Mohua Banerjee and Mihir K. Chakraborty
and Soma Dutta and Andrzej Skowron

Foundations of Rough Set Theory

December 1, 2025

Springer

Contents

Chapter 1
Introduction

1.1 Introduction

This is a treatise on rough set theory. In 1982, Polish computer scientist Zdzisław Pawlak proposed the theory through his publication entitled *Rough Sets* [**?**]. In it he declares: “The present approach may be considered as an alternative to fuzzy set theory and tolerance theory”. Numerous research articles, books and monographs have since been published, the research being widely extended in the areas of theoretical studies as well as applications.

The theory of fuzzy sets, founded in 1965 by Lotfi Zadeh (another computer scientist) [**?**], was on its steep slope of ascension when rough set theory came into being. We shall observe later that the word ‘alternative’ in the above quote is not to be interpreted in the sense of ‘contrary’, but in the sense of ‘other’. Both rough set and fuzzy set theories have overlapping areas in the realm of conceptual issues as well as fields of application. From some aspect one theory serves the purpose better, and from some other aspect, the other theory excels.

At the conceptual level, it is interesting to note that among several overlaps that both the theories have, lies one very important issue, viz. that of ‘vagueness’, its philosophy, mathematical representation and use. This fact, however, is not a mere coincidence.

Vagueness has been an outstanding topic of debate through ages. Great minds of antiquity, both of the East and the West, delved into the notion exhibited in various forms. The Buddhist teacher Nagasena asked the king Miranda: “Are you the same person, oh Lord, who used to play in the lap of your mother?” The allusion is towards the problem of *identity*, when change takes place continually or through indiscernibly close steps. The same issue is famous in Western literature as the Sorites Paradox. Typical instances of this paradox are a gradually balding head, or a heap of sand with gradually increasing or diminishing size. The age-old story of Theseus’ ship, would also fall under the same category. The ship was stranded in the sea-shore for a long time. As its parts decayed, they were replaced bit by bit. At each stage, it remained 'Theseus' ship'. But when it returned home, it was a totally new object. What is its identity?

Coming back to the Eastern context, we see that the Indian logicians talked about four corners indicating the states of belongingness of an object x to a concept P, viz.

(i) x is P
(ii) x is non-P
(iii) neither x is P nor x is non-P
(iv) x is P and x is non-P.

States (iii) and (iv) (particularly (iv)) are conceivable, only if the concept P is vague. This lattice, viz.

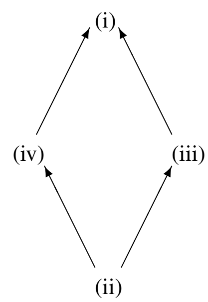


Fig. 1.1 Tetralemma

is called the *tetralemma* (Catuskoti [**?**]). This 4-valued lattice with the same interpretation of the corners, has played a very significant role after Belnap's publication [**?**]. Teacher Nagarjuna [**?**] added to the above four a fifth possibility: *none of* (i)-(iv). Apart from the Buddhists, another important school of ancient Indian thought is reflected in the Naiyayika tradition. An account of what could be the approaches of this school towards vague terms, is available in [**?**].

All these are instances of the fact that understanding vagueness was considered to be an important topic of knowledge or scholarly discourse. Despite the great tide of modern rationality all over the world, a perception that there exists an essential role of vagueness in the human knowledge system as well as life was never really obliterated from Eastern thought. ** Evidence?** **Western thought?** Ironically, a gift of modernity – computer science – spectacularly ushered the study of vagueness from a rather disrespectful margin, directly into centre-stage.

1.2 Vagueness

What is meant by vagueness after all? A closely related question is, which items could be ascribed with the adjective 'vague'? An object, a concept, or a linguistic entity? Can there be vague objects or is it only human conceptualization that admits vagueness? The debate on this continues. That there are linguistic entities which could be considered as vague is accepted, but what is not accepted generally is that these are the only possible candidates for vagueness – some believe that there can be vague objects, while others are of the opinion that no *actual* object is vague. However, whatever category or categories of entities admit vagueness, by a 'vague entity' is meant one that possesses an *unsharp boundary*. For example, a particular hill (an actual object indeed) is vague, since it is not clear exactly from which points on the earth's surface, it rises. Similar is the case of a particular patch of red that fades into yellow passing through orange. Again, 'hill' as a concept is vague, as it is dilemmatic to indicate when, at which stage, a heap turns into a hill. The linguistic entities such as 'tall', 'bald' or 'beautiful' are obviously vague, since there are objects to which the application of these words as adjectives is not determinate. Considering any physical object as a collection (of small parts, e.g. atoms), it is possible to treat vagueness of objects as vagueness of concepts, concepts being understood, extensionally, as a collection of smaller objects. This leads us to the age-old problem of identity. Is this the same ship of Theseus, although (all) its original parts are replaced? So, in our opinion, even though we admit vague objects, vagueness of concepts and that of languages do, in fact, matter. In our discussion, we shall be concerned only with these two. We would also like to draw attention to the concept-language interplay, that is, to the role of linguistic practices in the process of concept-formation within a community.

A concept is vague if it has unsharp boundary, or equivalently, there are borderline cases. Concepts 'hill', 'red' or 'tall' are all vague in this sense. There are several explanations for the existence of borderline instances put forward by researchers dealing with vagueness. Let a be a borderline case of the concept P, i.e. it is indeterminate whether P applies to a or not. Reasons for the indeterminacy may be the following [**?**].

(a) *a is either P or non-P, but it is not known which, or even not knowable*.
(b) *a is actually neither P nor non-P*.
(c) *a is partially P, and partially non-P*.
(d) *Depending on the context (perspective), a is sometimes P, and sometimes non-P*.

Fuzzy set theorists take the view-point (c), and assign an intermediate truth value to the sentence 'a is P'. Thus many-valuedness comes into the picture. It should be mentioned here that the value which a fuzzy set theorist would like to attach to the sentence 'a is P' is not absolute. The membership function for the concept P used for a purpose allows for small variations. But for a particular discourse, this function has to be fixed. In another context, it may be different. The language-users should somehow come to an agreement on this. Let us keep in mind the fact that in case of crisp statements (admitting values 'true' and 'false' only), the same procedure is adopted. "The geometry of space-time is non-Euclidean" is true because the scientist community agrees on this truth value. "Ravana is ten-headed" is true because the particular community agrees on the statement. To ascribe 'truth' or 'falsehood', each community follows some procedure – which may differ for different purposes. So, although an intermediate truth value is assigned to the sentence, it varies from context to context. In this sense, a fuzzy set theorist's account of vagueness corroborates (d) as well. This, of course, is the epistemic approach to truth. The ontic approach will eventually lead to the question if there are vague objects, which we shall by-pass in this treatise.

On the other hand, a rough set theorist's account clearly indicates the borderline instances of a concept in a given context. They are neither *P*, nor non-*P*, so that view-point (ii) is adopted. But a change in the context may bring such an instance within *P*, or push it to non-*P*. Moreover, through rough membership functions, grades of belongingness to the concept are assigned to these elements. Thus in the rough set approach to vagueness, the traces of view-point (c) are also observed. The detail of the above observations is given later in this chapter.

Remark 1.1. The resemblance of (b) and (c) with the tetralemma corners 'neither *P* nor non-*P*' and 'both *P* and non-*P*' (cf. Fig. 1.1) respectively, is evident. However, it is not the intention of this treatise to go any further into this comparison.

That vagueness in general is different from *probability* has somewhat been accepted nowadays after the long, fierce debates that took place during the years immediately following the advent of fuzzy set theory in 1965. So Pawlak did not have to fight that battle. Yet he he did point to an excellent distinctive criterion viz. "Vagueness is the property of sets... whereas uncertainty is the property of an element" [**?**]. Uncertainty leads to probabilistic studies. It is often said of course, that vagueness is uncertainty too, but not of the probabilistic kind. However, right from the beginning, Pawlak wanted to point at the distinction between rough set theory and fuzzy set theory. In the introduction to his short communication [**?**] he says "we compare this concept with that of the fuzzy set and we show that these two concepts are different." Different in what sense? Early Pawlak (during the 80's) was firm in his belief that rough set properly addresses vagueness, since it talks about 'boundaries' of a set and the property 'rough' is ascribed to a set. On the other hand, although the qualifier 'fuzzy' has been ascribed to sets too, in reality the theory deals with degree of membership of an object in a 'set' and hence deals with some kind of uncertainty of *belongingness* of objects. So according to the above quoted norm, fuzzy set theory does not address vagueness 'proper'. However, in later Pawlak, perhaps a change in opinion is observed as reflected in the following categorical remark: "Both fuzzy and rough set theory represent two different approaches to vagueness. Fuzzy set theory addresses gradualness of knowledge, expressed by the fuzzy membership – whereas rough set theory addresses granularity of knowledge expressed by indiscernibility relation" [**?**]. It may be mentioned that the relationship between the two theories, quite naturally, was a favourite topic of study in those turbulent decades.

We shall discuss the role of indiscernibility to some length in the foundations of fuzzy set theory as well as rough set theory and thus in vagueness.

In our opinion, there is an essential indiscernibility underlying all kinds of vagueness – an indiscernibility giving rise to granularity and another giving rise to gradualness. A justification for this claim may be called for here. "Perhaps everyone can agree, at least, that the presence of an actual or possible sorites series is sufficient for vagueness" (cf. Shapiro [**?**]). Starting from a positive instance of a vague concept *P*, say 'bald', through stages of indiscernible changes (e.g. addition of one hair) one arrives at an instance of 'non-bald'. This is because there is no stage where 'bald' changes to 'non-bald' by addition of one hair. Thus following Shapiro, a test for vagueness of a concept P may be the generation of a sorites series with respect to P. But 'indiscernibles' may be of various types, though one can probe into some core features of this elusive notion. These investigations also constituted a favourite topic in the 80's and 90's of the past century [**?**, **?**, **?**]. Discussions on indiscernibility go hand in hand with those on identity. The next section is devoted to this topic.

1.3 Indiscernibility and Identity

Objects a and b (or names a and b of objects) are said to be indiscernible relative to a specified collection $\mathscr{P}$ of properties (or names of properties in the collection $\mathscr{P}$), if and only if for any P in $\mathscr{P}$, a has P implies b has P and conversely (or the sentence $P(a) \leftrightarrow P(b)$ is true in some model of the language). The passage from the objects a, b to their names 'a', 'b' and from property P to its name 'P' is important. This is particularly so in the context of vagueness when objects and properties are not well-defined (or not well-formed / in the formative stage / in flux). However, to avoid complexity of presentation, we shall refrain from using names – we shall rather be concerned with a particular model of the language having 'actual' objects and properties. But it should be clear that we are in a discourse, talking about objects and properties in a language – not actually dealing with them.

An oft-encountered and important indiscernibility relation in the language of mathematical theories as well as in the everyday-language of life is that of 'real' (Waszkiewicz [**?**]) or 'pure' (Hodes [**?**]) identity, which is understood to hold between objects when they are 'one and the same'. As Leibniz would put it, a and b are identical when for all conceivable properties P, a has P if and only if b has P.

Through the years, there have been numerous debates over this very intuitive, but at the same time very elusive notion of 'identity'. Elusive, as many have attempted definitions of the notion (e.g. Quine [**?**], Brody [**?**], cf. Savellos [**?**]), which again many have refuted (e.g. Wiggins [**?**], Savellos [**?**]). Some conclude that it is a primitive, indefinable notion (Savellos [**?**]) and some that it is a purely metasystem relation and not situated in either the logic or the language of the structure concerned (Waszkiewicz [**?**]). That is to mention only a slice of the controversy surrounding the notion among mathematicians, logicians and philosophers. On the whole, 'identity' has come to occupy a prepotent position in mathematics, philosophy and in real life too.

Amidst all the polemics, however, a few have dared to air the view that identity is, after all, dispensable or, at least, possibly impracticable. Waszkiewicz and Conway among others, merit mention in this regard. In [**?**], Waszkiewicz says that the idea of 'real' identity (identity under all possible circumstances) is not very interesting; moreover, it is not clear whether such a notion can be applied to things with the nature more complicated than the nature of the mathematical abstracts. It is his opinion that in the case of mathematical structures, the intuitive notion 'the same' is formalized by the classical notion of isomorphism. He also points out that in our practice, we identify things with respect to *some* criteria only − contrary to the view of Leibniz (cf. Tarski [**?**]), Tarski [**?**] or Brody [**?**], who advocate the communion of *all conceivable* properties as a criterion for deciding identity. This last point of Waszkiewicz is strengthened by support from Quine [**?**] and Gorsky [**?**]. We shall be upholding this point of view, as will be clearer in Sections 1.3.1 and 1.4. To Conway [**?**] it appears that while building a mathematical theory, (i) objects may be created from earlier objects in any reasonably constructive fashion and (ii) equality among the created objects can be any desired equivalence relation.

Especially in real life, one encounters various approximations and has to act in vague situations. In such situations, generally speaking, identity turns ‘approximate’. Mathematical theories to model such situations lead to corresponding mathematical theories of ‘approximate identity’ or indiscernibility. Equivalence relations, isomorphisms, vague identity (Pultr [**?**]), fuzzy identity (Gottwald [**?**]), fuzzy equivalence (Zadeh [**?**]), indistinguishability (Trillas and Valverde [**?**]), rough equality (Pawlak [**?**]) etc. serve as examples of indiscernibility relations, the last few coming from the two previously mentioned major mathematical theories that deal with vagueness and/or partial (imprecise) knowledge, viz. fuzzy set theory and the theory of rough sets.

We shall see below, identity leads to the notion of indiscernibility and vice-versa. While indiscernibility has basically an epistemological content, identity is supposed to be an ontological notion. Yet, ‘understanding’ identity has remained perpetually elusive. The Leibnizian principle in this regard may be taken as the point of reference:

$x = y$ if and only if x has every property that y has and conversely.

In other words,

$x = y$ if and only if x has P implies y has P and conversely, for every property P. (LP)

There are two conditionals involved in (LP):

$x = y$ implies x has P if and only if y has P, for all P (1)

and

x has P if and only if y has P, for all P, implies $x = y$. (2)

(1) is called the principle of ‘indiscernibility of identicals’, whereas (2) is the principle of ‘identity of indiscernibles’. While there is general consensus over (1), the second principle has raised controversies. We shall, however, take (LP) as the initial definition of identity and incorporate certain essential modifications.

There are quite a few problems with the principle. Firstly, the word ‘every’ of ‘every property’ in the definiens is inconceivable. Leibniz wished to take all conceivable properties. Secondly, (LP) may characterize the identity of static objects only. And thirdly, understanding of properties and their identity comes prior to that of identity of objects – which one is more transparent is debatable [**?**]. As regards the third issue, viz. the object-property dichotomy, some discussion will be taken up later. Because of the first two problems, it seems reasonable to modify the Leibnizian principle as follows:

> $x = y$ if and only if x has P implies y has P and conversely, where P belongs to a 'specified' collection of properties. (MLP)

Extensionally, 'properties' can be considered as subsets of the universe of discourse within which identity should be understood, i.e. the elements of which are well-individuated. Without this, subsets cannot be defined at all. Let P be a property for objects in the universe X having an understood identity. Then

$x = y$ and x has P imply y has P. (S)

(S) gives one wing of (MLP) that may be called the modified principle of indiscernibility of identicals. In current literature, this is termed the 'substitutivity principle', and we shall refer to this principle and its variants or generalizations by this name as well.

We are inclined to claim that, to understand identity formally, it is only possible to reach upto some indiscernibility with certain suitable substitutivity conditions. Thus the slogan becomes

> Identity is indiscernibility, with appropriate substitutivity conditions.

The conditional (2) in (LP) gives the indiscernibility, and (1) substitutivity.

Now, what is the intuition behind the idea of indiscernibility (indistinguishability)? For some reason or other, objects a and b cannot be separated from each other. So, as a relation it may be reflexive and symmetric (i.e. tolerance) or reflexive, symmetric and transitive (equivalence). In both cases, clusters are formed with mutually indistinguishable objects. But, while in the first case two clusters may overlap, in the latter case they do not. In rough set literature, such clusters are called 'granules'. In this treatise, we shall be primarily concerned with the second kind of indiscernibility. There may be yet another notion of indiscernibility – a graded one. This approach is pursued in fuzzy set theory where it is presumed that objects a and b may be indiscernible to some extent or to a degree. In this approach, transitivity is not given up, but weakened as follows:

$$Ind(x,y) \ \& \ Ind(y,z) \leq Ind(x,z),$$

where $Ind(x,y)$ represents the indiscernibility degree between x and y, and & is an algebraic operation (a t-norm, perhaps) on a suitable truth set. This graded relation without being reduced to tolerance, relaxes the notion of hard transitivity and elegantly, takes care of the gradualness aspect by using an interactive conjunction operator as follows.

Let $x_1, x_2, x_3, x_4, \ldots$ be a sequence of objects such that $Ind(x_i, x_{i+1}) = 0.5$, for all i. Now let us take the product ($\times$), which is a t-norm, as the operator for &. Since $Ind(x_1, x_2)$ & $Ind(x_2, x_3) \leq Ind(x_1, x_3)$, we get $Ind(x_1, x_3) \geq 0.5 \times 0.5 = 0.25$.
If the least value 0.25 is taken then

$Ind(x_1, x_4)$ may be taken as $Ind(x_1, x_3)$ & $Ind(x_3, x_4) = 0.25 \times 0.5 = 0.125$.

Thus indiscernibility degree gradually diminishes. It means that the indiscernibility between x_1 and x_4 is less than that between x_1 and x_3, and this is further less than the indiscernibility between x_1 and x_2 – a feature quite intuitively acceptable. Symmetry is naturally expected of indiscernibility. In the fuzzy case it means that $Ind(x,y) = Ind(y,x)$. We shall discuss the reflexivity property later at an appropriate place.

Once the basic relation is defined, be it a tolerance, equivalence or fuzzy equivalence relation, the next step is to set the appropriate substitutivity conditions.

1.3.1 Some examples of indiscernibility relations

In this subsection, some typical indiscernibilities shall be illustrated. Among these, Example 1 carries the seed of rough set theory and hence is the most relevant one for this book. All the examples, except Example 2 however, are demonstrations of the point of view that indiscernibilities are equivalences (weak or strong) along with suitable substitutivity conditions. In this sense, an indiscernibility is more than equivalence and comparable to the notion of congruence in algebra. The more general notion of indiscernibility when tolerance is taken instead of equivalence, is considered in Example 2.

1.3.1.1 Example 1. Indiscernibility in rough set theory

A pair (U,R), where U is a non-empty set and R an equivalence relation on U, is called an *approximation space* [**?**]. Consider now an *information system* [**?**] with a set U of objects $x_1, \ldots, x_m$ and a set A of 'attributes' $a_1, \ldots, a_n$, such that every attribute assigns a 'value' from a set V to each object. This information may be represented by a table, where the entry corresponding to an object-attribute pair would be some value from V. Suppose $U := \{x_1, x_2, x_3\}$, $A := \{$colour(c), size (s), shape$(sh)\}$, and $V := \{$blue(bl), red(r), green(g), small(sm), big(b), round(ro), square$(sq)\}$. An information system on U, A, V would be given by a table, such as Table 1.1 below.

Table 1.1 An Information System

	c	s	sh
x_1	g	sm	ro
x_2	bl	b	sq
x_3	g	sm	ro
x_4	r	sm	sq

One may then define a binary relation Ind on the set U:

> $Ind(x,y)$, if and only if $a_i(x)$ is the same as $a_i(y)$ for all $a_i \in A$, $a_i(x)$ being the value of the object x with respect to the attribute a_i.

So the objects x and y are indiscernible with respect to the attributes given.
Ind is, clearly, an equivalence relation on U. So the pair (U,Ind) is an approximation space. The equivalence classes in the above example are: $\{x_1,x_3\},\{x_2\},\{x_4\}$.

The substitutivity principle that follows in this case is

> $Ind(x,y) \wedge x \in P$ imply $y \in P$, where P is any union of equivalence classes of Ind (such a P is termed *definable*). (1)

Let us give two basic definitions of rough set theory at this juncture.

