1997). Therefore, between each pair of relations, two more relations can be introduced (one for each source of hesitation) and precisely:

- between P and I, the relations K and H; positive reasons are clear, but negative reasons are not;
- between P and R, the relations V and Q; negative reasons are clear, but positive reasons are not;
- between R and I, the relations U and J; symmetric hesitation

plus relation L between R and I (non symmetric hesitation).

These ten relations constitute the **PC** preference structure that can be defined using the DDT language as follows:

- $\forall x, y\ \mathbf{T}p(x,y) \Leftrightarrow \mathbf{T}s(x,y) \wedge \mathbf{F}s(y,x)$;
- $\forall x, y\ \mathbf{T}h(x,y) \Leftrightarrow \mathbf{T}s(x,y) \wedge \mathbf{K}s(y,x)$;
- $\forall x, y\ \mathbf{T}k(x,y) \Leftrightarrow \mathbf{T}s(x,y) \wedge \mathbf{U}s(y,x)$;
- $\forall x, y\ \mathbf{T}i(x,y) \Leftrightarrow \mathbf{T}s(x,y) \wedge \mathbf{T}s(y,x)$;
- $\forall x, y\ \mathbf{T}j(x,y) \Leftrightarrow \mathbf{K}s(x,y) \wedge \mathbf{K}s(y,x)$;
- $\forall x, y\ \mathbf{T}u(x,y) \Leftrightarrow \mathbf{U}s(x,y) \wedge \mathbf{U}s(y,x)$;
- $\forall x, y\ \mathbf{T}r(x,y) \Leftrightarrow \mathbf{F}s(x,y) \wedge \mathbf{F}s(y,x)$;
- $\forall x, y\ \mathbf{T}l(x,y) \Leftrightarrow \mathbf{K}s(x,y) \wedge \mathbf{U}s(y,x)$;
- $\forall x, y\ \mathbf{T}q(x,y) \Leftrightarrow \mathbf{F}s(x,y) \wedge \mathbf{K}s(y,x)$;
- $\forall x, y\ \mathbf{T}v(x,y) \Leftrightarrow \mathbf{F}s(x,y) \wedge \mathbf{U}s(y,x)$;

Tsoukiàs and Vincke (1995) proved that such a preference structure is a *maximal well-founded fundamental relational system of preferences* under the following three axioms.

A1 any preference structure on a set A should be a f.r.s.p. (fundamental relational system of preferences), i.e. should define a partition on $A \times A$ for any given A; in other words the preference relations included in the preference structure should be exhaustive for all possible situations and not redundant;

A2 the preference structure should follow the axiom of "independence from irrelevant alternatives"; roughly speaking if a specific ordered couple belongs (and in which way) to a specific relation, the evaluation should depend on information concerning only this ordered couple;

A3 the preference structure should be "well-founded" in the sense that any binary relation in it should be univocally defined by its properties.

As a consequence, some theoretical and operational problems in the field of multiple criteria decision aid can find elegant and definite solutions (see Tsoukiàs and Vincke, 1998, Tsoukiàs et al., 2002).

6 Rough sets semantics for DDT

6.1 About rough sets

Pawlak (1982) introduced rough sets theory as a new approach to the treatment of uncertain information, in particular the capability of distinguishing objects described in a more or less accurate way (see also Pawlak, 1991).

Following Pawlak, given $U \neq \emptyset$ a set or universe of objects and R (R being a family of equivalence relations R_i), $R \neq \emptyset$, then we define an *"indiscernibility"* relation $IND(R)$ as

$$IND(R) = \bigcap_U R \text{ intersection of all the equivalence relations belonging to} R$$

and $U/IND(R)$ or U/R as the family of all the equivalence classes of the equivalence relation $IND(R)$ on U.

We denote the couple (U, R) as a knowledge base $\mathcal{B}$. Usually, such a knowledge base takes the form of an "information table" defined by a set of elements x_j, a set of attributes R_i, the values $R_i(x_j)$, and eventually a decision attribute classifying elements in a given set X or in $\neg X$. The indiscernibility relation induces the definition of equivalence classes so that given an element x_j we denote as $[x_j]$ the equivalence class of this elements as follows:

$$[x_j] = \{y \in U : \forall i \; R_i(y) = R_i(x_j)\}$$

Given a $\mathcal{B} = (U, R)$ for each subset $X \subseteq U$ we associate two sets:

$$X_R = \bigcup_{x2\,X} \{[x] \in U/R : \; [x] \subseteq X\}$$

$$X^R = \bigcup_{x2\,X} \{[x] \in U/R : \; [x] \cap X \neq \emptyset\}$$

the R-lower and R-upper approximation of X by the description R of U, respectively. In other words, given a set of objects X, difficult to be describe, it is possible to approximate it using the description U/R with two sets:
- the lower approximation, which are the elements of U that are surely in X (following the classification U/R);
- the upper approximation which are the elements of U which possibly are in X (following the classification U/R);.