Definition 1.1. Let P be any subset of U. The lower approximation $\underline{P}$ of P in the approximation space (U,Ind) is the set $\{x \in U : y \in P,$ whenever $Ind(x,y)\}$. The upper approximation $\overline{P}$ of P is the set $\{x \in X : y \in P,$ for some y with $Ind(x,y)\}$.
The set $\overline{P} \setminus \underline{P}$ is the boundary $Bn(P)$ of P in (U,Ind).
P is a Pawlak rough set [**?**] in (U,Ind), provided $Bn(P) \neq \emptyset$.

It follows that for any subset P of U, the following substitutivity principle holds.

$Ind(x,y) \Rightarrow (x \in \underline{P} \Leftrightarrow y \in \underline{P})$, and
$Ind(x,y) \Rightarrow (x \in \overline{P} \Leftrightarrow y \in \overline{P})$. (2)

One may observe that *only* subsets P of U which are definable, satisfy substitutivity principle (1). In particular, any subset of U consisting of all the elements that are assigned the same value for an attribute, would satisfy the condition. Besides, there is another substitutivity property of Ind, viz.

$Ind(x,y) \wedge x \in P \Rightarrow y \in \overline{P}$, for any $P \subseteq X$. (3)

Why should (1), (2) and (3) be called substitutivity conditions? The reason will be evident from the structure of the principle of 'indiscernibility of identicals', viz. (1) and (S) of Section 1.3. However in (3) above, we have implication only in one direction and after all

$$Ind(x,y) \wedge y \in \overline{P} \Rightarrow x \in P$$

does not hold generally. The information system, however, says the following by (2): if $Ind(x,y)$ and $y \in \overline{P}$ then $x \in \overline{P}$.

So we have

$$Ind(x,y) \Rightarrow ((x \in P \Rightarrow y \in \overline{P}) \wedge (y \in \overline{P} \Rightarrow x \in \overline{P})), \text{ for any } P \subseteq X.$$

Thus in order to generalize the principle of indiscernibility of identicals, it seems reasonable to drop the biconditional in favour of one-sided implication, i.e. the condition (S). Here is a reading of (3): if x is indiscernible with y and x has the property P then y possibly has the property P. For a weak identity, viz. Ind, such a claim is quite plausible. (3) is a weak substitutivity principle.

1.3.1.2 Example 2. Tolerance-based *Ind*

In real world situations, there may be gaps in information – the value of an object for some attribute may not be known (see e.g. [**?**, **?**]). In such cases, if the user decides to take a 'liberal' approach, an indiscernibility relation arises which is reflexive and symmetric (tolerance), but not transitive. An instance of this may be observed in Table 1.2. Consider the same sets of objects, attributes and attribute-values as in Example 1, viz. $U := \{x_1, x_2, x_3\}$, $A := \{\text{colour}(c), \text{shape}(s)\}$, and $V := \{\text{blue}(b), \text{green}(g), \text{round}(r), \text{square}(sq)\}$.

Table 1.2 An Incomplete Information System

	c	*s*	*sh*
x_1	*r*	*sm*	*ro*
x_2	–	*sm*	*ro*
x_3	*bl*	*sm*	*ro*
x_4	*g*	*b*	*sq*
x_5	*g*	*b*	–
x_6	*bl*	–	*sq*
x_7	–	*b*	*ro*

$Ind(x,y)$, if and only if for all attributes a, $a(x) = a(y)$, when these values are known. In other words, their known values are not different, i.e. they could be similar – e.g. $Ind(x_1,x_2)$, $Ind(x_2,x_3)$. But note that $Ind(x_1,x_3)$ does not hold.

With lower and upper approximations as in Definition 1.1, one gets the substitutivity conditions stated below.

$$Ind(x,y) \wedge (x \in \underline{P} \Rightarrow y \in P), \text{ and}$$
$$Ind(x,y) \wedge (x \in P \Rightarrow y \in \overline{P}).$$

Thus, combining,

$$Ind(x,y) \Rightarrow ((x \in \underline{P} \Rightarrow y \in P) \wedge (y \in P \Rightarrow x \in \overline{P})).$$

It may be remarked that various other kinds of substitutivity conditions arise in this case (cf. [**?**]). But in this book, we shall not deal with the tolerance-based approach in much detail.

1.3.1.3 Example 3. Formal set theory

It is revealing to observe that what is known to be equality or identity in set theory is actually equivalence with substitutivity, in other words, indiscernibility. We have taken the formal system NBG because the above fact is transparent in it.

In NBG, $\in$ ('being a member of') is the only kind of basic predication, and identity of two entities (sets) X and Y is presented first, by the abbreviation

(i) $X = Y$ stands for $\forall Z(Z \in X \leftrightarrow Z \in Y)$.
This is followed by the axiom

(ii) $X = Y \to \forall Z(X \in Z \leftrightarrow Y \in Z)$.

The two together mean that X and Y are taken to be identical, if and only if they include as members, exactly the same entities and are included in exactly the same entities.

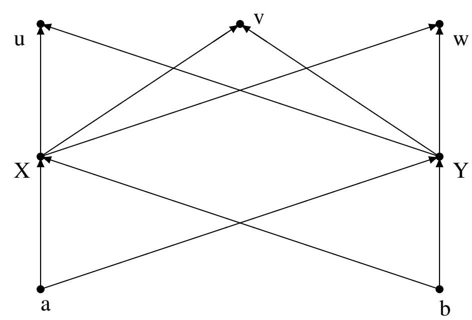


Fig. 1.2

In Fig. 1.2, the entities X and Y are to be taken as identical, because both include exactly the elements a, b, and are included exactly in u, v, w.

It turns out that '=' is ultimately characterized by the following conditions.

$X = X$
$X = Y \to Y = X$
$X = Y \to (Y = Z \to X = Z)$ and
$X = Y \to (\phi(X,Z) \leftrightarrow \phi(Y,Z))$,

where $\phi(X,Z)$ is any atomic formula, i.e. either $X \in Z$ or $Z \in X$, and $\phi(Y,Z)$ is obtained from $\phi(X,Z)$ by substituting Y for X. Thus '=' is an equivalence with a substitutivity condition and hence an indiscernibility.

1.3.1.4 Example 4. First order theory with equality

Among the usual symbols in the alphabet of the language of classical first-order logic, a particular 2-place predicate symbol E is called the *equality* predicate. The following axioms are taken for E.

1. $E(x,x)$
2. $E(x,y) \to E(y,x)$
3. $E(x,y) \to (E(y,z) \to E(x,z))$
4. $E(x,y) \to (P(x,x) \to P(x,y))$,

where $P(x,x)$ is any atomic formula with x as a free variable in it and $P(x,y)$ is obtained from $P(x,x)$ by replacing one or more x's in $P(x,x)$ by y's. All the axioms are not independent. 2 and 3 may be derived as theorems from the remaining two. 4 is the substitutivity principle. This principle extends to any formula by the following schemata

4′. $E(x,y) \to (\phi(x,x) \to \phi(x,y))$,

where $\phi(x,x)$ is any formula. 4′ is available as a theorem of the theory.

It can also be established that the following formulae are theorems.

5. $E(x,y) \to E(f(x_1,\ldots,x,\ldots,x_n), f(x_1,\ldots,y,\ldots,x_n))$,

for every function symbol f of the language with arity n, and

6. $E(x,y) \to (R(x_1,\ldots,x,\ldots,x_n) \leftrightarrow R(x_1,\ldots,y,\ldots,x_n))$,

for every predicate symbol R with arity n.
In fact, 1,5,6 constitute an alternative set of independent axioms for equality. Hence 5 and 6 together can also be called the substitutivity principles for the theory.

Through all these formalizations, what we have been able to achieve is only 'indistinguishability' relative to functions and relations that we are interested in. There seems to be no way to get rid of this crisis. As Wilfrid Hodges [**?**] puts it: "It would be pleasant if we could find a theory whose models are exactly the structures with standard identity. Alas, there is no such theory."

1.3.1.5 Example 5. Equality in many-valued logics

In many-valued logics, the truth value of a formula is not only true or false, it may also assume intermediate values. For equality statements, however, one is usually inclined not to accept such intermediate values, they must be all-or-none type, either two things are equal or they are not. This view leads to the notion of absolute equality [**?**]. There are non-absolutist views too in which equality is considered to be a graded relation analogous to the other predicates of the language [**?**]. Let us discuss the nature of the substitutivity principle when equality is of this latter kind.

Let the truth-value set be $\{1,2,\ldots,N\}$, where for some S, $1 \leq S < N$, the set $\{i : 1 \leq i \leq S\}$ is the set of designated values and the rest, i.e. $\{j : S < j \leq N\}$, is undesignated.

The language is the same as first-order language with the following changes: no function symbols are taken (for simplicity), all the connectives $\wedge, \vee, \to$ and $\neg$ are taken to be primitive (because of the lack of interdefinability), for each integer i, $1 \leq i \leq N$, there is a unary connective J_i. The language as well as the formal theory shall be denoted by *SML* following [**?**].

An interpretation I of the language associates to each n-ary predicate symbol P, a mapping $P_I : U^n \to \{1,2,\ldots,N\}$ and $E_I : U^2 \to \{1,2,\ldots,N\}$ to the special equality symbol E where U is a non-empty set, the domain of the interpretation.A model of the language is the pair (U,I). Relative to an interpretation, every formula receives a value in $\{1,2,\ldots,N\}$ in the usual manner.

A model of the theory is a set U and an interpretation I such that for every valuation v of the variables the axioms receive the designated value.

We now state the axioms related with the equality predicate E.

1. $J_1E(x,x)$
2. $J_kE(x,y) \to J_kE(y,x),\ k \in \{1,2,\ldots,N\}$
3. For each n-ary predicate P,

$$J_{k'}E(x,y) \wedge J_kP(x_1,\ldots,x,\ldots,x_n) \to \sum_{p=mnf(k',k)}^{mxf(k',k)} J_pP(x_1,\ldots,y,\ldots,x_n),$$

where $mnf(k',k) := max\{1, k-(k'-1)\},\ mxf(k',k) := min\{N, k+(k'-1)\}$, and

$$\sum_{i=m}^{n} \phi_i := \phi_m \vee \ldots \vee \phi_n,\ m \leq n.$$

1 and 2 are many-valued versions of reflexivity and symmetry, and 3 is the substitutivity principle. The following form of transitivity follows from the above axioms.

4.
$$J_{k'}E(x,y) \wedge J_kE(y,z) \to \sum_{p=mnf(k',k)}^{mxf(k',k)} J_pE(x,z).$$

To explain the intended meanings of the above formulae, it should be noticed that the J- operators are like second-level operators; if the truth-value of a formula ϕ is k then that of $J_k\phi$ is 1 (*True*), otherwise N (*False*). So, if the value of $E(x,y)$ is k for some assignment of x and y, for the same assignment, the value of $J_kE(x,y)$ is 1. This means: "that x and y are equal to an extent k" is true to the extent 1.

Now the intuitive interpretations of the axioms may be given.
Axiom 1 fulfils the demand that everything should be equal to itself to the highest degree, i.e. 1.
Axiom 2 satisfies the demand of symmetry of the function E_I, viz. if x and y are equal to the extent k then y and x are also equal to the same extent.
Axiom 3 captures the intention that if x and y are equal to the extent k' and x has P-hood to the extent k then y has P-hood to an extent ranging between $mnf(k',k)$ to $mxf(k',k)$. Thus instead of a precise information about the degree of P-hood of y, we now obtain a range of possible degrees.
Axiom 4, which is deducible from axioms 2 and 3, states that if x and y are equal to the extent k' and y and z are equal to the extent k then x and z are equal to an extent p such that $mnf(k',k) \leq p \leq mxf(k',k)$.

It should be stated that the principle of substitution cannot be extended to an arbitrary formula $\phi(x_1,\ldots,x,\ldots,x_n)$, as is in the case of two-valued classical logic. But "as long as two items are similar to degree 1, a corresponding theorem for formulae with higher complexity can be proved. But when the degree of similarity is not 1then for complex expressions we must calculate the deviation in truth-values occasioned by substitution, from the deviation in truth-values of the atomic constituents" [**?**]. All these seem quite reasonable.

1.3.1.6 Example 6. Indiscernibility in fuzzy logics

Since there are many kinds of fuzzy logics, we shall restrict ourselves to only two of them.

(A) Many-valued first-order fuzzy logic (cf. Bolc and Borowik [**?**], Novak [**?**])

In this fuzzy logic, the truth-value of a formula, like in many-valued logic, is not only true and false, but (unlike many-valued logic) comes from a more generalized structure, viz. a complete residuated lattice $L := (L, \wedge, \vee, ^*, \Rightarrow, 0, 1)$. Here

$(L, \wedge, \vee)$ is a complete lattice, 0,1 being the least and greatest elements,
$(L, ^*, 1)$ is a commutative monoid, * being called *multiplication*,
* is isotone in both the arguments, i.e. $\alpha \leq \beta$ and $\alpha' \leq \beta'$ imply $\alpha^*\alpha' \leq \beta^*\beta'$,
$\Rightarrow$ is antitone in the first and isotone in the second argument, i.e. $\alpha \leq \beta$ implies that $(\alpha \Rightarrow \gamma) \geq (\beta \Rightarrow \gamma)$ and $(\gamma \Rightarrow \alpha) \leq (\gamma \Rightarrow \beta)$, for any γ,
the adjunction property holds, viz. for every α, β, γ in L, $\alpha^*\beta \leq \gamma$ if and only if $\alpha \leq \beta \Rightarrow \gamma$.

A fuzzy binary relation $\varepsilon : U^2 \to L$ is said to be an equivalence (a fuzzy equivalence) on U, if and on ly if the following versions of fuzzy reflexivity (e_1), fuzzy symmetry (e_2) and fuzzy transitivity (e_3) hold, for all a, b, c in U.

(e_1) $\quad \varepsilon(a,a) = 1$
(e_2) $\quad \varepsilon(a,b) = \varepsilon(b,a)$
(e_3) $\quad \varepsilon(a,b)^*\varepsilon(b,c) \leq \varepsilon(a,c)$.

Let us assume, again for simplicity, that no function symbol is present in the language. As in many-valued logic, an interpretation I associates to each n-ary predicate symbol P, a fuzzy set P_I which is a mapping from U^n to L, where U is the domain of interpretation. Unlike many-valued logic, I assigns to every constant, a fuzzy point. (A fuzzy point μ_a for $a \in U$, is a mapping from U to L such that $\mu_a(a) \neq 0$ and $\mu_a(x) = 0$, for all $x \in U,\ x \neq a$.)

Usually * is preserved for computing 'conjunction' and $\Rightarrow$ for 'implication' in the fuzzy logic of our discussion. The 'negation' is computed by taking $\neg\phi$ as $\phi \to \bot$, $\bot$ being a symbol of the language standing for 'absurdity', the name for 0 of the truth set L. In fact, every truth-value has a name in it.

The notions of syntactic and semantic consequences are also different from those in many-valued case. Every formula has a degree α of theoremhood and in every model a degree of validity. The first is denoted by $\vdash_\alpha \phi$ and the second by $M \models_\beta \phi$.** (For detail see Novak [**?**].)

In order that the fuzzy logic is sound, it is to be observed that the following validity theorem holds.

Proposition 1.1. *For any formula ϕ, if $\vdash_\alpha \phi$ then for every model M, $M \models_\beta \phi$ such that $\alpha \leq \beta$.*

That is, for being sound, the degree of theoremhood in the system of a formula ϕ must not exceed the value that ϕ receives in any model relative to any valuation.

If for some formula ϕ, $M \models_1 \phi$ then ϕ is called a tautology.

The axioms for the equality predicate E may be taken exactly as in ordinary first-order logic, viz. E_1, E_2, E_3 and the substitutivity principle 4 given in Example 4. It is intended that in every model these should be tautologies in the above sense. So we obtain the following proposition.

Proposition 1.2. *The fuzzy relation ε, which is the interpretation of E in any model M with domain U, satisfies the conditions*

$$\varepsilon(x,x) = 1$$
$$\varepsilon(x,y) = \varepsilon(y,x)$$
$$\varepsilon(x,y)^* \varepsilon(y,z) \leq \varepsilon(x,z)$$
$$\varepsilon(x,y)^* (ext_M \phi)(x) \leq (ext_M \phi)(y),$$

where $(ext_M \phi)(x)$, the extension of ϕ, here, is a fuzzy subset of U.
The proof is immediate from the following properties of the truth set L:

$\alpha \leq \beta$, if and only if $\alpha \Rightarrow \beta = 1$, and
$\alpha \Rightarrow (\alpha \Rightarrow \gamma) = (\alpha^* \beta) \Rightarrow \gamma.$

The first three conditions of Proposition 1.2 show that ε is a fuzzy equivalence and the fourth one, when interpreted, gives the following substitutivity principle:

> The degree of ϕ-hood of 'y' is at least as large as the degree of ϕ-hood of 'x' multiplied by the degree of equality of 'x' and 'y'.

(B) Fuzzy indiscernibility in terms of vague properties (cf. Chakraborty and Banerjee [**?**])

This is, in fact, a generalization of Example 1, where the attributes are fuzzy – for instance, 'colour-red'. Let U and L be as in (A) above. For an arbitrary collection $\{a_i\}_{i \in I}$ of elements of L, I being a non-empty index set, we define $\bigwedge_* \{a_i\}$ as

$$inf\{a \in L : a = a_{i_1} * \ldots * a_{i_k},\ \{a_{i_1}, \ldots, a_{i_k}\} \text{ is any finite subfamily of } \{a_i\}_{i \in I}\}.$$

Let $A_i : U \to L,\ i \in I$, be a collection of fuzzy subsets of U representing vague attributes. Let $\varepsilon : U \times U \to L$ be defined by

$$\varepsilon(a,b) := \bigwedge_* \{A_i(a) \Leftrightarrow A_i(b)\},\ \ a,b \in U,$$

where $\alpha \Leftrightarrow \beta$ is $(\alpha \Rightarrow \beta)^*(\beta \Rightarrow \alpha)$, for $\alpha, \beta \in L$.
In the crisp case, the meaning of the right-hand expression is that all the attributes that we are interested in, if possessed by 'a' is also possessed by 'b' and conversely. Thus, it is the fuzzy version of the Leibnizian principle. Each $A_i(a) \Leftrightarrow A_i(b)$ is the measure of the degree of indiscernibility of 'a' and 'b' relative to the vague attribute A_i and the right-hand side as a whole is such a measure with respect to the totality of attributes $A_i,\ i \in I$.

We now have the following proposition [**?**].

Proposition 1.3. *ε defined as above satisfies the conditions $(e_1), (e_2), (e_3)$ and the substitutivity principle given by*

$$\varepsilon(a,b)^* A_i(a) \leq A_i(b),$$

for all $a, b \in U$ and A_i belonging to the collection generating ε.

Let us give an example, where $L := [0,1]$ and * is the Łukasiewicz conjunction operator, viz. $\alpha^*\beta := max(\alpha+\beta-1,0)$, for $\alpha,\beta \in [0,1]$. Suppose $U := \{x_1, x_2, x_3\}$, and consider the fuzzy attributes 'colour-red' and 'colour-yellow', represented by $A_1, A_2 : U \to [0,1]$ given respectively as:
$A_1(x_1) = .3$, $A_1(x_2) = .5$ and $A_1(x_3) = 0$, and
$A_2(x_1) = 0$, $A_2(x_2) = .1$ and $A_2(x_3) = 0$.

Then the indiscernibility ε is given by Table 1.3.

Table 1.3

ε	x_1	x_2	x_3
x_1	1	.7	.7
x_2	.7	1	.4
x_3	.7	.4	1

Substitutivity principle in the fuzzy context is also known as 'saturatedness'. For more details about this notion, one is referred to Jacas [**?**], Pultr [**?**], Valverde [**?**]. Discussions on fuzzy identity from a category-theoretic standpoint may be found in Wyler [**?**]. In this book, a related discussion shall be taken up in a later chapter.

1.4 Concept carrying its identity criterion

In some of the examples, it is noticed that an indiscernibility (identity) is created (generated) in terms of some basic predications and this is in conformity with the Leibnizian principle of identity of indiscernibles. The other predications have to be so formed that they would satisfy the substitutivity condition with respect to the generated indiscernibility, i.e. the properties describable by the more complex formulae would comply with the other half of Leibnizian principle, viz. the indiscernibility of identicals. In this section, we shall discuss this duality from a more intuitive angle and at the end, argue for the basic claim that at the root of any vague concept lies an indiscernibility, and any vague concept gives rise to an indiscernibility. Since our focus is on rough set theory and we would like to make some comparisons with fuzzy set theory, indiscernibilities involved in these two theories only shall be taken up. It is said that rough set theory deals with granularity, while fuzzy set theory deals with gradualness. So vague concepts are dealt with in two different ways. Two different kinds of indiscernibilities are created – one giving rise to granularity, and the other gradualness.

We refer to Examples 1 and 4 of the previous section. Let there be only three basic (monadic) predicates P,Q,R in the first-order language. Then the indiscernibility E is defined as

$$E(x,y) := (Px \leftrightarrow Py) \wedge (Qx \leftrightarrow Qy) \wedge (Rx \leftrightarrow Ry). \qquad (^*)$$

As claimed in Example 4, E satisfies the substitutivity condition

$$E(x,y) \wedge \phi(x)) \rightarrow \phi(y),$$

for any formula ϕ.

Thus all the properties in this context are given by the formulae $\phi(x)$, with x as free variable. In the interpretation, the referent of $\phi(x)$ is a subset of the domain U. So these subsets of U may be called 'properties'. $\phi(x)$ will depend ultimately on the basic predicates P,Q,R and various logical combinations of the atomic formulae constituted out of them. From the viewpoint of interpretation, Fig. 1.3 below shows the extensions of predicates P,Q and R in a domain U of 15 elements $x_1,\ldots,x_{15}$.

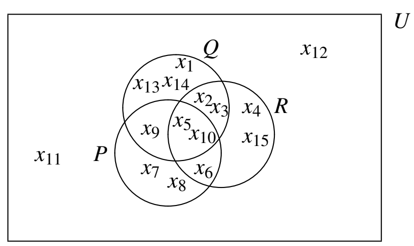


Fig. 1.3

The indiscernibility created in U by definition $(^*)$, is given by the partition on U generated by P,Q and R. In other words,

$$E(x,y) \text{ holds if and only if } (x \in P \leftrightarrow y \in P) \wedge (x \in Q \leftrightarrow y \in Q) \wedge (x \in R \leftrightarrow y \in R).$$

Now E satisfies the substitutivity condition for some subsets of U given by the formulae $\phi(x)$. One can immediately see that a subset S of U satisfies the condition relative to E, if and only if S is the union of some of the equivalence classes.

We could also demonstrate the indiscernibility E by constructing the approximation space given by Table 1.4.

Y stands for 'yes, belongs to' and N for 'no, does not belong to'. The properties, i.e. sets satisfying the substitutivity conditions are precisely the definable sets (cf. (1), Example 1). Thus concepts P,Q,R create the indiscernibility E which is an equivalence relation and satisfies the substitutivity condition with respect to, exactly, the definable subsets of U. Definable sets are, in this sense, crisp relative to the indiscernibility E.

A concept represented by any other subset of U would be considered 'vague' relative to E. This vagueness may be ascribed to the concept by all the three criteria (b), (c) and (d) mentioned in Section 1.1. Let us elaborate on this.

Table 1.4

	A	B	C
x_1	Y	N	N
x_2	Y	Y	N
x_3	Y	Y	N
x_4	N	Y	N
x_5	Y	Y	Y
x_6	N	Y	Y
x_7	N	N	Y
x_8	N	N	Y
x_9	Y	N	Y
x_{10}	Y	Y	Y
x_{11}	N	N	N
x_{12}	N	N	N
x_{13}	Y	N	N
x_{14}	Y	N	N
x_{15}	N	Y	N

(b): Any subset P of X endowed with the indiscernibility E is 'approximated' by the two definable subsets: the lower and upper approximations $\underline{P}$ and $\overline{P}$ (cf. Definition 1.1) respectively of the set (and hence the concept) P. Any object belonging to the *boundary region*, viz. $\overline{P} \setminus \underline{P}$ is a borderline case of the concept, since not all objects indiscernible with the given object would belong to P. Thus the substitutivity principle is violated. If this principle is adhered to, it would be only reasonable to consider an object to belong to the concept P if it belongs to the lower approximation, and to the concept non-P if it belongs to the complement of the upper approximation. Objects belonging to $\overline{P} \setminus \underline{P}$ are neither P nor non-P in this sense.

(c): There is an approach to rough set theory through rough membership functions [**?**]. Let $[x]$ denote the equivalence class of the object x of U with respect to a given indiscernibility. One defines the rough membership degree $\mu_P(x)$ of an object x relative to a given subset P of U by means of the number of objects in P that are indiscernible with x. Formally, $\mu^P(x) := \frac{|[x] \cap P|}{|[x]|}$. Thus x receives the membership value 1 when it belongs to $\underline{P}$, 0 when in $\overline{P}^c$ and values between 0 and 1 when it is in $\overline{P} \setminus \underline{P}$. So an element in $\overline{P} \setminus \underline{P}$ may be considered to belong partially to the concept P, and partially to non-P. This is similar to the fuzzy set approach, but there are many differences too – these would be taken up in a later chapter.

(d): Objects belonging to the region $\overline{P} \setminus \underline{P}$ relative to E may be considered as borderline cases by this criterion too. If the number of basic concepts defining E is increased, i.e. the context is changed, a new indiscernibility E_1 (say) is obtained. The equivalence classes and hence the partition generated in U are finer. So, it may be the case that the present borderline case comes within to the lower approximation of P with respect to E_1, or is pushed to the complement of the upper approximation. In the first case, it would be considered to belong to the concept P, and in the second, to belong to non-P. Thus when the context is finer, some objects that could not be brought under the concept earlier, come within it. On the other hand, when the context is coarser, some objects that were non-P earlier, may not be given that status any more.

The context, i.e. the partition, may change quite irregularly. In such a scenario, the status of an object belonging to $\overline{P} \setminus \underline{P}$ relative to E may fall under P or non-P quite erratically. But this fact would remain true also for objects within $\underline{P}$ or outside $\overline{P}$. So it is an interesting project to study change in the context when some of the equivalence classes remain undisturbed. Work in this direction may be found, for example, in [**?**, **?**].

It is also possible to interpret a concept by a variable extension. Given an indiscernibility E, a concept may be represented by two extensions P and Q such that $\underline{P} = \underline{Q}$ and $\overline{P} = \overline{Q}$. P, Q are called *roughly equal* [**?**]. An object of P belonging to $\overline{P} \setminus \underline{P}$ may not belong to Q and vice-versa. Thus by this interpretation, the indiscernibility fixes, for a concept, some objects to lie definitely within the concept and some objects to be definitely outside it, and others are sometimes within and sometimes outside the concept.

In the whole discussion, to impart vagueness by criterion (iv) of Section 1.1 to a concept represented in rough set theory, one aspect remains common: the substitutivity principle does not hold for all objects of the domain, and the borderline instances are characterized by this property. So we arrive at an important question: could a concept be taken as vague, if there are objects that do not satisfy the substitutivity principle in the classical (Leibnizian) sense? Or, should we look for some more general version of the principle that would give an apt account of the feature arising from rough set theory?

But before taking up this issue, let us pass on to the indiscernibility embedded in a vague concept represented by fuzzy set theory.

As mentioned before, an indiscernibility (cf. Example 6) is, first of all, an equivalence. In this case, it should be a fuzzy equivalence, given by

$$\varepsilon(x,y) \leq \varepsilon(x,x)$$
$$\varepsilon(x,y) = \varepsilon(y,x)$$
$$\varepsilon(x,y) * \varepsilon(y,z) \leq \varepsilon(x,z)$$

If a fuzzy subset A of X is defined by

$$A(x) := \varepsilon(x,x),\ x \in U,$$

then it can be easily checked that

$$\varepsilon(x,y)^* A(x) \leq A(y). \qquad (^*)$$

This last inequality is the fuzzy version of the substitutivity principle. Thus a concept (vague) represented by the fuzzy subset A is created out of the equivalence E, which turns out to be an indiscernibility relative to some fuzzy subsets or vague concepts. Since $A(x) := \varepsilon(x,x)$, the degree of existence of an object x in the concept A is the degree to which x is indiscernible with itself. It should be mentioned that A is not the only fuzzy set satisfying substitutivity relative to E. In fact, it is possible to establish a necessary and sufficient condition for a fuzzy set to satisfy this criterion [**?**].

Let us now return to the issue of the relationship between the existence of an object in a concept, and the indiscernibility generated by the concept. The philosophy regarding the sameness of the degree of existence of an object x in a concept created by some indiscernibility relation ε and the object's indiscernibility degree with itself $\varepsilon(x,x)$ is ratified by several authors (cf. [**?**]). The following construction from [**?**] in the context of rough sets ratifies it further. Besides, this construction establishes a connection between the rough set approach and fuzzy set approach to vague concepts as well.

Let U be a domain and $L(4)$ be the ordered set $\{0 \leq 1 \leq 2 \leq 3\}$, which is a complete distributive lattice (or a complete Heyting algebra).
Let $Ind^* : U \times U \to L(4)$ be an indiscernibility relation that satisfies conditions
$H_1 :\ Ind^*(x,y) = Ind^*(y,x)$ (Symmetry)
$H_2 :\ Ind^*(x,y) \wedge Ind^*(y,z) \leq Ind^*(x,z)$ (Transitivity)
and the following roughness conditions
$R_1 :\ 1 \leq Ind^*(x,x)$ for $x \in U$
$R_2 :$ if $2 \leq Ind^*(x,y)$, then $x = y$
$R_3 :$ if $Ind^*(x,y) = 1$, then $Ind^*(x,x) = Ind^*(y,y)$
$R_4 :$ if $Ind^*(x,x) = 2$, then there exists y such that $Ind^*(x,y) = 1$

The significance of the roughness conditions shall be clear from Proposition 2 below. The following two propositions establish that (U, Ind^*) is a representation of any Pawlak rough set (defined in Example 1, Section 1.3.1) in U.

Proposition 1.4. *Let (U, Ind^*) be given. Then the relation R defined by xRy if and only if $Ind^*(x,y) \geq 1$ is an equivalence relation, and the pair (I,B) defined by $I = \{x : Ind^*(x,x) = 3\}$, $B = \{x : Ind^*(x,x) = 2\}$ constitute the lower approximation and boundary of a Pawlak rough set in (U,R).*

**Proof

Proposition 1.5. *Let (U,R) be an approximation space in which (I,B), the lower approximation and boundary pair determines a Pawlak rough set. Consider the mapping $Ind^* : U \times U \to L(4)$ given by*

$$Ind^*(x,x) = \begin{cases} 3 & \text{if } x \in I \\ 2 & \text{if } x \in B \\ 1 & \text{if } x \in U \setminus (I \cup B) \end{cases}$$

and

$$Ind^*(x,y) = \begin{cases} 1 & \text{if } x \neq y,\ xRy \text{ holds} \\ 0 & \text{if } x \neq y,\ xRy \text{ does not hold.} \end{cases}$$

Then Ind^* satisfies the conditions $H_1, H_2, R_1, R_2, R_3, R_4$.

The following important feature is also observed in the two constructions by Propositions 1 and 2.
$(U, Ind^*) \xrightarrow{Prop1} (U,R,I,B) \xrightarrow{Prop2} (U, Ind^*) \xrightarrow{Prop1} (U,R,I,B)$.

Thus an indiscernibility satisfying the roughness conditions gives rise to a ‘rough set’ in an approximation space and vice versa. A rough set is not defined as yet. But we shall notice in the next chapter that the 4-tuple (U,R,I,B) is, in fact, one among several definitions of the concept.

From the definition of Ind^*, it follows that

$Ind^*(x,y) \leq Ind^*(x,x)$, and
$Ind^*(x,x) \wedge Ind^*(x,y) \leq Ind^*(y,y)$.

Now if a fuzzy subset A of U is defined by $A(x) := Ind^*(x,x)$, then we get that

> the degree of belongingness $A(x)$ of x in A is the same as its degree of indiscernibility with itself $Ind^*(x,x)$, and **if the degree of belongingness of x to the concept and the degree of indiscernibility of x with y then the degree of belongingness of y to the concept.

It is also significant to notice that the fuzzy set theoretic operators ‘max’ and ‘min’ are now applicable to obtain the union and intersection.

A summary of what has been said so far is the following:

- The underlying indiscernibility relation for any vague concept in U is a relation Ind^* satisfying the conditions H_1,H_2,R_1,R_2,R_3,R_4.
- Such a relation, which is a particular kind of fuzzy equivalence relation, determines uniquely a rough set in the approximation space (U,R), where R is the underlying indiscernibility and conditions R_1,R_2,R_3,R_4 determine the lower approximation and boundary of the rough set.
- Conversely, any rough set in (U,R) given by the lower approximation and boundary can be generated by an indiscernibility relation satisfying H_1,H_2,R_1,R_2,R_3,R_4.
- This representation is one-to-one.

Thus fuzzy sets and rough sets come closer. That means from a deeper angle of observation a fuzzy concept and a rough concept are similar in that each would be generated out of an indiscernibility with appropriate substitutivity conditions. Thus a bridge is built between these two important theories of vagueness.

1.5 Vagueness in Physics

1.6 Vagueness in Computer Science

References

1. Ballmer, T. T., Pinkal, M.: (Eds.) Approaching Vagueness. North-Holland, Amsterdam (1983)
2. Banerjee, M.: A Categorial Approach to the Algebra and Logic of the Indiscernible, Ph.D. Thesis, University of Calcutta, India (1995)

3. Banerjee, M., Khan, M. A.: Rough set theory: a temporal logic view. In: Chakraborty, M. K. et al. (eds.) Logic, Navya-Nyāya & Applications, Homage to Bimal Krishna Matilal, Studies in Logic, Vol. 15, pp. 1–20. College Publications, London (2008)
4. Belnap, N.D.Jr.: A Useful Four-Valued Logic. In: Dunn, J. M., Epstein, G. (eds.) Modern Uses of Multiple-Valued Logic, pp. 8–37. D. Reidel Pub. Co., Dordrecht (1977)
5. Bolc, L., Borowik, P.: Many-valued Logics I, Theoretical Foundations. Springer-Verlag (1992)
6. Brody, B.: Identity and Essence. Princeton Univ. Press, Princeton, New Jersey (1980)
7. Chakraborty, M. K., Banerjee, M.: Saturatedness and a hierarchy of approximate identities. Int. J. Uncertainty, Fuzziness and Knowledge-Based Systems **2**(4), 429–444 (1994)
8. Chakraborty, M. K., Banerjee, M.: Rough dialogue and implication lattices. Fundamenta Informaticae **75**(1-4), 123–139 (2007)
9. Chakraborty, M. K., Orłowska, E.: Substitutivity principles in some theories of uncertainty. Fundamenta Informaticae **32**(2), 123–139 (1997)
10. Chatterjee, A., Chakraborty, M. K.: "Marry your daughter to a handsome person": the Nyāya technique of precisification. In: Chakraborty, M. K. et al. (eds.) Logic, Navya-Nyāya & Applications, Homage to Bimal Krishna Matilal, Studies in Logic, Vol. 15, pp. 65–79. College Publications, London (2008)
11. Conway, J. H.: On Numbers and Games. Academic Press (1976)
12. Gorsky, D.: Generalisation and Cognition. Progress Publishers, Moscow (1987)
13. Gottwald, S.: Fuzzy set theory with t-norms and ϕ-operators. In: Di Nola, A., Ventre, A. G. S. (eds.) The Mathematics of Fuzzy Systems, pp. 143–195. TÜV Rheinland, Köln (1980)
14. Grzymała-Busse, J. W.: Rough set strategies to data with missing attribute values. In: Raś, Z. W. and Zemankova, M. (eds.) ISMIS 1991, LNCS, Vol. 542, pp. 368–377. Springer (1991)
15. Hodes, H. T.: The modal theory of pure identity and some related decision problems. Zeitschr. für Math. logik und Grund. der Math. **30**, 415–423 (1984)
16. Hodges, W.: Elementary predicate logic. In: Gabbay, D., Guenthner, D. (eds.) *Handbook of Philosophical Logic*, Vol. 1, pp. 1–131. Reidel, Dordrecht (1983)
17. Horn, L.: A Natural History of Negation. Univ. of Chicago Press, Chicago (1989)
18. Jacas, J.: On the generators of T-indistinguishability operator. Stochastica **XII**(1), 49–63 (1988)
19. Khan, M. A., Banerjee, M.: Formal reasoning with rough sets in multiple-source approximation systems. Int. J. Approximate Reasoning **49(2)**, 466–477 (2008)
20. Kryszkiewicz, M.: Rough set approach to incomplete information systems. Information Sciences **112**, 39-49 (1998)
21. Morgan, Ch.: A theory of equality for a class of many-valued predicate calculi. Zeitschrift für Math. Logik und Grund. der Math. **20**, 427–432 (1974)
22. Morgan, Ch.: Similarity as a theory of graded equality for a class of many-valued predicate calculi. In: Proc. Int. Symp. Multiple-valued Logic 1975, pp. 436–449. New York (1975)
23. Munitz, M. K.: (Ed.) Identity and Individuation. New York Univ. Press, New York (1971)
24. Novak, V.: Fuzzy Sets and their Applications. Adam Hilger, Bristol and Philadelphia (1986)
25. Novotný, M., Pawlak, Z.: On rough equalities. Bull. Polish Acad. Sc. (Th. Comp. Sc.) **33**(1-2), 99–104 (1985)
26. Obtułowicz, A.: Rough Sets and Heyting Algebra Valued Sets. Bull. Polish Acad. Sc. (Mathematics) **13**(9-10), 667–671 (1987)
27. Pawlak, Z.: Rough sets. Int. J. Comp. Inf. Sci. **11**, 341–356 (1982)
28. Pawlak, Z.: Rough sets and fuzzy sets. Fuzzy Sets and Systems **17**, 99–102 (1985)
29. Pawlak, Z.: Rough Sets. Theoretical Aspects of Reasoning about Data. Kluwer Academic Publishers, Dordrecht (1991)
30. Pawlak, Z.: Rough sets, present state and further prospects. Technical Report, Warsaw University of Technology 15/19 (1995)
31. Pawlak, Z.: A treatise on rough sets. Transactions on Rough sets **IV**, 1–17 (2005)
32. Pultr, A.: Fuzziness and fuzzy equality. In: Skala, H. J., Termini, S., Trillas, E. (eds.) Aspects of Vagueness, pp. 119–135. D. Reidel Pub. Co., Dordrecht (1984)

33. Quine, W. V. O.: Set Theory and its Logic. Belknap Press, Cambridge, Massachusetts (1978)
34. Raju, P. T.: The principle of four-cornered negation in Indian philosophy. Review of Metaphysics **7**, 694–713 (1954)
35. Savellos, E. E.: On defining identity. Notre Dame J. of Formal Logic **31**(3), 476–484 (1990)
36. Shapiro, S.: Vagueness in Context. Clarendon Press, Oxford (2006)
37. Skala, H. J., Termini, S., Trillas, E.: Aspects of Vagueness. D. Reidel Pub. Co., Dordrecht, Holland (1984)
38. Tarski, A.: Introduction to Logic and to the Methodology of Deductive Sciences. Oxford Univ. Press (1965)
39. Trillas, E., Valverde, L.: An inquiry into indistinguishability operators. In: Skala, H. J., Termini, S., Trillas, E. (eds.) Aspects of Vagueness, pp. 231–256. D. Reidel Pub. Co., Dordrecht (1984)
40. Valverde, V.: On the structure of F-indistinguishability operators. Fuzzy Sets and Systems **17**, 313–328 (1985)
41. Waszkiewicz, J.: The notions of isomorphism and identity for many-valued relational structures. Studia Logica **XXVII**, 93–98 (1971)
42. Wiggins, W.: Sameness and Substance. Harvard Univ. Press, Cambridge, Massachusetts (1980)
43. Wyler, O.: Lecture Notes on Topoi and Quasitopoi. World Scientific, Singapore (1991)
44. Zadeh, L. A.: Fuzzy sets. Inf. Control **8**, 338–353 (1965)
45. Zadeh, L. A.: Similarity relations and fuzzy orderings. Inf. Sci. **3**, 177–200 (1971)

Chapter 2
Preliminaries of Rough Set Theory

We present some basic notions of rough set theory [**?**]. Some of these have already been introduced in Chapter 1, but we recall them for the sake of continuity.