We finally define as $B(X) = X^R \setminus X_R$ the R-boundary region of X, that is the set of elements for which there is a doubt about their belonging to X.

6.2 Semantics for DDT

We will now try to show that in rough sets theory there is an implicit equivalence between rough sets semantics and DDT semantics. We will show that the true extension of a predicate S can be seen as its lower approximation within a knowledge base $\mathcal{B}$, that its false extension

can be seen as the lower approximation of the negation, its contradictory extension as the boundary region, the unknown extension being (almost) always empty.

The idea to associate a multiple-valued logic to rough sets is not new. Pagliani (1997) makes an extensive presentation on this issue and shows that approximation spaces and rough sets can be connected to three-valued Łukasiewicz algebras (see also Rine, 1991) and to chain based lattices. The results in this section confirm this approach, since only three extensions of a DDT predicate are non empty within an approximation space.

Given a structure M of similarity type ρ, let $S(x_1, \cdots x_m)$ be an atomic formula, such that $x_1 \cdots x_m$ are individual variables, $S \in \rho$ with arity m. Let also $\mathcal{B} = (U, R)$ be a knowledge base in M such that $U \in M$ and R is a set of equivalence relations on U such that $\bigcup_i \{U/R_i\} = U$. Let also DDT to be the language adopted. In the following, we omit variable assignments for clarity of the presentation. We give the following definition.

Definition 6.1

$$\mathcal{B} \models_t S(x) \; \textbf{iff} \; [x] \cap S \neq \emptyset$$

$$\mathcal{B} \models_f S(x) \; \textbf{iff} \; [x] \cap \neg S \neq \emptyset$$

where S and $\neg S$ are two complementary sets representing a classification of U.

Under such a definition the following proposition holds.

Proposition 6.1 *Given a formula $S(x)$, a knowledge base $\mathcal{B} = (U, R)$, S^t, S^k, S^u, S^f denoting the extensions of $\mathbf{T}S(x)$, $\mathbf{K}S(x)$, $\mathbf{U}S(x)$, $\mathbf{F}S(x)$, respectively. We have:*
1. $S^t = S_R$
2. $S^k = S^R \cap \neg S^R$
3. $S^u = \emptyset$
4. $S^f = \neg S_R$

Proof
1. $S^t = \{x : \mathcal{B} \models_t S(x) \text{ and } \mathcal{B} \not\models_f S(x)\}$
Therefore $S^t = \{x : [x] \cap S \neq \emptyset \text{ and } [x] \cap \neg S = \emptyset\} = S_R$.

2. $S^k = \{x : \mathcal{B} \models_t S(x) \text{ and } \mathcal{B} \models_f S(x)\}$
Therefore $S^t = \{x : [x] \cap S \neq \emptyset \text{ and } [x] \cap \neg S \neq \emptyset\} = S^R \cap \neg S^R$.

3. Obvious since it is not possible to have $[x] \cap S = \emptyset$ and $[x] \cap \neg S = \emptyset$ at the same time, $S, \neg S$ being a partition.

4. $S^f = \{x : \mathcal{B} \models_f S(x) \text{ and } \mathcal{B} \not\models_t S(x)\}$
Therefore $S^t = \{x : [x] \cap \neg S \neq \emptyset \text{ and } [x] \cap S = \emptyset\} = \neg S_R$.

∎

Hence the "true" extension of the predicate $S(x)$ under the knowledge base $\mathcal{B} = (U, R)$ is its lower approximation using the description R and the "false" extension is the lower approximation of the negation of $S(x)$. The "contradictory" extension is the intersection between the upper approximations of the predicate and its negation and the "unknown" extension is empty.

It is easy to see that it is sufficient to exchange the definitions of unknown and contradictory extensions and always obtain an empty contradictory extension, while the unknown extension will be the intersection of the two approximations. This is not surprising. The reader will find the algebraic reasons for that in Pagliani (1997).

6.3 Extensions

It is possible to extend rough sets theory in order to obtain a complete correspondence with DDT semantics, in the sense of obtaining that all four extensions are not empty (the unknown included)? The impediment is the definition of the indiscernibility relation and more precisely its property of reflexivity. Under such a property, any element of U belongs to at least one class, the one defined by itself and that for any classification. The unknown extension should be defined by $[x] \cap S = \emptyset$ and $[x] \cap \neg S = \emptyset$. Since $S, \neg S$ is a partition of U and $[x]$ are never empty this can never occur. If $[x]$ was defined by a non reflexive relation it could be the case.