2.1 Approximations of sets

Definition 2.1. An *approximation space* is a pair (U,R), where U is a non-empty set (the domain of discourse), and R an equivalence relation on it.

R represents indiscernibility at the object level. The idea is to approximately describe a concept, extensionally represented by a subset A (say) of U, in the context of the domain categorized/partitioned by the relation R. For this purpose, we have the *lower approximation* $\underline{A}_R$ and *upper approximation* $\overline{A}_R$ of A; the former approximates A from within, and the latter from outside. A goal could be to refine the partition so that the lower approximation becomes larger while the upper becomes smaller, until the two coincide with the set A. In that case, A is said to be *definable*. We elaborate on this in Section 2.1.2.

Let $[x]$ denote the equivalence class in U of the object x with respect to the relation R. Recall Definition 1.1.

Definition 2.2.

(a) The *lower approximation* of a subset A of U, is defined as the set

$$\underline{A}_R \equiv \{x \in U : [x] \subseteq A\}.$$

(b) The *upper approximation* of A is defined as the set

$$\overline{A}_R \equiv \{x \in U : [x] \cap A \neq \emptyset\}.$$

(c) The *boundary* of A is defined as the set

$$Bn_R(A) \equiv \overline{A} \setminus \underline{A}.$$

Example 2.1. **

Exercise. For approximation spaces $(U,R_1),(U,R_2)$, if $R_1 \subseteq R_2$ then $\underline{A}_{R_2} \subseteq \underline{A}_{R_1}$, and $\overline{A}_{R_1} \subseteq \overline{A}_{R_2}$.
In other words, as mentioned above, the approximations become 'better' as the partition on U becomes finer. Equivalently, the boundary of the set becomes thinner as the partition becomes finer.

If there is no confusion, we shall drop the suffix R from the notations in Definition 2.2.

2.1.1 Some elementary properties

Let (U,R) be an approximation space, and $A \subseteq U$. The following is an easy exercise. $|A|$ denotes the cardinality of the set A.

Proposition 2.1.

1. $\underline{A} \subseteq A \subseteq \overline{A}$, $\underline{U} = U = \overline{U}$, $\underline{\emptyset} = \emptyset = \overline{\emptyset}$.
2. $(\underline{A})^c = \overline{(A^c)}$, $(\overline{A})^c = \underline{(A^c)}$.
3. *(a)* $\underline{A} \cap \underline{B} = \underline{A \cap B}$, (b) $\overline{A} \cup \overline{B} = \overline{A \cup B}$.
4. *(a)* $\underline{A} \cup \underline{B} \subseteq \underline{A \cup B}$, (b) $\overline{A \cap B} \subseteq \overline{A} \cap \overline{B}$.
5. $A \subseteq B$ *implies* $\underline{A} \subseteq \underline{B}$, $\overline{A} \subseteq \overline{B}$.
6. $\underline{\underline{A}} = \underline{A}$, $\overline{\overline{A}} = \overline{A}$.
7. $Bn(A) = \emptyset$, *if and only if* $\underline{A} = \overline{A}$.
8. $[x] \subseteq Bn(A)$ *implies* $|[x]| \geq 2$. *In other words, an elementary set which is a singleton, cannot be a constituent of* $Bn(A)$.

Observation 1 It is easy to furnish examples of sets in approximation spaces where the converse of 4 (a) and (b) do not hold. (*Exercise*).

We shall see some implications of this observation in Chapter 9, on deductive reasoning with rough sets.

2.1.2 Definable sets

The equivalence classes under R are called *elementary sets* of (U,R). When the boundary $Bn(A)$ of a set A in (U,R) is empty, it means that A is exactly describable by unions of elementary sets.

Definition 2.3. A set $A \subseteq U$ is *definable* (or *exact*) in (U,R), if and only if $\overline{A} = \underline{A}$.

Example 2.2. **

We then have

Observation 2

1. *$\emptyset, U, \underline{A}, \overline{A}, Bn(A)$ are definable.*
2. *Any elementary set is definable.*
3. *(i) $\underline{A}$ is the maximal definable set contained in A,*
   (ii) $\overline{A}$ is the minimal definable set containing A

One may now introduce the following kinds of sets in (U, R). representing different 'shades' of definability.

Definition 2.4.
(a) A is *roughly definable*, if and only if $\underline{A} \neq \emptyset$ and $\overline{A} \neq U$.
(b) A is *internally undefinable*, if and only if $\underline{A} = \emptyset$ and $\overline{A} \neq U$.
(c) A is *externally undefinable*, if and only if $\underline{A} \neq \emptyset$ and $\overline{A} = U$.
(d) A is *totally undefinable*, if and only if $\underline{A} = \emptyset$ and $\overline{A} = U$.

The following are easy to show.
Exercise. Let $A \subseteq U$.
(a) A is definable (roughly definable/totally undefinable), if and only if so is A^c.
(b) A is externally (internally) undefinable, if and only if A^c is internally (externally) undefinable.

2.2 Rough Inclusions and Equalities

Given two subsets A and B of U, there are the following nine possibilities.

(i) $\underline{A} \subseteq \underline{B}$, (ii) $\underline{A} \subseteq B$, (iii) $\underline{A} \subseteq \overline{B}$,
(iv) $A \subseteq \underline{B}$, (v) $A \subseteq B$, (vi) $A \subseteq \overline{B}$,
(vii) $\overline{A} \subseteq \underline{B}$, (viii) $\overline{A} \subseteq B$, (ix) $\overline{A} \subseteq \overline{B}$.

Some of these can be deduced from others. For instance, (i) and (ii) are deducible from each other. In fact, it can be shown that five sets of mutually equivalent inclusion relations emerge, and form the *implication lattice* (Fig. 2.1) [**?**]. The inclusion at an arrow-head is deducible from the one at its tail-end.

Let $A, B \subseteq U$.

Definition 2.5.
(a) A is *roughly lower (upper) included* in B, denoted $A \underset{\sim}{\subseteq} B$ $(A \overset{\sim}{\subset} B)$, if and only if $\underline{A} \subseteq \underline{B}$ $(\overline{A} \subseteq \overline{B})$.
(b) A is *roughly included* in B, denoted $A \underset{\sim}{\overset{\sim}{\subseteq}} B$, if and only if $\underline{A} \subseteq \underline{B}$ and $\overline{A} \subseteq \overline{B}$.
(c) A is *roughly lower (upper) equal* to B, denoted $A \eqsim B$ $(A \simeq B)$, if and only if $\underline{A} = \underline{B}$ $(\overline{A} = \overline{B})$.

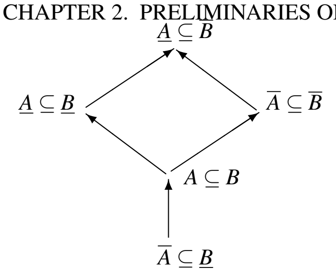


Fig. 2.1 The implication lattice

(d) A is *roughly equal* in B, denoted $A \approx B$, if and only if $\underline{A} = \underline{B}$ and $\overline{A} = \overline{B}$.

Example 2.3. **

Clearly, the relations $\eqsim$, $\simeq$ and $\approx$ are equivalence relations on the power set $P(U)$ of U, thus yielding the quotient sets $P(U)/\eqsim$, $P(U)/\simeq$ and $P(U)/\approx$ respectively. Rough equality may be regarded as an indiscernibility at the concept level indicating that, relative to the given partition of the domain, one is unable to discern between the sets concerned.

We also have the following properties, for any $A, B, A', B' \subseteq U$.

Proposition 2.2.

1. (a) *$A \eqsim B$, if and only if $A \cap B \eqsim A$ and $A \cap B \eqsim B$.*
   (b) *$A \simeq B$, if and only if $A \cup B \simeq A$ and $A \cup B \simeq B$.*
2. (a) *$A \simeq A'$ and $B \simeq B'$ imply $A \cup B \simeq A' \cup B'$.*
   (b) *$A \eqsim A'$ and $B \eqsim B'$ imply $A \cap B \eqsim A' \cap B'$.*
3. (a) *$A \simeq B$ implies $A \cup B^c \simeq U$.*
   (b) *$A \eqsim B$ implies $A \cap B^c \eqsim \emptyset$.*
4. (a) *$A \subseteq B$ and $B \simeq \emptyset$ imply $A \simeq \emptyset$.*
   (b) *$A \subseteq B$ and $B \simeq U$ imply $A \simeq U$.*
5. *$A \simeq B$, if and only if $A^c \eqsim B^c$.*
6. (a) *$A \eqsim \emptyset$ or $B \eqsim \emptyset$ imply $A \cap B \simeq \emptyset$.*
   (b) *$A \simeq U$ or $B \simeq U$ imply $A \cup B \simeq U$.*
7. *$\underline{A} = \bigcap\{Y \subseteq U : A \eqsim Y\}$, $\overline{A} = \bigcup\{Y \subseteq U : A \simeq Y\}$.*

There is also a nice connection between $Up(R) \equiv \{\overline{Y} : Y \subseteq U\}$, the set of upper approximations of subsets of U, and the singleton equivalence classes in $P(U)/\approx$ [**?**]:

$$Up(R) = \{Y \subseteq U : \{Y\} \in P(U)/\approx\}.$$

One side is immediate, and the other follows from the fact that if Y is the only subset of U roughly equal to itself, it must be definable in (U, R).

We now have an enhanced implication lattice (Fig. 2.2) [**?**]. We shall remark on the significance of the implication lattices in Chapter 9, on deductive reasoning with rough sets.

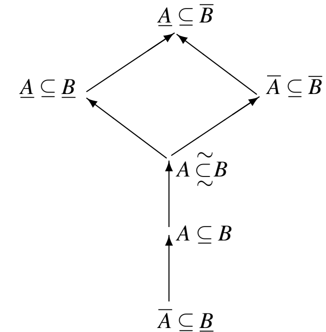


Fig. 2.2 The enhanced implication lattice

2.3 Rough Sets

Let us now come to the theme of our treatise, a 'rough set' in an approximation space (U,R). Interestingly, we find quite a few definitions so far, three of which are 'equivalent' to each other in a straightforward way. In fact, the equivalence could be extended to a general case, but under certain conditions. We give all the definitions below.

Definition 2.6. [**?**] $A \subseteq U$ is a *rough set* in (U,R), provided $Bn(A) \neq \emptyset$.

For generality's sake however, we could remove the restriction in the definition above, and term *any* subset A of U rough. A definable set then becomes a special case of a rough set. Moreover, to keep the context clear, we could have

Definition 2.7. The triple (U,R,A) is called a *rough set* [**?**].

Definition 2.8. (cf. [**?**]) The pair $(\underline{A},\overline{A})$, for each $A \subseteq U$, is called a *rough set* in (U,R).

Definition 2.9. [**?**] The pair $(\underline{A},\overline{A}^c)$, for each $A \subseteq U$, is called a *rough set* in (U,R).

Definition 2.10. [**?**] Given an approximation space (U,R), a *rough set* is an ordered quadruple (U,R,L,B), where (i) L,B are disjoint subsets of U, (ii) both L and B are definable sets in (U,R), and (iii) for each $x \in B$ there exists $y \in B$ such that $x \neq y$ and xRy (i.e. no equivalence class contained in B is a singleton).

Definition 2.11. [**?**] A *rough set* in (U,R) is an equivalence class of $P(U)/\approx$.

Remark 2.1. The above definition therefore identifies all roughly equal sets, and chooses a representative entity out of each such group to define a rough set.

It may be observed that Definitions 2.8, 2.9 and 2.11 are equivalent to each other for any given (U,R). This is because, for any $A \subseteq U$, the entities $(\underline{A},\overline{A})$, $(\underline{A},\overline{A}^c)$ and the equivalence class $[A]$ of A in $P(U)/\approx$ are identifiable. More formally, there are bijections between the following three families: one constituted of pairs of the form $(\underline{A},\overline{A})$, another of pairs of the form $(\underline{A},\overline{A}^c)$ and the third of equivalence classes $[A]$, where A ranges over all subsets of U. Again, for a fixed (U,R), a quadruple (U,R,L,B) is essentially the pair (L,B), and due to condition (iii) of Definition 2.10, one can always find a subset A of U such that $\underline{A} = L$ and $Bn(A) = B$. Hence Definition 2.10 may be reformulated as follows: the pair $(\underline{A},Bn(A))$ for each $A \subseteq U$ is a rough set so long as (U,R) remains unchanged. So, via this interpretation, Definition 2.10 also becomes equivalent to 2.8, 2.9 and 2.11. These definitions have been said to represent the 'set-oriented' view of rough sets, viz. when rough sets are defined through pairs of definable sets. In contrast, Definitions 2.6 and 2.7 are said to represent the 'operator-oriented' view [**?**], where the term 'rough' is used as an adjective for the subset A of U, and *operators* of lower and upper approximations are defined on U, in order to approximately describe A. Note that if we fix (U,R) and consider the power set over U and the family of pairs $(\underline{A},\overline{A})$, $A \subseteq U$, the natural map between the two associating A with the pair $(\underline{A},\overline{A})$ may not be injective, as any set B roughly equal but not equal to A would also map to the same pair.

We should remark here that, starting with the equivalent definitions mentioned above, one arrives at different (though related) algebras, by taking different definitions of union, intersection, complementation and other algebraic operations. We shall encounter these in Chapter 6. In the next section, we indicate some possibilities of defining intersections and unions of rough sets.

2.4 Intersections and Unions

Recall from Observation 1 that, in general, the equalities may not hold in (1) and (2), for a given approximation space (U,R). It is clear that, for $A,B \subseteq U$,

Observation 3

(a) $\underline{A\cup B} = \underline{A}\cup\underline{B}$, *if and only if there is no equivalence class* $[x]$ *in* U *such that* $[x] \subseteq A\cup B$, $[x] \not\subseteq A$ *and* $[x] \not\subseteq B$.

(b) $\overline{A\cap B} = \overline{A}\cap\overline{B}$, *if and only if there is no equivalence class* $[x]$ *in* U *such that* $[x]\cap A \neq \emptyset \neq [x]\cap B$, *but* $[x]\cap(A\cap B) = \emptyset$.

So one may ask if there are subsets C,D of U such that, in the approximation space (U,R), $\underline{A}\cup\underline{B} = \underline{C}$, $\overline{A}\cup\overline{B}$ $(=\overline{A\cup B}) = \overline{C}$, and $\underline{A}\cap\underline{B}$ $(=\underline{A\cap B}) = \underline{D}$, $\overline{A}\cap\overline{B} = \overline{D}$. The answer is yes, and we now present three pairs (C_i,D_i), $i = 1,2,3$ of such sets, which then clearly must be coordinate-wise roughly equal.

2.4.1 (C_1,D_1)

Taking a cue from Observation 3(a), we determine the set C_1. Include in C_1, one of the sets, say A, the lower approximation of the other ($\underline{B}$), and add all the elements of $A\cup B$ that are in $Bn(A\cup B)$. In other words,

$$C_1 \equiv A \cup \underline{B} \cup ((A \cup B) \cap Bn(A \cup B)).$$

So, effectively, equivalence classes of the kind mentioned in Observation 3(a) are not contained in C_1.

To obtain D_1, we add to $A \cap B$, all elements of A (or B) that lie in the equivalence classes mentioned in Observation 3(b), i.e. those $[x]$ which are within $\overline{A} \cap \overline{B}$, but outside $\overline{A \cap B}$. So

$$D_1 \equiv (A \cap B) \cup (A \cap (\overline{A} \cap \overline{B}) \cap (\overline{A \cap B})^c).$$

Denote C_1 by $A \sqcup B$, and D_1 by $A \sqcap B$. We then have

Proposition 2.3.

1. $\underline{A \sqcup B} = \underline{A} \cup \underline{B}$, $\overline{A \sqcup B} = \overline{A} \cup \overline{B}$.
2. $\underline{A \sqcap B} = \underline{A} \cap \underline{B}$, $\overline{A \sqcap B} = \overline{A} \cap \overline{B}$.
3. $A \sqcup B \approx B \sqcup A$; $A \sqcap B \approx B \sqcap A$.
4. $A \sqcap B \approx (A^c \sqcup B^c)^c$; $A \sqcup B \approx (A^c \sqcap B^c)^c$.

2.4.2 (C_2, D_2)

Another way to come up with the sets C and D is as follows [**?**].

Definition 2.12. An *upper sample* P of A in (U,R) is a subset of U such that $P \subseteq A$ and $\overline{P} = \overline{A}$. An upper sample P is *minimal*, if there is no upper sample Z of A with $Z \subset P$.

Observe that $A \sqcup B$ of Section 2.4.1 is an upper sample of $\overline{A} \cup \overline{B}$.
A little reflection will show that *any* minimal upper sample P_0 of A must be formed by taking exactly one element of A from every equivalence class included in $\overline{A}$. So, if $[x] \subseteq P_0$, then $[x] = \{x\}$ necessarily.

Proposition 2.4. *Any two minimal upper samples of A are roughly equal.*

Proof. Let P, P' be two minimal upper samples of A. By definition, $\overline{P} = \overline{A} = \overline{P'}$. $[x] \subseteq P \Rightarrow [x] = \{x\} \Rightarrow [x] \subseteq P'$, and similarly the converse. Thus $\underline{P} = \underline{P'}$.

Let P be a minimal upper sample of $\overline{A} \cup \overline{B}$, and P' be a minimal upper sample of $\overline{A} \cap \overline{B}$. We take $C_2 \equiv \underline{A} \cup \underline{B} \cup P$, and $D_2 \equiv (\underline{A} \cap \underline{B}) \cup P'$.

Proposition 2.5.

(a) $\overline{\underline{A} \cup \underline{B} \cup P} = \overline{A} \cup \overline{B}$.
(b) $\underline{\underline{A} \cup \underline{B} \cup P} = \underline{A} \cup \underline{B}$.
(c) $\overline{(\underline{A} \cap \underline{B}) \cup P'} = \overline{A} \cap \overline{B}$.
(d) $\underline{(\underline{A} \cap \underline{B}) \cup P'} = \underline{A} \cap \underline{B}$.

Proof. Let us prove (a) and (b). We have $P \subseteq \overline{A} \cup \overline{B}$ and $\overline{P} = \overline{\overline{A} \cup \overline{B}} = \overline{A} \cup \overline{B}$. Moreover, P is formed by taking exactly one element from every equivalence class in $\overline{A} \cup \overline{B}$.

(a) $\overline{\underline{A} \cup \underline{B} \cup P} = \underline{A} \cup \underline{B} \cup \overline{P} = \underline{A} \cup \underline{B} \cup \overline{A} \cup \overline{B} = \overline{A} \cup \overline{B}$.

(b) Let $[x] \subseteq \underline{A} \cup \underline{B} \cup P$. We show that $[x] \subseteq \underline{A} \cup \underline{B}$. If $[x] \subseteq \underline{A}$ or $[x] \subseteq \underline{B}$, there is nothing to prove. As we cannot have part of $[x]$ inside $\underline{A}$ (or $\underline{B}$) and part in P, the only remaining possibility is $[x] \subseteq P$. But then $[x] = \{x\}$. $P \subseteq \overline{A} \cup \overline{B} = \overline{A \cup B}$ implies that $\{x\} = [x] \cap (A \cup B) \neq \emptyset$, i.e. $x \in A$ or $x \in B$. Thus $[x] \subseteq \underline{A} \cup \underline{B}$.

Conversely, $\underline{A} \cup \underline{B} \subseteq \underline{A} \cup \underline{B} \cup P$ implies that $\underline{A} \cup \underline{B} = \underline{\underline{A} \cup \underline{B}} \subseteq \underline{\underline{A} \cup \underline{B} \cup P}$.

2.4.3 (C_3, D_3)

Yet another approach [**?**] is to consider a set B_0 roughly equal to B:

$$B_0 \equiv \underline{B} \cup (B \cap \overline{A}^c) \cup (\overline{B} \cap A \setminus \underline{A}) \cup (B \cap \underline{A}).$$

Define $C_3 \equiv A \cup B_0$, and $D_3 \equiv A \cap B_0$. One can then show that

Proposition 2.6.
(a) $B_0 \approx B$.
(b) $\underline{A} \cup \underline{B} = \underline{A \cup B_0}$ *and* $\overline{A} \cap \overline{B} = \overline{A \cap B_0}$.

(*Exercise*).

An important question that arises in this context is whether we can construct the sets C and D *without referring to the lower and upper approximations of* A *and* B. This is an open problem.** From latest work by Dubois and Ciucci

2.5 Rough Membership Functions

The notion of rough membership function was defined by Pawlak and Skowron in [] and applied to develop rough mereology []. They defined it with respect to the approximation space (X, R), where X was taken to be finite. We extend the notion to an arbitrary set X. However, the equivalence classes or blocks $[.]_R$ generated by R are all of finite cardinality.

Definition 2.13. Given a subset A of X, a *rough membership function* f_A is a mapping from X to $Ra[0,1]$, the set of rational numbers in $[0,1]$, defined by

$$f_A(x) = \frac{Card([x]_R \cap A)}{Card[x]_R},$$

for all $x \in X$.

The rough membership function can also be interpreted as the conditional probability that x belongs to A, given the partition R which may be induced by a set of attributes B. This interpretation was used by several researchers in the rough set community [**?**, **?**, **?**, **?**, **?**, **?**, **?**] . Note also that the ratio on the right hand side of the equation (**??**) is known as the confidence coefficient in data mining [**?**, **?**]. It is worthwhile to mention that set inclusion to a degree has been considered by Łukasiewicz [**?**] in studies on assigning fractional truth values to logical formulas.

One can observe that the rough membership function has the following properties.

Proposition 2.7.

(i) $f_A(x) = 1$ *if and only if* $x \in \underline{A}$.
(ii) $f_A(x) = 0$ *if and only if* $x \in (\overline{A})^c$.
(iii) $0 < f_A(x) < 1$ *if and only if* $x \in Bd(A) := \overline{A} \setminus \underline{A}$.
(iv) $f_A(x) = f_A(y)$ *for* xRy.

Observation 4 *Each block* $[.]_R$ *being finite, there is a fixed set of rational numbers in* $[0,1]$ *that are admissible values for the elements of the block. If* $Card([.]_R) = n$*, then* $Card([.]_R \cap A)$ *may be exactly one of the numbers* $0, 1, \ldots, n-1, n$*, and hence for any* $x \in X$*,* $f_A(x)$ *shall be exactly one of the values* $\{0, 1/n, \ldots, n-1/n, 1\}$*. The set of admissible values associated with* $[.]_R$ *is determined right at the beginning, when the partition is formed in* X*. It will be denoted by* $admiss - value[.]_R$*. Under a rough membership function* f_A*, all the elements of* $[.]_R$ *receive the same value out of the set* $admiss - value[.]_R$*. This value shall also be referred to as the value of the block under* f_A*, and denoted by* $f_A([.]_R)$.

Example 2.4. from Anirban's paper**

We obtain further properties of rough membership function.

Proposition 2.8.

1. *If* $f_A = f_B$ *then* $A \approx B$*, but the converse does not hold.*
2. *If* $A \approx B$ *(A is roughly equal to B) then* $f_A(x) = 1$ *if and only if* $f_B(x) = 1$ *and* $f_A(x) = 0$ *if and only if* $f_B(x) = 0$.
3. *If for some* $A,\ x \in X,\ 0 < f_A(x) < 1$ *then there exists* $B \neq A$ *such that* $f_A = f_B$.
4. $f_{A^c}(x) = 1 - f_A(x)$*, for all* $A, x \in X$.
5. *If* $A \subseteq B$ *then* $f_A \leq f_B$*, but the converse does not hold.*
6. *If* $f_A \leq f_B$ *then* $A \underset{\sim}{\overset{\sim}{\subseteq}} B$*, i.e. A is roughly included in B.*
7. $max[0, f_A(x) + f_B(x) - 1] \leq f_{A \cap B}(x) \leq min[f_A(x), f_B(x)]$.
8. $max[f_A(x), f_B(x)] \leq f_{A \cup B}(x) \leq min[1, f_A(x) + f_B(x)]$.
9. $f_{A \cup B}(x) = f_A(x) + f_B(x) - f_{A \cap B}(x)$.

Proof. Proof of (vii), (viii) and (ix) from Yao.**

Exercise. Prove (i)-(vi) of the above proposition.

From the properties presented in Proposition 2.8 it follows that the rough membership differs essentially from the fuzzy membership [**?**], for properties (7) and (8) show that the membership for intersection and union of sets, in general, cannot be computed – as in the case of fuzzy sets – from their constituents' membership. Thus formally the rough membership is different from fuzzy membership. Moreover, the rough membership function depends on an available knowledge (represented by the partition induced by a set of attributes). Besides, the rough membership function, in contrast to fuzzy membership function, has a probabilistic flavor.

Definition 2.14. Let $\equiv$ be the binary relation defined on the power set $P(X)$ by

$$A \equiv B \text{ if and only if } f_A = f_B.$$

Clearly, $\equiv$ is an equivalence relation generating a partition on $P(X)$.

Observation 5 *On $P(X)$, an equivalence relation $\approx$ (rough equality) has already been defined, giving rise to rough sets (cf. Definition 2.11). Now, here is another equivalence relation $\equiv$ – it generates a finer partition on $P(X)$. That is, the power set $P(X)$ receives two partitions due to the relations $\approx$ and $\equiv$ such that each equivalence class $[.]_{\approx}$ is the union of some equivalence classes due to $\equiv$.*

We call an equivalence class $[.]_{\equiv}$ a *membership function based rough set*, or *MF-rough set*, while $[.]_{\approx}$ is a rough set vide Definition 2.11. An MF-rough set is a rough set if and only if $A \approx B$ implies $f_A = f_B$.

Example 2.5. in consonance with previous example.**

We now establish the following crucial theorems for rough membership functions. Let us note that there are two operations $\wedge$ and $\vee$ on the set of all rough membership functions, defined componentwise, with the help of respectively the 'min' and 'max' operations on $Ra[0,1]$. In other words, for all $x \in X$,

$$(f_A \wedge f_B)(x) := min(f_A(x), f_B(x)), (f_A \vee f_B)(x) := max(f_A(x), f_B(x)).$$

Likewise, using the natural order relations $\leq, <, \geq, >$ on $Ra[0,1]$, we have induced order relations, also denoted $\leq, <, \geq, >$ respectively, on the set of all rough membership functions.

Theorem 2.1. *Given (X,R) and subsets A, B of X, there exists a subset P of X such that $f_P = f_A \wedge f_B$, that is, for all $x \in X$, $f_P(x) = min(f_A(x), f_B(x))$.*

Proof. The set P is constructed as follows.

(i) Elements of all blocks in $\underline{A} \cap \underline{B}$ are included.

(ii) Elements of all blocks in $(\overline{A})^c \cup (\overline{B})^c$ are excluded.

(iii) From any other block $[.]_R$ (that is, from outside the two above regions of (i) and (ii)), all the elements belonging to either $[.]_R \cap A$ or $[.]_R \cap B$ are included, choosing those from the set with a smaller cardinality. The rest of the elements of $[.]_R$ are excluded.

Exercise. Show that P thus formed, satisfies the property as stated in Theorem 2.1.

**To give the following, or leave as exercise?

Theorem 2.2. *Given (X,R) and subsets A,B of X, there exists a subset Q of X such that $f_Q = f_A \vee f_B$, that is, for all $x \in X$, $f_P(x) = max(f_A(x), f_B(x))$.*

Proof. The set Q is constructed as follows.

(i) Elements of all blocks in $\underline{A} \cup \underline{B}$ are included.

(ii) Elements of all blocks in $(\overline{A})^c \cap (\overline{B})^c$ are excluded.

(iii) From any other block $[.]_R$ (that is, from outside the two above regions of (i) and (ii)), all the elements belonging to either $[.]_R \cap A$ or $[.]_R \cap B$ are included, choosing those from the set with a greater cardinality. The rest of the elements of $[.]_R$ are excluded.

Exercise. Show that Q thus formed, satisfies the property as stated in Theorem 2.2.

Corollary 2.1. *The subsets P and Q defined in Theorems 2.1 and 2.2 satisfy the following properties.*

$$\underline{P} = \underline{A} \cap \underline{B}, \quad \overline{P} = \overline{A} \cap \overline{B}$$
$$\underline{Q} = \underline{A} \cup \underline{B}, \quad \overline{Q} = \overline{A} \cup \overline{B}.$$

Proof. Easy exercise.

Remark 2.2. Recall that, in Section 2.4, we had presented three pairs (C_i, D_i), $i = 1,2,3$ of sets to answer the question about distribution of the lower and upper approximation operators over union and intersection. Corollary 2.1, in fact, gives us a fourth such pair of sets.

Apart from $\wedge$ and $\vee$, another binary operator $\Rightarrow$ may be defined for rough membership functions. This operation would play a vital role in the development of a logic for these functions.

Definition 2.15. For all $x \in X$, $(f_A \Rightarrow f_B)(x) := f_A(x) \Rightarrow f_B(x)$, where $\Rightarrow$ is the Gödelian implication function on the set $Ra[0,1]$, viz.

$$a \Rightarrow b := \begin{cases} 1 \text{ if } a \leq b \\ b \text{ if } a > b. \end{cases}$$

Theorem 2.3. *Given $A, B \subseteq X$, $f_A \Rightarrow f_B = f_S$, for some $S \subseteq X$.*

Proof. If $f_A \leq f_B$ then $(f_A \Rightarrow f_B)(x) = 1$, for all $x \in X$. So, in this case, $f_A \Rightarrow f_B = f_X$. If $f_A > f_B$, $(f_A \Rightarrow f_B)(x) = f_B(x)$, for all $x \in X$, whence in this case, $f_A \Rightarrow f_B = f_B$. If for some $x \in X$, $f_A(x) \leq f_B(x)$ and for some $y \in X$, $f_A(y) > f_B(y)$, then $(f_A \Rightarrow f_B)(x) = 1$ and $(f_A \Rightarrow f_B)(y) = f_B(y)$. In this case, a subset S of X is defined as

$$S := \{\bigcup_x [x]_R | f_A(x) \leq f_B(x)\} \cup \{\bigcup_y (B \cap [y]_R) | f_A(y) > f_B(y)\}.$$

So, $f_S(x) = 1 = f_A(x) \Rightarrow f_B(x)$ and $f_S(y) = f_B(y) = f_A(y) \Rightarrow f_B(y)$.
Thus, in all cases, there is S such that $f_A \Rightarrow f_B = f_S$.

Proposition 2.9. *Given (X,R), the set $\{f_A | A \subseteq X\}$ is closed with respect to the operations $\wedge, \vee, \Rightarrow, ^c$, where the unary complement operation ^{c} is given by*

$$f_A^c(x) := 1 - f_A(x),$$

for any $x \in X$.

Exercise. Prove the closure property for the complement operation ^{c} .

2.5.1 Some perspectives on the concept of rough membership

Fuzzy set theory starts in solving problems with some primitive membership functions and granulates them aiming at inducing target membership functions corresponding, e.g, to concepts expressed in natural language. The fuzziness in granules and their values characterize the ways in which human concepts of granulation are formed, organized and manipulated .

The rough set theory provides an effective model to discover knowledge from decision systems with upper approximation and lower approximation of decision classes as its core concepts and in making decisions according to the definition of indistinguishibility (indiscernibility) relation and attribute reducts. Different variants of the conventional rough sets are available mainly by redefining the indistinguishibility relation and approximation operators. The rough set approach (RS) can be used to granulate a set of objects into information granules (IGs). The granulation process is aiming at inducing relevant granules for approximation of target complex vague concepts.

Both fuzzy and rough set theory represent two different approaches to vagueness. Fuzzy set theory addresses *gradualness* of knowledge, expressed by the fuzzy membership, whereas rough set theory addresses *granularity* of knowledge, expressed by the indiscernibility relation [**?**]. These two approaches are using different logical languages for expressing granules and combination of these languages proved to be very useful in discovering relevant granules for approximation of complex vague concepts which work as complementary on collections of borderline cases leading to improvement of the granulation quality (see, e.g., [**?**, **?**, **?**]).

In the perspective of knowledge transformation, the task of analysing data and solving problems by fuzzy sets or rough sets is actually to induce a mapping from the sensory information granules represented by the original finest-grained data to the compound knowledge often represented in natural language. These approaches are also integrated synergistically within themselves and with other knowledge acquisition models, which yield, e.g., rough neural computing, neural fuzzy computing and rough fuzzy computing. The rough-fuzzy computing provides a strong paradigm, than fuzzy sets or rough sets separately in handling uncertainty arising both from the overlapping characteristics of concepts/ classes/regions and granularity in the domain of discourse [**?**].

One of the consequences of perceiving objects by information about them is that for some objects one cannot decide if they belong to a given set or not. However, one can estimate the degree to which objects belong to sets. This is a crucial observation in building foundations for approximate reasoning. Dealing with imperfect knowledge implies that one can only characterize satisfiability of relations between objects to a degree, not precisely. One of the fundamental relations on objects is a rough inclusion relation describing that objects are parts of other objects to a degree. The rough mereological approach [**?**, **?**, **?**, **?**, **?**] based on such a relation is an extension of the Leśniewski mereology [**?**] .

2.6 Roughness of a set

** Accuracy measure

2.7 Issues

**

References

1. Banerjee, M., Chakraborty, M. K.: A category for rough sets. Foundations of Computing and Decision Sciences **18**(3-4), 167–180 (1993)
2. Bonikowski, Z.: A certain conception of the calculus of rough sets. Notre Dame J. of Formal Logic **33**, 412–421 (1992)
3. Gehrke, M., Walker, E.: The structure of rough sets. Bull. Polish Acad. Sc. (Math.) **40**, 235–245 (1992)
4. Komorowski, J., Pawlak, Z., Polkowski, L. and Skowron, A.: Rough sets: A tutorial. In: Pal, S. K., Skowron, A. (eds.) Rough Fuzzy Hybridization: A New Trend in Decision–Making, pp. 3–98. Springer-Verlag, Singapore (1999)
5. Pagliani, P.: Rough set theory and logic-algebraic structures. In: Orłowska, E. (ed.) Incomplete Information: Rough Set Analysis, pp. 109–190. Springer Physica-Verlag (1998)
6. Yao, Y.: Two views of the theory of rough sets in finite universes. Int. J. Approx. Reasoning **15**, 291–317 (1996)

Chapter 3
Information Systems

As mentioned in Chapter 1, a realization of an approximation space is obtained through an *information system*. In this chapter, we discuss this in detail.

An information system is represented by a data table containing rows labeled by objects of interest, columns labeled by attributes and entries of the table are attribute values. For example, a data table can describe a set of patients in a hospital. The patients can be characterized by some attributes, like *age, gender, blood pressure, body temperature*, etc. With every attribute a set of its values is associated, *e.g.*, values of the attribute *age* can be *young, middle*, and *old*. Attribute values can be also numerical. In data analysis, the basic problem that we are interested in is to find the *patterns* in data, i.e., to find a relationship between some sets of attributes; for example, one might be interested in knowing whether *blood pressure* depends on *age and gender*. In this chapter, we present different kinds of information systems, depending on the availability of information about the objects of the domain with respect to a set of attributes, nature of clustering objects based on their attribute values etc.

3.1 Deterministic information system

When the information about a set of object with respect to a set of attributes is completely available the corresponding information system is deterministic in nature. What do we mean by 'complete information' is explained below.

Definition 3.1. A tuple $\mathscr{S} := (U, C, \{Val_a\}_{a\in C}, f)$ is called a *deterministic information system* (DIS), where

- U is a non-empty set of *objects*;
- C is a non-empty set of *attributes* or *features*;
- Val_a is a non-empty set of *values* for each attribute a;
- $f : U \times C \to \cup\, \{Val_a :\ a \in C\}$ assigns a unique value from Val_a to each $f(x,a)$ for $x \in U$ and $a \in C$.

Equivalent to Definition 3.1, the notion of deterministic information system, or simply *information system* can also be defined as follows.

Definition 3.2. A pair $\mathbb{A} := (U,C)$ is called an *information system* [**?**], where U is a non-empty *universe* of *objects* and C is a non-empty set of *attributes*. Each attribute $a \in C$ is represented as a function $a : U \longrightarrow V_a$ where V_a is the set of values of the attribute a, called the *domain* of a.

We shall use either of these equivalent definitions in our discussion on information systems. One should note that any information system can be represented by a data table with the rows labeled by the objects and the columns labeled by the attributes[1]. The pair (x,a), where $x \in U$ and $a \in C$ defines the particular entry in the table representing the value $a(x)$.

Observe that in a deterministic information system, each object in the universe takes a definite and unique value for every attribute present in the information system. This is what is meant by the phrase 'complete information'. However, the situation may vary, and we get other notions of information system, some of which will be discussed in subsequent sections.

Consider the following example of an information system, where the universe of objects consists of toys, and the attributes are respectively colour, size and feel. The values corresponding to the attributes assigned to the toys by the function f is given as the entries of the table

Toys	Colour (C)	Size (S)	Feel (F)
O_1	Blue	Big	Hard
O_2	Blue	Big	Hard
O_3	Red	Medium	Hard
O_4	Red	Medium	Hard
O_5	Green	Medium	Soft
O_6	Green	Big	Soft

Note that toys O_1, O_2 assume the same values for all the attributes, and so they cannot be distinguished from each other with respect to the given information. Similar is the case forO_3 and O_4. Thus, comes the notion of *indiscernibility* of objects with respect to the available information.

Definition 3.3. Given a deterministic information system $\mathscr{S} := (U,C,\{Val_a\}_{a\in\mathscr{A}}, f)$ and a set $B \subseteq C$, an *indiscernibility relation* $Ind_{\mathscr{S}}(B)$ on U is defined as follows.

$$(x,y) \in Ind_{\mathscr{S}}(B), \text{ if and only if } f(x,a) = f(y,a) \text{ for all } a \in B.$$

It is easy to see that $Ind_{\mathscr{S}}(B)$ is an equivalence relation on U. Here we should note that with addition of attributes, the indiscernibility relation becomes finer.

Proposition 3.1. *Given an information system* $\mathscr{S} := (U,C,\{Val_a\}_{a\in\mathscr{A}}, f)$ *if* $B_1, B_2 \subseteq C$ *are such that* $B_1 \subseteq B_2$ *then* $Ind_{\mathscr{S}}(B_2) \subseteq Ind_{\mathscr{S}}(B_1)$.

[1] Note, that in statistics or machine learning such a data table is called a sample [**?**]. One can also compare data tables corresponding to information systems with relations in relational databases [**?**].

To illustrate the above proposition, let us consider the following table by adding a column for *shape* in the above mentioned table.

Toys	Colour	Size	Feel	Shape
O_1	Blue	Big	Hard	Round
O_2	Blue	Big	Hard	Round
O_3	Red	Medium	Hard	Oval
O_4	Red	Medium	Hard	Round
O_5	Green	Medium	Soft	Square
O_6	Green	Big	Soft	Oval

We can notice that the toys O_3, O_4 are indiscernible by the relation $Ind_{\mathscr{S}}(B)$, where $B = \{$Colour, Size, Feel$\}$; whereas they are discernible with respect to the indiscernibility relation $Ind_{\mathscr{S}}(C)$ where $C = \{\text{Colour}, \text{Size}, \text{Feel}, \text{Shape}\}$.

3.2 Non-deterministic information system

In the context of DIS, each object has exactly one value associated to each attribute. Sometimes this is also called *single-valued information system* [Lipski's work]. In contrast to the single-valued context, a notion of *multi-valued information system* is thought of as an information system where an object can assume multiple values with respect to an attribute. There might be many different interpretations for assuming multiple values. It could be the case that for an object it is known that it can assume some possible values for an attribute; however, exactly which value that is not known. Let us consider the following table.

Example 3.1.