Recently Słowiński and Vanderpooten (1997 and 2000) proposed to extend rough sets theory using similarity relations instead of equivalence relations. However, their minimal requirement is reflexivity of similarity. Stefanowski and Tsoukiàs (1999 and 2001) proposed to use similarity relations to handle incomplete information tables (elements of U may have unknown values for some attributes). They introduce the following definition:

$$\forall x, y \in U : \ H(x, y) \Leftrightarrow \forall i \text{ such that } R_i(x) \neq *, \ R_i(y) \neq * \ R_i(x) = R_i(y)$$

where $H(x, y)$ is a similarity relation and $*$ denotes an unknown value. In other terms, "x is similar to y" iff the known values of x are equal to the known values of y. It is not allowed to compare unknown values. Such a relation is transitive, but not symmetric (it is actually a partial order representing inclusion: an object x whose vector of values is $[*, 1, 2, 3]$ is similar to object y whose vector of values is $[0, 1, 2, 3]$, but the inverse is not true). Lower and upper approximations are thus defined on the basis of similarity classes instead of equivalence classes (for details see Słowiński and Vanderpooten 1997 and Stefanowski and Tsoukiàs, 1999, 2001).

In extreme situations, totally unknown objects (values in all attributes are unknown) are not similar to themselves by definition of the relation H. Formally speaking, relation H is not reflexive. The similarity class of such elements will be empty.

In such extreme situations we will have a non-empty "unknown extension" $S^u = \{x : \mathcal{B} \not\models_t S(x) \text{ and } \mathcal{B} \not\models_f S(x)\} = \{x : \ [x] \cap S = \emptyset \text{ and } [x] \cap \neg S = \emptyset\}$, since there is an x for which $[x]$ could be empty ($[x]$ denoting here a similarity class of x, $[x] = \{y : \ H(x, y)\}$). In reality the consequences are in reality marginal: some complementarity conditions no longer hold such as $S_R = (\neg S^R)^c$, since the complement of the upper approximation of the negation will also contain the unknown extension.

6.4 Rough Inference

One of the most common uses of rough sets is rules induction for classification purposes. Such rules however do not define a real inference scheme since they do not belong to a deductively closed system (the question was addressed by Orlowska and Pawlak, 1984 and Fariñas Del Cerro and Orlowska, 1985, Orlowska, 1988). One of the main advantages of the semantic equivalence between DDT and rough sets is that rules induced by rough approximations can be translated into logic formulas in DDT and therefore be incorporated in any deductive system based on such a language.

Given a knowledge base $\mathcal{B} = (U, R)$, under the usual form of an information table where R_i are the attributes and d is a decision attribute partitioning U into sets S_1, S_2, ..., S_m (a classification), the induced rules take the form:

$$\bigwedge_i R_i(v_l) \rightarrow d(S_j)$$

where v_l is a possible value for attribute R_i and $d(S_j)$ denotes the decision to classify the object described in the antecedent into class S_j.

Such rules are induced directly from $\mathcal{B}$ using elements of U which "support" them (elements which have the same description as the antecedent). Consider such an element: x
- if $[x] \subseteq S_j$ then the rule is considered certain;
- if $[x] \cap S_j \neq \emptyset$ the rule is considered possible.

Using the DDT^2 language we can introduce the following types of rules:
- $\bigwedge_i R_i(v_l) \rightarrow \mathbf{T}d(S_j)$ iff $[x] \subseteq S_j$
- $\bigwedge_i R_i(v_l) \rightarrow \triangle d(S_j)$ iff $[x] \cap S_j \neq \emptyset$
- $\bigwedge_i R_i(v_l) \rightarrow \mathbf{F}d(S_j)$ iff $[x] \subseteq \neg S_j$

Such a translation from DDT semantics is straightforward. The interesting part is that such rules can be combined by axiom schemata of the DDT language (which is a deductively closed system) of the type:
- $\forall x \, \forall S_j \ \ \mathbf{T}S_{i \neq j}(x) \rightarrow \mathbf{F}S_j(x)$;
- $\forall x \, \forall S_j \ \ \neg \triangle S_{i \neq j}(x) \rightarrow \bigvee_j \triangle S_j(x)$;

and/or specific inference rules associated to the knowledge base as axioms (originating from different sources) of the type (for instance):
- $\forall x \ \mathbf{F}S_2(x) \rightarrow \mathbf{F}S_3(x)$.