Car	Price	Mileage	Size
1	High	High	Full
2	Low	{Average,High}	Full
4	High	Low	Full
5	Low	Average	Full
6	Low	High	Full

In the above example, a disjunctive interpretation of multiple values has been considered. There can be situations leading towards a conjunctive interpreation of multiple values too. For example, we can consider objects as persons and one of the attributes as *language* having values such as Bengali, English, French, Polish etc. Now for a particular person the value of the attribute 'language' can be a subset of the above languages. Here the interpretation of a set of values is conjunctive. There can be even more possibilities such as inclusive disjunction, exclusive disjunction. In the above mentioned structure of the information systems (Definitions 3.1, 3.2) one can bring in the required changes by moving from single value to multiple value, or even to missing value (discussed in the next section). However, in order to reflect the desired interpretation of a subset of values one needs to go further (see Relational attribute system []). For now, let us stick to the multiple-valued context with disjunctive interpretation as conjunctive interpretation does not lead to non-determinism.

Definition 3.4. A tuple $\mathscr{K} := (U, C, \{Val_a\}_{a\in C}, f)$ is called a *non-deterministic information system* (NDIS) where U, C and Val_a are the same as in Definition 3.1. The function $f : U \times C \to 2^V$, where $V = \cup\{Val_a : \; a \in C\}$, satisfies $f(x,a) \subseteq Val_a$, for $x \in U$, $a \in C$.

It is to be noted that a deterministic information system is a special case of a non-deterministic information system, where $f(x,a)$ is a singleton set.

In the context of DIS, we have observed that based on the values of a set of attributes, say $B \subseteq C$, we can cluster a subset of objects from U as indistinguishable, and thus U gets partitioned by some subsets of indistinguishable objects. Now question arises how to have a parallel notion of indistinguishability (or distinguishability) in the context of NDIS as here from $f(x,a)$ it does not get reflected exactly which value x assumes for the attribute a. In this regard, in the literature [**?**] one can find different notions of indistinguishability (or distinguishability), induced by different relations other than that of indiscernibility relation. A few of them are presented below.

One straightforward way of generalizing the notion of indiscernibility can be just considering the same definition as presented in Definition 3.3. Such a notion of indistinguishability is called *strong indiscernibility* []. A notion of *weak indiscernibility*, denoted as *WInd*, can be obtained just by relaxing the constraint in the Definition 3.3 by the following condition.

$(x,y) \in WInd_{\mathscr{K}}(B)$, if and only if $f(x,a) = f(y,a)$ for some $a \in B$..

On the other hand, one can capture the notion of 'sameness' between two objects by incorporating a notion of similarity.

For instance, in [**?**] a notion of similarity, denoted as $Sim_{\mathscr{K}}(B)$, is defined as follows.

$(x,y) \in Sim_{\mathscr{K}}(B)$, if and only if $f(x,a) \cap f(y,a) \neq \emptyset$, for all $a \in B$.

The above notion is called the *strong similarity relation*. If in the above definition of $Sim_{\mathscr{K}}(B)$ we replace 'for all $a \in B$' by 'some $a \in B$' we would have a notion of *weak similarity*.

Based on the set inclusion relation, there can be other notions of indistinguishability such as *forward inclusion relation*, *backward inclusion relation* etc []. For example, the forward inclusion relation, denoted as $\subseteq_{\mathscr{K}}(B)$, is defined as follows.

$(x,y) \in \subseteq_{\mathscr{K}}(B)$, if and only if $f(x,a) \subseteq f(y,a)$, for all $a \in B$.

In case of backward inclusion relation, instead of $f(x,a) \subseteq f(y,a)$, for all $a \in B$ the condition would be $f(y,a) \subseteq f(x,a)$, for all $a \in B$. Both of these relations can be replaced by their respective weak versions by considering 'for some $a \in B$' instead of 'for all $a \in B$'.

In [], based on the set theoretic complementation operation another notion of indistinguishability is defined. The relation is named as *incomplementarity relation*. Let us denote the relation as $Incom_{\mathscr{K}}(B)$.

$(x,y) \in Incom_{\mathscr{K}}(B)$, if and only if $f(x,a) \neq f(y,a)^c$, for all $a \in B$.

We will not go into the technical details of how these different notions of indistinguishability contribute in the further studies of clustering the universe of objects and finding general decision rules about dependencies among different sets of attributes and decision attributes. However, it can be noted that compare to the indiscernibility relation generated from a DIS, the above notions of indistinguishability are not necessarily equivalence relations. For instance, $Sim_{\mathscr{K}}(B)$ is a reflexive, symmetric relation, $\subseteq_{\mathscr{K}} (B)$ is a reflexive, transitive relation, and $Incom_{\mathscr{K}}(B)$ is a symmetric relation. So, clearly in the context of NDIS we would have many different flexible ways of clustering objects based on their properties.

An immediate point to be observed here that in the above example car1 and car2 are similar in the sense of $Sim_{\mathscr{K}}(B)$ with respect to the attributes Mileage and Size, even though they are not indiscernible in the classical sense. On the other hand, Cars5 and Car6 both can be regarded as, in some sense, special cases of car2 as $(car5, car2), (car6, car2) \in \subseteq_{\mathscr{K}} (B)$; thus they are in some sense indistinguishable as well.

3.3 Incomplete information system

It may sometimes be the case that for some objects from the universe there is no information available with respect to some of the attributes. To represent such a situation, usually a distinguished value $*$ is taken as an attribute value so that for an attribute a, $f(x,a) = *$ signifies that we do not have information about the object with respect to the attribute a. The incompleteness in the information system is addressed by the following definitions.

Definition 3.5. A tuple $\mathscr{S} := (U, C, \{Val_a\}_{a\in C} \cup \{*\}, f)$ is called an *information system* (IS), where

- U, C, Val_a are as in Definition 3.1 and $* \notin \bigcup_{a\in C} Val_a$;
- $f : U \times C \to \bigcup\{Val_a : a \in C\} \cup \{*\}$ such that $f(x,a) \in Val_a \cup \{*\}$.

An information system which satisfies $f(x,a) = *$ for some $x \in U$ and $a \in C$ is called an *incomplete information system* (IIS).

Here to be noted that a deterministic information system can be considered as a special case of an IIS $\mathscr{S} := (U, C, \{Val_a\}_{a\in C} \cup \{*\}, f)$, where $f(x,a) \neq *$ for all $x \in U$ and $a \in C$.

Below an example of an incomplete information system is presented.

Car	Price	Mileage	Size
o_1	High	High	Full
o_2	Low	$*$	Full
o_3	$*$	$*$	Compact
o_4	High	$*$	Full
o_5	$*$	$*$	Full
o_6	Low	High	Full

Let us now focus on the notion of indiscernibility relation in the context of an IIS. If we keep the same definition as before, i.e., for any two elements $x, y \in U$, $(x, y) \in Ind_{\mathscr{S}}(B)$ iff $f(x,a) = f(y,a)$ for all $a \in B$, then question arises how do we interpret the case that at least one of $f(x,a), f(y,a)$ for some a can be unknown. For example, let us consider the objects o_1 and o_4 in the above table. Both the objects assume the same values with respect to the attributes Price and Size; whereas with respect to the attribute Mileage o_1 receives the value High and that of o_4 is unknown. That this value is not known can have different interpretations. It can be the case that the value for Mileage is *missing* for some reason, but it is possible to acquire this value. So, in that context $f(o_4, Mileage) = *$ can have the potential value 'High'. So, with respect to the earlier definition of $Ind_{\mathscr{S}}(B)$ though $(o_1, o_4) \notin Ind_{\mathscr{S}}(B)$, considering $B = \{Price, Mileage, Size\}$, in reality they can be indistinguishable.

Now let us consider another scenario by adding an attribute GPS (Global Positioning System) in the set of attributes. Now there can be one car, say o_1, with such a facility of GPS and another, say o_4, without that facility. Now if the possible values for GPS are 'Advanced' and 'Poor', then for the second car $*$ represents *absence* of a possibility to acquire this value. Thus, as in the context of *missing* value even if we may agree to consider o_1 and o_4 as indistinguishable assuming that o_4 may have the potential of having high mileage, in the context of *absence* of GPS we would not agree to that interpretation. So, clearly there can be different interpretations of $*$, and consequently different notions of indistinguishability.

These different interpretations of unknown values, from the perspectives of *missing* and *absence*, are discussed in [**?**, **?**, **?**, **?**], and based on that different notions of indistinguisability relations are defined. Below we present an brief overview of different notions of indistinguishability.

Considering the interpretation of *missing value* in [**?**, **?**], a notion of similarity relation is defined.

Definition 3.6. Given an information system $\mathscr{S} := (U, C, \{Val_a\}_{a \in C} \cup \{*\}, f)$ for any $B \subseteq C$, a binary relation $Tol_{\mathscr{S}}(B)$ over U is defined as follows. $(x, y) \in Tol_{\mathscr{S}}(B)$ if and only if , $f(x,a) = f(y,a)$ or $f(x,a) = *$, or $f(y,a) = *$, for all $a \in B$.

Clearly, $Tol_{\mathscr{S}}(B)$, called as a *tolerance relation* [**?**], is a reflexive, symmetric, non-transitive relation. Based on this tolerance relation a notion of tolerance class is defined. That is, given any $x \in U$, $I_B(x) = \{y \in U : (x, y) \in Tol_{\mathscr{S}}(B)\}$ is called the tolerance class of x with respect to the set of attributes B. This notion of tolerance class has been used as a basis for the notions parallel to the classical notions of lower and upper approximations.

Definition 3.7. Given $X \subseteq U$, the lower and upper approximations of X with respect to $Tol_{\mathscr{S}}(B)$, denoted as $\underline{X}_{Tol_{\mathscr{S}}(B)}$ and $\overline{X}_{Tol_{\mathscr{S}}(B)}$ respectively, are defined as follows.

$$\underline{X}_{Tol_{\mathscr{S}}(B)} := \{x \in U : I_B(x) \subseteq X\},$$
$$\overline{X}_{Tol_{\mathscr{S}}(B)} := \{x \in U : I_B(x) \cap X \neq \emptyset\}.$$

So, it is easy to notice that based on the notion of tolerance relations $\{o_1, o_4, o_5\}$ are indistinguishable. Here to be noted that for o_5 even though most of the values are missing, o_5 is regarded as similar to o_1 and o_4. This can be counter-intuitive in some practical contexts.

Let us now concentrate on another interpretation of $*$ representing *absence* of a property characterized by a particular attribute or impossibility of obtaining value for such an attribute. As mentioned in [**?**], "Under such a perspective each object may have a more or less complete description, depending on how many attributes has been possible to apply." From this perspective x can be considered similar to y only if they have the same known values.

Definition 3.8. Given an information system $\mathscr{S} := (U, C, \{Val_a\}_{a \in C} \cup \{*\}, f)$ for any $B \subseteq C$, a binary relation $Sim_{\mathscr{S}}(B)$ over U is defined as follows. $(x,y) \in Sim_{\mathscr{S}}(B)$ if and only if , $f(x,a) \neq *$ and $f(x,a) = f(y,a)$, for all $a \in B$.

As it is clear from the context, we prefer to use the same notation for similarity relation as it used in the context of a DIS. It is easy to observe that here $Sim_{\mathscr{S}}(B)$ is a reflexive, transitive, but non-symmetric relation. In fact, this is a partial order relation. Moreover, based on the above notion we can interpret similarity from two perspectives. For any object $x \in U$ we can construct following two sets.

$R(x) = \{y \in U : (y,x) \in Sim_{\mathscr{S}}(B)\}$ - the set of objects similar to x
$R^{-1}(x) = \{y \in U : (x,y) \in Sim_{\mathscr{S}}(B)\}$ - set of objects to which x is similar.

Now based on above two sets the lower and upper approximations of a set are defined. The intuition is that an object x is considered to be surely belonging to X if all objects similar to x belong to X. On the other hand all objects which are similar to an object in X, are considered to be potentially belonging to X.

Definition 3.9. Given $X \subseteq U$, the lower and upper approximations of X with respect to $Sim_{\mathscr{S}}(B)$, denoted as $\underline{X}_{Sim_{\mathscr{S}}(B)}$ and $\overline{X}_{Sim_{\mathscr{S}}(B)}$ respectively, are defined as follows.

$$\underline{X}_{Sim_{\mathscr{S}}(B)} := \{x \in U : R^{-1}(x) \subseteq X\},$$
$$\overline{X}_{Sim_{\mathscr{S}}(B)} := \cup\{R(x) : x \in X\}.$$

In [**?**], authors showed that given an information system $\mathscr{S}$ and a set X of objects, the upper and lower approximations of X obtained by $Sim_{\mathscr{S}}(B)$ are respectively the refinements of the ones obtained by $Tol_{\mathscr{S}}(B)$. That is, for any $X \subseteq U$ and set of attributes B, $\underline{X}_{Tol_{\mathscr{S}}(B)} \subseteq \underline{X}_{Sim_{\mathscr{S}}(B)}$ and $\overline{X}_{Sim_{\mathscr{S}}(B)} \subseteq \overline{X}_{Tol_{\mathscr{S}}(B)}$.

There is another way to design a notion of similarity in the context of an IIS. The approach proposes to consider all possible conversions of an incomplete information system to a complete information system, and then the notion of similarity between two objects is defined based on their values in all of those DIS, which are converted from the original IIS.

Definition 3.10. Let $\mathscr{S} := (U, C, \bigcup_{a \in C} Val_a \cup \{*\}, f)$ be an IIS. A deterministic information system $\mathscr{S}' := (U, C, \bigcup_{a \in C} Val_a \cup \{*\}, f')$ is said to be a *completation* of $\mathscr{S}$ if $f(x,a) \neq *$ implies $f'(x,a) = f(x,a)$ for all $a \in C$ and $x \in U$.

Thus, given an IIS, there can be different DIS's such that the already known values from the original IIS remains the same in each completation of the IIS. Now a new notion of similarity, denoted as $Sim^{com}_{\mathscr{S}}(B)$, can be defined as follows.

Definition 3.11. $(x,y) \notin Sim^{com}_{\mathscr{S}}(B)$ if and only if $(x,y) \notin Ind_{\mathscr{S}'}(B)$ for all completations $\mathscr{S}'$ of $\mathscr{S}$.

Thus two objects are distinguishable with respect to $Sim^{com}_{\mathscr{S}}(B)$ if and only if these are distinguishable with respect to the classical indiscernibility relation $Ind_{\mathscr{S}'}(B)$ in all the deterministic information systems obtained by assigning any value from V_a to an object with missing value for an attribute a. In other words, if two objects are similar in the sense of $Sim^{com}_{\mathscr{S}}(B)$, then there is a possibility that they are indistinguishable with respect to some of the completation of the original IIS. Of course, in the case of deterministic information systems, indiscernibility relation and similarity relation would coincide.

Based on the above notion of similarity a notion of similarity class can be obtained as follows. Given an IIS $\mathscr{S}$ the similarity class of an object $x \in U$ with respect to a set of attributes B is given by $Sim^{com}{}_{\mathscr{S}}(B)(x) := \{y \in U : (x,y) \in Sim^{com}{}_{\mathscr{S}}(B)\}$.

Definition 3.12. The lower and upper approximations of X with respect to similarity relation $Sim^{com}_{\mathscr{S}}(B)$, denoted by $\underline{X}_{Sim^{com}_{\mathscr{S}}(B)}$ and $\overline{X}_{Sim^{com}_{\mathscr{S}}(B)}$, are defined as follows:

$$\underline{X}_{Sim^{com}{}_{\mathscr{S}}(B)} := \{x \in U : Sim^{com}_{\mathscr{S}}(B)(x) \subseteq X\},$$
$$\overline{X}_{Sim^{com}_{\mathscr{S}}(B)} := \{x \in U : Sim^{com}_{\mathscr{S}}(B)(x) \cap X \neq \emptyset\}.$$

Using Definition 3.11, it is not difficult to obtain the following result.

Proposition 3.2.

(i) $x \in \underline{X}_{Sim^{com}{}_{\mathscr{S}}(B)}$ *if and only if* $x \in \underline{X}_{Ind_{\mathscr{S}'}(B)}$ *for all completations* $\mathscr{S}'$ *of* $\mathscr{S}$.
(ii) $x \in \overline{X}_{Sim^{com}{}_{\mathscr{S}}(B)}$ *if and only if* $x \in \overline{X}_{Ind_{\mathscr{S}'}(B)}$ *for some completation* $\mathscr{S}'$ *of* $\mathscr{S}$.

3.4 Dynamic information system

We already know that the information signature of an object in an information system is relative to a set of attributes or parameters. If we change the set of parameters, then the same object may have different information signature. So, it is quite clear that the information about an object depends on the window of specification through which we observe it. Information signature of an object with respect to an attribute may change based on time. On the other hand, a previously considered attribute may become irrelevant at a further point of time to define a concept or to classify an object as an instance of the concept. An information system without the time parameter is basically static in nature. In this section we present dynamic information system as an extension of the notion of information system.

In [**?**], Orłowska has added a temporal dimension to the study of information systems. The notion of an information system is extended by adding a set T of time points with a linear order $<$ on T.

Definition 3.13. A tuple $\mathscr{DS} := (U, C, \{Val_a\}_{a\in C}, T, <, f)$ is called a *dynamic information system*, where

- U, C, Val_a are as in Definition 3.1;
- T is a non-empty set of *time points*;
- $<$ is a linear order on T;
- $f : U \times T \times C \to \cup\ \{Val_a :\ a \in C\}$ is such that $f(x,t,a) \in Val_a$, for any $x \in U$, $t \in T,\ a \in C$.

So, contrary to an information system, in case of a dynamic information system, the value that f assigns to an object x for any attribute a, becomes dependent on the chosen time point t. Let us note that in Definition 3.13 the attribute set C does not vary with time. Let us now consider an information systems containing personal data, such as age, address, education etc. In such an information system for some particular objects the values corresponding to some attributes may change with time. Moreover, with time there can be a need to add some relevant attributes or delete some unnecessary attributes as well. These aspects are considered in [**?**].

Definition 3.14. A dynamic information system is a family of quadruples $\mathscr{DS}_W$: $= \{(U_t, C_t, V_t, f_t)\}_{t\in T}$ where

- T is a discrete set of time points, denoted by $0, 1, 2 \ldots N$,
- U_t is the set of objects at the time point t,
- C_t is the set of attributes at the moment t,
- $V_t = \cup_{a\in C_t} V_{t_a}$ where V_{t_a} denotes the value set for the attribute a at time t,
- $f_t : U_t \times C_t$ is a function assigning value from V_t to each pair (x, a) from $U_t \times \mathscr{A}_t$.

It is to be noted that according to Definition 3.13 for different time points a given set of objects U with respect to a fixed set of attributes C may have different values. On the other hand, according to Definition 3.14 with respect to two time points $t_1 \neq t_2$ the respective sets of attributes C_{t_1} and C_{t_2} and the respective sets of objects U_{t_1}, U_{t_2} may be different. Based on Definition 3.14 the notions of *time invariant dynamic system* and *time varying dynamic information system* can be introduced.

A dynamic information system $\mathscr{DS}_W$ is a time invariant dynamic information system if the following two conditions hold.

(i) $Z_T = \cap_{t\in T} D_t \neq \phi$ where $D_t = U_t \times C_t$ and

(ii) for all $t,t' \in T$ and $(x,a) \in Z_T$, $f_t(x,a) = f_{t'}(x,a)$.

On the other hand, a dynamic information system is a time varying dynamic system if the following two conditions hold.

(i) $Z_T = \cap_{t \in T} D_t = \phi$ or
(ii) $Z_T \neq \phi$ and for some $(x,a) \in Z_T$ $f_t(x,a) \neq f_{t'}(x,a)$.

In [**?**], a dynamic information system based on *time sequence* (finite set of time points) is defined, and four different indiscernibility relations are defined based on that. For a universe of objects U, a set of attributes C and a time sequence ΔT_i containing time points $\{t_{i1}, t_{i2}, \ldots, t_{ik}\}$ they defined value of an object $o \in U$ for an attribute $a \in C$ over the time sequence ΔT_i. Corresponding to the time sequence ΔT_i they defined a value transition sequence for an object o with respect to an attribute a, which can be denoted as $\{a^{t_{i1}}(o), a^{t_{i2}}(o), \ldots, a^{t_{ik}}(o)\}$. So, contrary to the value assigning functions introduced in Definitions 3.13 and 3.14, here we have $f(o,a,\Delta T_i)$, on which using a projection function $\Pi_j(f(o,a,\Delta T_i))$ one can obtain $a^{t_{ij}}(o)$, the value of the object o with respect to the attribute a at the time point t_{ij} from the time sequence ΔT_i.

Now based on such a notion of dynamic information system two objects $o,o' \in U$ can be indiscernible from different perspectives.

- Two objects o,o' are indiscernible with respect to a set of attributes B over a time sequence ΔT_i if for all $a \in B$ the value transition sequence of o and o' are the same over the time sequence ΔT_i, that is $\{a^{t_{i1}}(o), a^{t_{i2}}(o), \ldots, a^{t_{ik}}(o)\} = \{a^{t_{i1}}(o'), a^{t_{i2}}(o'), \ldots, a^{t_{ik}}(o')\}$.
- Two objects o,o' are indiscernible with respect to a set of attributes B at some time point t_{ij} of a time sequence ΔT_i if for all $a \in B$ $a^{t_{ij}}(o) = a^{t_{ij}}(o')$.
- Two objects o,o' are indiscernible with respect to a set of attributes B over a transition period $\langle t_{ij}, t_{il} \rangle$ from a time sequence ΔT_i if for all $a \in B$ the value transition of o at the time point t_{ij} to the time point t_{il} are the same as the object o'. That is, $a^{t_{ik}}(o) = a^{t_{ik}}(o')$ for $j \leq k \leq l$.
- Two objects o,o' are indiscernible with respect to a set of attributes B from a certain time point t_{ij} of a time sequence ΔT_i if for all $a \in B$ $a^{t_{il}}(o) = a^{t_{il}}(o')$ for all $t_{il} \in \Delta T_i$ such that $l \geq j$.

3.5 Other information systems

IS with relation structure on attribute-value set, fuzzy values, value set associated with types.

3.6 Fields of research related to information systems

After having a brief idea about different types of information systems, let us now present a list of different disciplines of research that are significant from different contexts of information synthesis, and are grounded on the notion of information system.

(i) Decision systems
(ii) Dependency of attributes
(iii) Boolean reasoning
(iv) Decision rules and accuracy measure
(v) Decision functions dealing with inconsistency in decision
(vi) Data reduction and decision reducts
(vii) Quality and complexity of reducts
(viii) Description length principle
(ix) Classifier induction
(x) Network of information systems
(xi) Aggregation of information systems
(xii) Deductive logics and corresponding algebras

A brief description of each of the above mentioned issues is given below. A more detailed discussion on some of the issues is also planned in the further chapters.

Often, in an information system $\mathbb{A} := (U, A)$, we distinguish a partition of A into two disjoint subsets $C, D \subseteq A$ of attributes, called *condition* and *decision* (*action*) attributes respectively. The tuple $\mathbb{A} = (U, C, D)$ is called a *decision system.* Any decision system $\mathbb{A}$ defines two partitions of the universe U; one is defined by the condition attributes from C and the second one is defined by the decision attributes from D. As the name suggests, decision systems provide the base for synthesizing data and abstracting relevant information from the perspective of decision making.

Thus, an important issue in data analysis is to discover dependencies between attributes in a given decision system $\mathbb{A} = (U, C, D)$. Intuitively, a set of attributes D depends totally on a set of attributes C, denoted by $C \Rightarrow D$, if the values of the attributes from C uniquely determine the values of the attributes from D. In other words, D depends totally on C, if there exists a functional dependency, as developed in the theory of relational databases [**?**], between values of C and D.

Here comes the next point, which is, how to represent dependency between a set of condition attributes and a set of decision attributes in a formal language and abstract out patterns or rules that help in decision making. Let us consider $V = \bigcup\{V_a \mid a \in C\} \bigcup\{V_d \mid d \in D\}$. An atomic formulae over $B \subseteq C \cup D$ and V are expressions of the form $a = v$, called *descriptors (selectors) over B and V*, where $a \in B$ and $v \in V_a$. The set of formulae over B and V, denoted by $\mathscr{F}(B,V)$, is the smallest set containing all atomic formulae over B and V and all formulae that are closed under the propositional connectives $\wedge$ (conjunction), $\vee$ (disjunction) and $\neg$ (negation). For any formulae $\varphi \in \mathscr{F}(B,V)$, by $\|\varphi\|_{\mathbb{A}}$ we denote the meaning of φ in the decision table $\mathbb{A}$ which is the set of all objects in U with the property φ. For example, if φ denotes the formula $a = v$, then $\|a = v\|_{\mathbb{A}} = \{x \in U \mid a(x) = v\}$.

The meaning of the compound formulae are defined as $\|\varphi \wedge \varphi'\|_{\mathbb{A}} = \|\varphi\|_{\mathbb{A}} \cap \|\varphi'\|_{\mathbb{A}}$, $\|\varphi \vee \varphi'\|_{\mathbb{A}} = \|\varphi\|_{\mathbb{A}} \cup \|\varphi'\|_{\mathbb{A}}$, and $\|\neg\varphi\|_{\mathbb{A}} = U - \|\varphi\|_{\mathbb{A}}$. The formulae from $\mathscr{F}(C,V)$ and $\mathscr{F}(D,V)$ are called *condition formulae* and *decision formulae of* $\mathbb{A}$ respectively. For any decision attribute $d \in D$ and $x \in U$, the value $d(x) \in V_d$. So, any object x belongs to a *decision class*, namely $\| \bigwedge_{d\in D} d = d(x)\|_{\mathbb{A}}$ of $\mathbb{A}$, and all decision classes of $\mathbb{A}$ create a partition of U, denoted as U/D.

A *decision rule* for $\mathbb{A}$ is any expression of the form $\varphi \to \psi$, where $\varphi \in \mathscr{F}(C,V)$, $\psi \in \mathscr{F}(D,V)$, and $\|\varphi\|_{\mathbb{A}} \neq \varnothing$. Formulae φ and ψ are referred to as the *predecessor* and the *successor* of the decision rule $\varphi \to \psi$. Decision rules are often presented in the form of "*IF* ... *THEN* ..." statements, and are widely used in different machine learning [**?**] techniques. Decision rule $\varphi \to \psi$ is *true* in $\mathbb{A}$ if and only if $\|\varphi\|_{\mathbb{A}} \subseteq \|\psi\|_{\mathbb{A}}$. Otherwise, one can measure its *truth degree* by introducing some inclusion measure of $\|\varphi\|_{\mathbb{A}}$ in $\|\psi\|_{\mathbb{A}}$.

Let us denote by $|\varphi|$ the number of objects from U that satisfies formula φ, *i.e.*, the cardinality of $\|\varphi\|_{\mathbb{A}}$. According to Łukasiewicz [**?**], one can assign the value $\dfrac{|\varphi|}{|U|}$ to formula φ, and the fractional value $\dfrac{|\varphi \wedge \psi|}{|\varphi|}$ to the implication $\varphi \to \psi$, under the assumption that $\|\varphi\| \neq \varnothing$. Much later the above proposed definition of fractional value by Łukasiewicz, was adapted by the machine learning and data mining communities in different notions such as in the definitions of accuracy of decision rules and confidence of association rules.

Usually, given $C, D \subseteq A$ each object x of a decision system determines a *decision rule* of the form as presented below.

$$\bigwedge_{a\in C} a = a(x) \to \bigwedge_{d\in D} d = d(x). \tag{3.1}$$

Now referring to $C \Rightarrow D$, mentioned above, we can say $C \Rightarrow D$ if and only if the rule (3.1) is true on $\mathbb{A}$ for every $x \in U$.

It can be the case that for each object $x \in U$, the combination of values for the expression $\bigwedge_{a\in C} a = a(x) \to \bigwedge_{d\in D} d = d(x)$ appears exactly once in the table. In that case, corresponding to every object the respective decision rule is different. Another possibility is that the equivalence classes of $Ind(C)$ are completely included in the equivalence classes of $Ind(D)$; that is, all objects y having the same description of the predecessor $\bigwedge_{a\in C} a = a(x)$ as the object x has, also have $\bigwedge_{d\in D} d = d(x)$ as the successor. Such kinds of decision system are consistent decision systems.

In practice, often there are such situations where for two objects x, x' the respective condition formulae $\bigwedge_{a\in C} a = a(x)$ and $\bigwedge_{a\in C} a = a(x')$ are the same but the respective decision formulae $\bigwedge_{d\in D} d = d(x)$ and $\bigwedge_{d\in D} d = d(x')$ are different. In such contexts $C \Rightarrow D$ does not hold. Such decision systems are called inconsistent decision systems, and instead of classical functional dependency between C and D there can be partial dependency between the sets of conditional attributes and decision attributes. In this regard, to measure the dependency between C, the set of conditional attributes and D, the set of decision attributes there are different possible definitions [**?**].

For instance, we say that D *depends on* C to a *degree* k $(0 \leq k \leq 1)$, denoted by $C \Rightarrow_k D$, if

$$k = \gamma(C,D) = \frac{|POS_C(D)|}{|U|}, \tag{3.2}$$

where

$$POS_C(D) = \bigcup_{X \in U/D} \mathsf{LOW}_C(X) = \bigcup_{u \in U} \{[u]_C : [u]_C \subseteq [u]_D\}, \tag{3.3}$$

called the *positive region* of the partition U/D with respect to C. That is $POS_C(D)$ is the set of all elements of U that can be uniquely classified as the blocks of the partition U/D, by means of C. If $k = 1$ we say that D *depends totally* on C, and if $k < 1$, we say that D *depends partially* (to *degree* k) on C. If $k = 0$ then the *positive region* of the partition U/D with respect to C is empty. The coefficient k expresses the ratio of all elements of the universe, which can be properly classified to blocks of the partition U/D, employing attributes of C and is called the *degree of the dependency.*

In the above definition of the positive region of U/D with respect to the set of condition attributes C, one can observe that the positive region considers those equivalence classes in which objects with the same description belong to the same decision class. Thus, while computing the dependency measure between the set of condition attributes C and the set of decision attributes D using equation 3.2 the objects, for which inconsistency in decisions arise, are ignored. This reflects one way of defining a decision function which focuses only on the consistent part of an inconsistent decision table. In literature [], there are several other interesting and intuitive approaches for defining such decision functions.

In the context of finding dependecy between the set C of conditional attributes and the set D of decision attributes, one important issue is to check if there exist some attributes in C which are redundant for classifying all possible vectors of values for D present in $\mathbb{A}$. Let us express this idea more precisely.

For a $C' \subseteq C$ we say C' is a *D-reduct* (reduct with *respect* to D) of C, if C' is a minimal subset of C such that

$$\gamma(C,D) = \gamma(C',D). \tag{3.4}$$

The intersection of all D-reducts is called a *D-core* (core with *respect* to D). As the core is the intersection of all reducts, in a sense, the core is the most important subset of attributes, since none of its elements can be removed without affecting the classification power of the attributes. Certainly, the reducts can be of more compound in nature. For example, the core can be empty but there can exist a partition of reducts into a few sets with non empty intersection. Many other kinds of reducts and their approximations are available in the literature [**?**, **?**, **?**, **?**, **?**, **?**, **?**, **?**], and these issues are discussed in Chapter 4.

Different notions of decision reduct basically are developed based on different conditions for dependency between a set of condition attributes and a set of decision attributes. Thus, in different kinds of reducts information encoded in original data table is preserved from different perspectives as well as to different degrees. Reducts are used to build the models or decision functions for the original data, and choosing a particular reduct or a set of reducts has impact on the size as well as the quality of the model in describing a given data set. Here one important issue is, how a particular decision function deals with the inconsistent cases appeared in a decision system. Then a particular notion of decision reduct is defined based on that notion of decision function. For example, a decision function, known as *generalized decision function*, considers all possible decision values occuring in a single equivalence class as equally likely. On the other, the decision function, known as *rough membership function*, prefers to consider all possible decision values with their respective probabilities of occurring in a particular equivalence class. So, obviously the question of quality and complexity of a decision reduct comes. These quality and complexity measures can be from different perspectives. The concern can be on preserving the information of the original decision table as much as possible or on characterizing efficiently a new test case without significant loss of data or on simplifying the decision function so that it leads to an easy but significantly correct characterization of a new test case.

For example, according to [**?**] reducts should preserve the distance between the vectors of attribute values for any two objects, provided the distance is greater than a given threshold; in [**?**, **?**], the distance between entropy distributions between any two objects matters while defining a reduct, and in [**?**], the reducts are defined relative to objects, that are used for generation of decision rules. A discussion concerning relationships among different kinds of reducts is presented in Chapter 4.

Classification problem is another important branch that has emerged with the aim of classifying an imprecise concept by a cluster of prototypical examples, which are already described with respect to a set of attributes. It is worthwhile mentioning that randomly generated reduct cannot be relevant for inducing a good classifier. The size together with the quality are two basic components, that are often tuned in selecting a relevant data reduct and thus a good model for the data set is obtained. It turns out that the different kinds of reducts can be efficiently computed by using heuristics on the Boolean reasoning approach [**?**].

Let us here briefly describe the minimum description length principle [**?**, **?**, **?**, **?**] as one of the technique to tune the size of a model. For a given decision table, there can be different decsion reducts, which are not comparable to each other with respect to the set inclusion relation. So, each decision reduct can generate a set of decision rules for the given decision table. The minimum description length principle allows to select the decision reduct with smallest number of elements and thus the antecedent parts of the corresponding decision rules have minimum length.

The discussion in the above paragraphs concerns about different aspects of information synthesis of a given information system or a decision system. In the context of multi-agent environment there can be a network of agents and each of them can be represented by the respective decision system. So, here too the question of aggregation of information comes. In the literature [], different techniques of aggregation of information from different databases are present. In Chapter **??** we attempt to present a brief summary in this regard.

Combining the issues related to some of the above topics, in Chapter 4 we present different aspects of decision systems keeping a special focus on data reduction and its related issues.

References

1. Bazan, J.G.: A comparison of dynamic and non-dynamic rough set methods for extracting laws from decision tables. In: L. Polkowski, A. Skowron (eds.) Rough Sets in Knowledge Discovery 1: Methodology and Applications, *Studies in Fuzziness and Soft Computing*, vol. 18, pp. 321–365. Physica-Verlag, Heidelberg (1998)
2. Brown, F.: Boolean Reasoning. Kluwer Academic Publishers, Dordrecht (1990)
3. Garcia-Molina, H., Ullman, J., Widom, J.: Database Systems: The Complete Book. Prentice Hall, Upper Saddle River, New Jersey (2002)
4. Grzymała-Busse, J.W., Rzasa, W.: Local and global approximations for incomplete data. In: Transactions on Rough Sets VIII: Journal Subline, pp. 21–34
5. Hastie, T., Tibshirani, R., Friedman, J.H.: The Elements of Statistical Learning: Data Mining, Inference, and Prediction. Springer-Verlag, Heidelberg (2001)
6. He, X., Xu, L., Shen, W.: Dynamic information system and its rough set model based on time sequence. In: Y.Q. Zhang, T.Y. Lin (eds.) Proceedings of the IEEE International Conference on Granular Computing, Atlanta, May 10-12, 2006, pp. 542–545. IEEE Press (2006)
7. Kryszkiewicz, M.: Rough set approach to incomplete information systems. Information Sciences **112**(1-4), 39–49 (1998)
8. Kryszkiewicz, M.: Rules in incomplete information systems. Information Sciences **113**, 271–292 (1999)
9. Łukasiewicz, J.: Die logischen Grundlagen der Wahrscheinlichkeitsrechnung, 1913. In: L. Borkowski (ed.) Jan Łukasiewicz - Selected Works. North Holland Publishing Company, Amsterdam, London, Polish Scientific Publishers, Warsaw, 1970, pp. 16-63 (1913)
10. Nguyen, H.S.: Approximate boolean reasoning: Foundations and applications in data mining. In: J.F. Peters, A. Skowron (eds.) Transactions on Rough Sets **V**: Journal Subline, *Lecture Notes in Computer Science*, vol. 4100, pp. 344–523. Springer, Heidelberg (2006)
11. Nguyen, H.S., Nguyen, S.H.: Rough sets and association rule generation. Fundamenta Informaticae **40**(4), 383–405 (1999)
12. Nguyen, H.S., Ślezak, D.: Approximate reducts and association rules - correspondence and complexity results. In: Proceedings of the 7-th International Workshop on Rough Sets, Fuzzy Sets, Data Mining, and Granular-Soft Computing (RSFDGrC'1999), Yamaguchi, November 9-11, 1999, pp. 137–145 (1999)
13. Orłowska, E.: Dynamic information system. Fundamenta Informaticae **5**, 101–118 (1982)
14. Orłowska, E.: Many-valuedness and uncertainty. International Journal of Multiple Valued Logic **4**, 207–227 (1999)
15. Orłowska, E., Pawlak, Z.: Representation of nondeterministic information. Theoretical Computer Science **29**, 27–39 (1984)
16. Pawlak, Z.: Information systems - theoretical foundations. Information Systems **6**, 205–218 (1981)
17. Rissanen, J.: Modeling by shortes data description. Automatica **14**, 465–471 (1978)
18. Rissanen, J.: Minimum-description-length principle. In: S. Kotz, N. Johnson (eds.) Encyclopedia of Statistical Sciences, pp. 523–527. John Wiley & Sons, New York, NY (1985)
19. Skowron, A.: Extracting laws from decision tables. Computational Intelligence: An International Journal **11**, 371–388 (1995)

20. Ślezak, D.: Various approaches to reasoning with frequency-based decision reducts: A survey. In: Rough Set Methods and Applications: New Developments in Knowledge Discovery in Information Systems, pp. 235–285
21. Ślezak, D.: Approximate reducts in decision tables. In: Sixth International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems IPMU'1996, vol. III, pp. 1159–1164. Granada, Spain (1996)
22. Ślezak, D.: Normalized decision functions and measures for inconsistent decision tables analysis. Fundamenta Informaticae **44**, 291–319 (2000)
23. Ślezak, D.: Approximate entropy reducts. Fundamenta Informaticae **53**, 365–387 (2002)
24. Stefanowski, Tsoukiàs, A.: On the extension of rough sets under incomplete information. In: N. Zhong, A. Skowron, S. Ohsuga (eds.) Proceedings of the 7th International Workshop on New Directions in Rough Sets, Data Mining, and Granular-Soft Computing (RSFDGrC '99), Yamaguchi, Japan, November 9-11, 1999

Chapter 4
Data Reduction and Information Synthesis

Information synthesis is the central theme of different fields of research, among which big data, data science, relational databases, information systems are a few to name. In dealing with huge data one needs to consider how effectively the information scattered in the data can be accummulated and can be represented in an user-friendly language. Data reduction in a way, that does not let significant information to lose, pertains to the first aspect. The second aspect leads to the generation of decision rules abstracted out from the minimal data that preserves all relevant information about the original data set. In this chapter, we concentrate on three main aspects of information synthesis from the perspective of decision making. One is decision reduct. The other two are respectively the decision rules and the decision function which are defined based on a particular decision reduct. In contrast to Chapter 3, here for simplicity we consider a decision system $\mathbb{A}$ with a single decision attribute d. The ideas presented below can be lifted to the case for a finite set D as well.

4.1 Decsion reducts

In Chapter 3, we already have presented the basic idea behind the notion of decision reduct. This corresponds to an attribute reduction criterion that must preserve the relevant information of all the objects with respect to the decision attribute. Elimination of redundant attributes is one of the important steps in feature selection algorithms [**?**], which is often used in machine learning techniques. In the statistical learning community also the idea of feature selection is used and is known as *subset selection* [**?**]. In this section we shall first present the classical notion of decision reduct, and then present different other generalized definitions of decision reduct that are developed in order to take care of the inconsistent decision systems.

The classical attribute reduction criterion is as follows.

Definition 4.1. Given $\mathbb{A} = (U, C \cup \{d\})$, a subset $B \subseteq C$ is said to be a decision superreduct, if and only if $Ind(B) \subseteq Ind(\{d\})$, and if there is no proper subset of B satisfying the above condition, then B is called a decision reduct.

	a_1	a_2	a_3	d
o_1	average	close	moderate	high
o_2	average	close	moderate	high
o_3	average	close	moderate	high
o_4	more than average	far	high	moderate
o_5	more than average	far	high	low
o_6	more than average	far	low	low
o_7	average	close	moderate	high
o_8	more than average	far	low	low
o_9	more than average	far	low	high

Table 4.1 Example of decision table $(U, A \cup \{d\})$, where $U = \{o_1, \ldots, o_9\}$ and $A = \{a_1, a_2, a_3\}$.