In this way it is possible to include induced rules in deductively closed systems thus enhancing their use (presently limited to classification schemes). Such a potentiality has been recently exploited by Greco et al., 1998, where a rough approximation of an outranking relation is exploited in order to obtain a final prescription in decision aiding situations.

7 Conclusions

A first order, four-valued logic is presented in the paper as an extension of Belnap's logic. The logic is equipped with a weak negation (preserving interlaced monotonicity on the bilattice of truth values) and a strong monotonic implication. A two-valued fragment, called DDT^2, is presented enabling the definition of strong two-valued sentences. Based on the idea that the

evaluation of the negative extension of a predicate is independent from the evaluation of the positive extension, a semantic is introduced: the complement of a predicate does not coincide with the extension of its negation and the universe of discourse may contain elements which do not belong to either of the two extensions. The resulting four possibilities correspond to the four truth values of the logic and define four possible extensions of any predicate. A double entailment relation is used to define such concepts and a strong entailment is introduced so as to have a correspondence with the evaluation function of the logic.

The application of the DDT language in preference modeling and decision aiding is outlined in the paper. Interested readers can refer to the quoted literature for more details. Moreover, the equivalence of rough sets semantics and DDT semantics is shown in the paper. Such an equivalence does not allow for the existence of all four the extensions of the DDT semantics, since the unknown extension is by definition empty. Some possible extensions are discussed, highlighting the fact that the reflexivity property of the relation used to classify objects impedes the existence of unknown extensions. Some extreme situations of non-reflexive relations are discussed, but are of limited interest. Finally, the equivalence of rough sets semantics and DDT semantics is exploited to obtain logical rules that can be included in deductively closed systems thus enhancing the potentialities of use of the induced classification rules.

A major open research question is how to exploit the paraconsistent nature of the DDT logic for defeasible reasoning to take into account new information which could be added in the knowledge base, such that a revision or an updating is required.

Acknowledgments

References

1. Belnap N.D., How a computer should think, *Proceedings of the Oxford International Symposium on Contemporary Aspects of Philosophy*, Oxford, England, 1976, 30 - 56.

2. Belnap N.D., A useful four-valued logic, in G. Epstein, J. Dumm, (eds.), *Modern uses of multiple valued logics*, D. Reidel, Dordrecht, 1977, 8 -37.

3. Bergstra J.A., Bethke I., Rodenburg P., A propositional logic with four values: true, false, divergent and meaningless", *Journal of Applied Non-Classical Logics*, **5**, 1995, 199 - 217.

4. da Costa N.C.A., Theory of inconsistent formal systems, *Notre Dame Journal of Formal Logic*, **15**, 1974, 497 - 510.

5. Doherty P., Driankov D., Tsoukiàs A., Partiality, Para-consistency and Preference Modelling, *IDA research report Lith-IDA-R-92-18*, Linköping University, 1992.

6. Dubarle D., Essai sur la généralisation naturelle de la logique usuelle, 1963, manuscript published posthumously in *Mathématique, Informatique, Sciences Humaines*, N° 107, 1989, 17 - 73.

7. Fages F., Ruet P., Combining explicit negation and negation by failure via Belnap's logic, *Theoretical Computer Science*, **171**, 1997, 61 - 75.

8. Fariñas del Cerro L., Orlowska E., DAL, a logic for data analysis, *Theoretical Computer Science*, **36**, 1985, 251 - 264.

9. Fenstad J.E., Halvorsen P-K., Langholm T., Van Benthem J., Situations, Language and Logic, in *Studies in Linguistics and Philosophy*, **34**, D. Reidel, Dordrecht, 1987.

10. Fitting M.C., Bilattices and the semantics of Logic Programming, *Journal of Logic Programming*, **11**, 1991, 91 - 116.

11. Font J.M., Moussavi M., Note on a six valued extension of three valued logics, *Journal of Applied Non-Classical Logics*, **3**, 1993, 173 - 187.

12. Ginsberg M., Multivalued logics: A uniform approach to reasoning in artificial intelligence, *Computational Intelligence*, **4**, 1988, 265 - 316.

13. Greco S., Matarazzo B., Słowiński R., Tsoukiàs A., Exploitation of a rough approximation of the outranking relation in Multi-Criteria Choice and Ranking, in T.J. Stewart and R.C. van der Honert (eds.), *Trends in Multi-Criteria Decision Making*, Springer Verlag, LNMES 465, Berlin, 1998, 45 - 60.

14. Kacprzyk J., Roubens M., (eds.), *Non Conventional Preference Relations in Decision Making*, Springer Verlag, LNMES n. 301, Berlin, 1988.

15. Kaluzhny Y., Muravitsky A.Y., A knowledge representation based on the Belnap's four valued logic, *Journal of Applied Non-Classical Logics*, **3**, 1993, 189 - 203.

16. Orlowska E., Logical Aspects of Learning Concepts, *International Journal of Approximate Reasoning*, **2**, 1988, 349 - 364.

17. Orlowska E., Pawlak Z., Logical Foundations of Knowledge Representation, *Institute of Computer Science, Polish Academy of Science Reports*, vol. 537,1984.

18. Pagliani P., Rough sets theory and logic algebraic structures, in E. Orlowska (ed.), *Incomplete Information: Rough Sets Analysis*, Verlag-Physica, Berlin, 1997, 109 - 190.

19. Pawlak Z., Rough Sets, *International Journal of Computer and Information Science*, **11**, 1982, 341 - 356.

20. Pawlak Z., *Rough Sets*, Kluwer Academic, Dordrecht, 1991.

21. Rescher N., Brandom R., *The logic of inconsistency*, Blackwell, Oxford, 1980.

22. Rine D.C., *Computer Science and Multiple Valued Logic: Theory and Applications*, North Holland, Amsterdam, 1991.

23. Roubens M., Vincke Ph., *Preference Modelling*, Springer Verlag, Berlin, 1985.

24. Ruet P., Complete set of connectives and complete sequent calculus for Belnap's Logic's, Document LIENS-96-28, cole Normale Suprieure, Paris, also in *Logic Colloquium 96*, San Sebastian, 1996.

25. Scott D., Continuous lattices, toposes, algebraic geometry and logic, in LNM 274, Springer, Berlin, 1972, 97 - 136.

26. Scott D., Some ordered sets in computer science, in I. Rival, (ed.), *Ordered Sets*, D. Reidel, Dordrecht, 1982, 677 - 718.

27. Simon H.A., Rational Decision Making in Business Organizations, *American Economic Review*, **69**, 1979, 493 - 513.

28. Słowiński R., Vanderpooten D., Similarity relation as a basis for rough approximations, in Wang P. (ed.) *Advances in Machine Intelligence and Soft Computing*, vol. IV., Duke University Press, 1997, 17 - 33.

29. Słowiński R., Vanderpooten D., A generalized definition of rough approximations based on similarity", *IEEE Transactions on Data and Knowledge Engineering*, **12**, 2000, 331 - 336.

30. Stefanowski J., Tsoukiàs A., On the extension of rough sets theory under incomplete information, in N. Zhong, A. Skowron, S. Ohsuga, (eds.), *New Directions in Rough Sets, Data Mining and Granular Soft Computing*, LNAI 1711, Springer Verlag, Berlin, 1999, 73 - 81.

31. Stefanowski J., Tsoukiàs A., Incomplete information tables and rough classification, *Computational Intelligence*, **17**, 2001, 454 - 466.

32. Thomason R., Horty J., Logics for inheritance theory, in M. Reinfrank, J. de Kleer, M.L. Ginsberg, E. Sandewall (eds.), *Non-Monotonic Reasoning*, vol. 346 of LNAI, Springer - Verlag, Berlin, 1987, 220 - 237.

33. Tsoukiàs A., A qualitative approach to face uncertainty in decision models, *Decision Support Systems*, **12**, 1994, 287 - 296.

34. Tsoukiàs A., Vincke Ph., A new axiomatic foundation of partial comparability, *Theory and Decision*, **39**, 1995, 79 - 114.

35. Tsoukiàs A., Vincke Ph., Extended preference structures in MCDA, in J. Clímaco (ed.), *Multicriteria Decision Analysis*, Springer Verlag, Berlin, 1997, 37 - 50.

36. Tsoukiàs A., Vincke Ph., Double Threshold Orders: a new axiomatization, *Journal of Multi- criteria Decision Analysis*, **7**, 1998, 285 - 301.

37. Tsoukiàs A., Perny P., Vincke Ph., From Concordance/Discordance to the Modelling of Positive and Negative Reasons in Decision Aiding, in D. Bouyssou, E. Jacquet-Lagrèze, P. Perny, R. Słowiński, D. Vanderpooten (eds.), *Aiding Decisions with Multiple Criteria: Essays in Honor of Bernard Roy*, Kluwer Academic, Dordrecht, 2002, 147 - 174.

38. Vincke Ph., *Multicriteria Decision Aid*, J. Wiley, New York, 1992.