In the context of inconsistent decision systems searching for such a subset of attributes, preserving all relevant information about the decision classes, is not that straightforward as $Ind(C) \not\subseteq Ind(d)$ and so there is no such subset B of C for which the above mentioned inclusion holds. For example let us consider the decision system presented in Table 4.1. It can be noticed that objects o_4, o_5 have the same descriptions with respect to the set of attributes $\{a_1, a_2, a_3\}$ but they belong to different decision classes. On the other hand, o_6, o_7, o_8 belong to the same equivalence class with respect to the set of attributes $\{a_1, a_2, a_3\}$ but they differ in their decision values. So, for no subset B of $\{a_1, a_2, a_3\}$, $Ind(B) \subseteq Ind(d)$. So, in such contexts the classical notion of decision reduct is approximated and generalized from different perspectives. Below we present a few such prevalent notions of decision reduct from the literature.

In Chapter 3, in Equation 3.3 the notion of positive region is defined with respect to C, the whole set of condition attributes. Let us here present the definition of the positive region for any B ($\subseteq C$).

Definition 4.2. Given $\mathbb{A} = (U, C, d)$ and $B \subseteq C$, the positive region for U/d induced by B is defined as $POS_B(d) = \{u \in U : \forall_{u' \in [u]_B} d(u) = d(u')\}$ or, equivalently, a set-theoretic sum of the lower approximations of all the decision classes $D_i = \{u \in U : d(u) = v_i\}$, where $i \in \{1, \ldots, r\} = V_d$, the value set for d.

Based on Definition 4.2 we can now present the corresponding notion of decision reduct as follows.

Definition 4.3. Given $\mathbb{A} = (U,\ C, d)$ and $B \subseteq C$ if B is the smallest set that satisfies $POS_B(d) = POS_C(d)$, B is said to be a POS-decision reduct.

So, the notion of decision reduct defined in Definition 4.3 imposes the condition that the set of indistinguishable objects with respect to the set of attributes C and having the same decision for d are the same as the set of objects which are indistinguishable with respect to B and have the same decision for d. More specifically, we know that as $Ind(C) \subseteq Ind(B)$, $[u]_C \subseteq [u]_B$. According to Definition 4.3, if $[u]_C \subseteq [u]_d$ then $[u]_B \subseteq [u]_d$. That is, in order to understand the features of different objects falling in different decision classes, it is enough to concentrate on the subset B of attributes.

In the definition of POS-decision reduct the key point is to focus only on the positive region of the decision classes with respect to the concerned set of condition attributes. If two objects u_1, u_2 are indistinguishable with respect to C, but have different decision values, then Definition 4.3 does not bother to check whether they are indistinguishable with respect to B. For example for Table 4.1 all the singleton subset $\{a_1\}$, $\{a_2\}$, $\{a_3\}$ are POS-decision reduct though none of them are sufficient to characterize the decision classes corresponding to the value 'moderate' and 'low'. So, the decision information of inconsistent objects are completely ignored while reducing the set of condition attributes. But often in real life situations, ignoring such information may lead to an improper modelling of the original data set.

In this regard, let us first present the notion of generalized decision function that preserves all information pertaining to different decisions encountered in a class $[u]_B$.

Definition 4.4. Given $\mathbb{A} = (U, C, d)$ and $B \subseteq C$, the generalized decision function induced by B is defined as $\partial_B(u) = \{d(u') : u' \in [u]_B\}$ for each $u \in U$.

Now, in contrast to POS-decision reduct, we present the following notion of decision reduct defined based on the generalized decision function.

Definition 4.5. Given $\mathbb{A} = (U, C, d)$ and $B \subseteq C$ if B is the smallest set that satisfies $\partial_B(u) = \partial_C(u)$, then B is said to be the ∂-decision reduct.

So, according to Definition 4.5 a subset B of C is considered to be a decision reduct if B is the smallest such subset of attributes that for any element $u \in U$, $[u]_B$ contains exactly those decision values that are encountered in the equivalence class $[u]_C$. Thus, compare to POS-decision reduct ∂-decision reduct does not throw away inconsistent cases from its consideration. For example, from Table 4.1 we can see that $\partial_{\{a_1,a_2,a_3\}}(o_i) = \{high\}$ for i = 1, 2, 3, 7, $\partial_{\{a_1,a_2,a_3\}}(o_i) = \{moderate, low\}$, for i = 4, 5 and $\partial_{\{a_1,a_2,a_3\}}(o_i) = \{low, high\}$ for i = 6, 8, 9. Now among different subsets of $\{a_1, a_2, a_3\}$ it can be observed that only $\{a_3\}$ is the ∂-decision reduct as for any o_i, $\partial_{\{a_1,a_2,a_3\}}(o_i) = \partial_{\{a_3\}}(o_i)$. So, though considering only the consistent part of the Table 4.1 all $\{a_1\}$, $\{a_2\}$, $\{a_3\}$ turn out to be POS-decision reducts, synthesizing information from the whole decision system $\{a_3\}$ is regarded as the only ∂-decision reduct.

It is to be noted that in the notion of ∂-decision reduct all decision values occurred in a particular equivalence class are considered as equally likely. However, in practice, while assigning a decision value to a newly appeared case, whose description matches to the description of a particular equivalence class, it is much more counter-intuitive to select a decision value with maximum frequency than to consider all possible decision values occurring in that particular equivalence class. In this regard, below we present the notion of *rough membership function*, which considers the probability distribution of the concerned equivalence class over the set of all decision values.

Definition 4.6. Given $(U, C \cup \{d\})$ and $B \subseteq C$, the rough membership function induced by B is defined as $\overrightarrow{\mu_B}(u) = \langle \mu_B^1(u), \mu_B^2(u), \ldots, \mu_B^r(u) \rangle$ where for each $u \in U$ and $i = 1, \ldots, r$, $\mu_B^i(u) = \frac{|\{u' \in [u]_B : d(u') = v_i\}|}{|[u]_B|}$.

The notion of decision reduct defined based on the notion of rough membership function is as follows.

Definition 4.7. Given $\mathbb{A} = (U,\ C,\ d)$ and $B \subseteq C$ if B is the smallest set that satisfies $\overrightarrow{\mu_B}(u) = \overrightarrow{\mu_C}(u)$, then B is said to be the μ-decision reduct.

That is, according to μ-decision reduct B is the desired subset of attributes if for any $u \in U$ the probability of any decision value in the equivalence class $[u]_B$ is the same as the probability of that decision value in the equivalence class $[u]_C$. Here to be noted, that compare to ∂-decision reduct the notion of μ-decision reduct also focuses on the frequency distribution of the decision values that are encountered in a particular equivalence class. Referring to Table 4.1, we can see that as ∂-decision reduct $\{a_3\}$ is also the only μ-decision reduct. However, $\{a_3\}$ as μ-decision reducts encodes more information than $\{a_3\}$ as ∂-decision reduct.

There are a few other notions of decision reducts which are defined based on the notions of lower and upper approximations of the decision classes. Below we present three such notions of decision reduct.

Definition 4.8. Given $\mathbb{A} = (U,\ C,\ d)$ and $B \subseteq C$, B is said to be a decision reduct if B is the smallest set such that

1. B generates the same upper approximation of each decision class as C.
2. B generates the same upper and lower approximations of each decision class as C.
3. B generates the same lower and upper approximations of each set-theoretic sum of the decision classes as C; in other words, $\cup\{[u]_B : [u]_B \cap Y \neq \emptyset\} = \cup\{[u]_C : [u]_C \cap Y \neq \emptyset\}$ as well as $\cup\{[u]_B : [u]_B \subseteq Y\} = \cup\{[u]_C : [u]_C \subseteq Y\}$ where $Y = \cup_{i=j_1}^{j_n} X_i$ for $j_1, \ldots, j_n \in \{1, \ldots, |V_d|\}$ and $X_1, \ldots, X_r$ are the r decision classes.

As presented in [**?**], we have the following interrelations among the above mentioned notions of decision reduct.

Theorem 4.1. *(i) The three notions of decision reduct presented in Definition 4.8 are equivalent.*
(ii) Each of the notions of decision reduct presented in Definition 4.8 is equivalent to the notion of ∂-decision reduct.

Theorem 4.2. *(i) Each of the notions of decision reduct presented in Definition 4.8 implies the notion of POS-decision reduct.*
(ii) The notion of μ-decision reduct implies the notion of ∂-decision reduct.

There is one more well known notion of decision reduct, which is stronger [**?**] than all the above notions of decision reduct.

Definition 4.9. Given $\mathbb{A} = (U,\ C,\ d)$ and $B \subseteq C$, B is said to be the *dis*-decision reduct if B is the smallest set such that B discerns the same pairs of objects with different decision values as C. That is, for every $u, u' \in U$, if $d(u) \neq d(u')$ and $[u]_C \neq [u']_C$ then $[u]_B \neq [u']_B$.

Theorem 4.3. *The notion of dis-decision reduct implies the notion of μ-decision reduct.*

4.2 Decision functions

When $(U,C\cup\{d\})$ is consistent, which means that $Ind(C)\subseteq Ind(\{d\})$ (or, in other words, every pair of objects belonging to different decision classes is discerned with respect to C), then all the variants of decision reduct, mentioned in Section 4.1 are equivalent to the classical formulation given in Definition 4.1. The relationships among these different notions of decision reduct, as mentioned in Section 4.1, become interesting when the underlying decision system is inconsistent [**?**]. More precisely, all these notions of decision reduct give rise to different decision functions which help to translate an inconsistent decision system to a consistent one.

We already have seen in the previous section how different notions of decision reduct are developed from different aspects of information synthesis. For example, in case of POS-decision reduct the main concern is to restore the information pertaining to the consistent part of a given decsion system. Whereas ∂-decision reduct and μ-decision reduct prefer to restore the decision information corresponding to the inconsistent objects as well; but they differ in restoring the details such as in what frequency a particular decision value occurs in a particular equivalence class, what is the cardinality of that particular equivalence class etc.

As these different notions of decision reducts are defined based on which part of the information of the original decision system is to be regarded as important and which part is to be ignored, they give rise to different attitudes of decision modelling. Thus, different notions of decision functions are developed based on them. Below we present a series of decision functions which are developed based on the notions of decision reduct presented in Section 4.1.

In the sequel below, for each considered variant of decision reduct given in the Definitions 4.3, 4.5, 4.7, 4.8, and 4.9, we present a translation of decision attribute d, for a given decision table $(U,C\cup\{d\})$, into a new decision attribute $d^{\#}$, in such a way that $(U,C\cup\{d^{\#}\})$ becomes consistent and the following hold.

- $d^{\#}$ agrees with d for all objects u for which equivalence class of $[u]_C$ is contained in a single decision class.
- The respective notion of decision reduct (Definitions 4.3, 4.5, 4.7, 4.8, 4.9) holds for a given $B\subseteq C$ in $(U,C\cup\{d\})$, if and only if $Ind(B)\subseteq Ind(\{d^{\#}\})$ holds in the consistent decision table $(U,C\cup\{d^{\#}\})$.

Below we present definitions for $d^{\#}_{POS}$, $d^{\#}_{\partial}$, $d^{\#}_{\mu}$, and $d^{\#}_{dis}$ so that for each respective translation of d the respective inclusions, namely $Ind(B)\subseteq Ind(\{d^{\#}_{POS}\})$, $Ind(B)\subseteq Ind(\{d^{\#}_{\partial}\})$, $Ind(B)\subseteq Ind(\{d^{\#}_{\mu}\})$, and $Ind(B)\subseteq Ind(\{d^{\#}_{dis}\})$ hold.

Definition 4.10. For an inconsistent decision system $(U,C\cup\{d\})$ and $B\subseteq C$, the translations of d are defined by the following decision functions.

(i) For POS-decision reduct $d^{\#}_{POS}$ is defined as

$$d^{\#}_{POS}(u)=\begin{cases} d(u) & \text{if } u\in POS_C(d)\\ \# & \text{otherwise}\end{cases}$$

where # ($\notin V_d$) is a new value.

(ii) For ∂-decision reduct $d^{\#}_{\partial}$ is defined as

$$d^{\#}_{\partial}(u) = \partial_C(u)$$

(iii) For μ-decision reduct $d^{\#}_{\mu}$ is defined as

$$d^{\#}_{\mu}(u) = \overrightarrow{\mu}_B(u)$$

(iv) For dis-decision reduct $d^{\#}_{dis}$ is defined as

$$d^{\#}_{dis}(u) = \begin{cases} d(u) & \text{if } u \in POS_C(d) \\ \#_{m(u)} & \text{otherwise} \end{cases}$$

where $\#_{m(u)}$ ($\notin V_d$) are the new decision values indexed by the ordinal numbers of the corresponding indiscernibility classes $[u]_C$, such that $\#_{m(u)} \neq \#_{m(u')}$ if $[u]_C \neq [u']_C$.

The decision system presented in Table 4.1 is extended to Table 4.2 by adding the columns for each of the above translated decision functions based on POS-decision reduct, ∂-decision reduct, μ-decision reduct and dis-decision reduct. Both $d^{\#}_{POS}$ and $d^{\#}_{dis}$ are defined by synthesizing the decision information corresponding to those objects for which the whole equivalence class behaves consistently. However, according to $d^{\#}_{POS}$, the objects belonging to an equivalence class with more than one decision values are simply ignored by putting a dummy decision value #. That is, if for u, u', $[u]_C$ and $[u']_C$ both contain inconsistent cases, then $d^{\#}_{POS}(u) = d^{\#}_{POS}(u')$ even when $[u]_C \neq [u']_C$. On the other hand, according to $d^{\#}_{dis}$ each object of the equivalence classes $[u]_C$ for which inconsistency in decision arises, is assigned to a unique dummy value $\#_{m(u)}$ where $m(u)$ encodes the ordinal number of the corresponding equivalence class. So, in contrast to $d^{\#}_{POS}$ the decision function $d^{\#}_{dis}$ does not completely ignore the information corresponding to the inconsistent objects as it prefers to keep track of the ordinal number of such equivalence classes. So, in some sense, according to $d^{\#}_{dis}$ though the dummy values assigned to the objects may not be used in the current context of decision making, the additional information tagged to them may be useful in further contexts of decision making. Actually, one can interpret dummy decision values # and $\#_{m(u)}$ with analogous differences in handling unknown values of conditional attributes in incomplete information systems [**?**]. Therein, two undetermined values could be, among other strategies, regarded as potentially the same (which is an analogous to #) or potentially different (which is an analogous to $\#_{m(u)}$). Compare to the above decision functions, in case of $d^{\#}_{\partial}$ the decision values corresponding to inconsistent elements are not erased from the new decision table. Those values are rather grouped together to show how diverse sets of decision values a particular indiscernibility class can assume. In case of $d^{\#}_{\mu}$, if objects u and u' are indistinguishable with respect to $Ind(C)$ but have different decision values, then they are unified under $d^{\#}_{\mu}$ by the vector of probability distribution of the decision values over the whole equivalence class. So, after translating Table 4.1 to Table 4.2

	m	d	$d^{\#}_{POS}$	$d^{\#}_{\partial}$	$d^{\#}_{\mu}$	$d^{\#}_{dis}$
o_1	1	high	high	{high}	$\langle 1,0,0\rangle$	high
o_2	1	high	high	{high}	$\langle 1,0,0\rangle$	high
o_3	1	high	high	{high}	$\langle 1,0,0\rangle$	high
o_4	2	moderate	#	{moderate, low}	$\langle 0,\frac{1}{2},\frac{1}{2}\rangle$	$\#_2$
o_5	2	low	#	{moderate, low}	$\langle 0,\frac{1}{2},\frac{1}{2}\rangle$	$\#_2$
o_6	3	low	#	{high, low}	$\langle \frac{1}{3},0,\frac{2}{3}\rangle$	$\#_3$
o_7	1	high	high	{high}	$\langle 1,0,0\rangle$	high
o_8	3	low	#	{high, low}	$\langle \frac{1}{3},0,\frac{2}{3}\rangle$	$\#_3$
o_9	3	high	#	{high, low}	$\langle \frac{1}{3},0,\frac{2}{3}\rangle$	$\#_3$

Table 4.2 New decisions constructed for Table 4.1. Column m denotes ordinal numbers of indiscernibility classes induced by the whole set of conditional attributes that objects belong to.

no information regarding different decision values of a particular equivalence class is lost as they are now encoded with the respective probabilities of the decision values. Therefore, compare to $d^{\#}_{\partial}$ the decision function $d^{\#}_{\mu}$ is more rich in encoding information.

From the variety of definitions of decision functions, discussed above, it is easy to observe that the given an object u from the universe and a set of attribute B ($\subseteq C$) the output under a particular decision function can vary from a set of decision values to a vector of probability distribution of the decision values, and even it can contain some auxiliary symbols as well. Instead of the universe of objects U, the domain of such functions can be rather considered as the vector of attribute values, or information signature based on the set of attributes B. The main point to be noted here, that though these decision functions return as output different kinds of mathematical entities, all of them follow certain common properties. To illustrate the issue let us consider the examples of $d^{\#}_{\partial}$ and $d^{\#}_{mu}$. To bring both of them under a uniform presentation let us consider a decision function $dec : Inf_B(U) \mapsto \Delta$ where $Inf_B(U)$ represents all possible vectors of values with respect to the set of attributes B, present in the decision system $\mathbb{A}$, and Δ is the set of r-dimensional vectors $\langle s_1, s_2, \dots, s_r\rangle$ such that $\Sigma^r_{i=1} s_i = 1$. For $d^{\#}_{\mu}$ the domain and range of dec perfectly match. The readers can check that for $d^{\#}_{\partial}(u)$ the output set of possible decision values can be equivalently presented as $\langle v_1, v_2, \dots, v_r\rangle$ where $v_i = \frac{1}{|\partial_B(u)|}$ if $v_i \in \partial_B(u)$, and 0 in all other cases. Thus, $d^{\#}_{\partial}$ also can be presented under dec.

Now, considering a decision system $(U, C \cup \{d\})$ some of the common properties of the above mentioned decision functions can be stated as below. A detailed discussion in this regard can be found in [].

(i) Zero property: If a decision value v_i does not occur in a particular equivalence class $[u]_C$, then $dec(Inf_C(u)) = \langle v_1, v_2, \dots, v_r\rangle$ where $v_i = 0$.
(ii) Monotonicity property: If for two decision classes X_i and X_j, $|X_i \cap [u]_C\}| \leq |X_j \cap [u]_C\}|$ then $dec(Inf_C(u))^i \leq dec(Inf_C(u))^j$ where the suffixes i and j represent the i-th and j-th component of $dec(Inf_C(u))$.
(iii) Inclusion property: If for some $u' \in [u]_C$, $d(u') = v_i$, then $dec(Inf_C(u'))^i > 0$.

Though the above listed properties are usually natural demands to a decision function, not all of them hold always. For example in case of $d^{\#}_{POS}$ the inclusion property does not hold. In order to present $d^{\#}_{POS}$ let us change the range set of dec by $\Delta \cup \{\langle 0,0,\dots 0\rangle\}$ where the null vector is introduced to represent the dummy value # introduced in Definition 4.10. Now let $u \notin POS_C(d)$; that is, there are more than one decision values occur in $[u]_C$. Hence, though $d(u) = v_i$, $dec(Inf_C(u)) = \langle 0,0,\dots 0\rangle$.

Interested readers may try to represent $d^{\#}_{dis}$ in the framework of dec and check its properties. Moreover, constructing different interesting decision functions reflecting common sense reasoning will be also a good task to explore. For many other interesting decision functions the readers are referred to [].

4.3 Decision rules

In Section 3.6 of Chapter 3 we have already discussed about the basic forms of decision rules. As mentioned in Section 3.6, for each object $x \in U$, a decision rule can be obtained, in the form of *if-then* kind of Boolean formulas, just by conjoining atomic formulas such as $a_i = a_i(x)$ for each $a_i \in C$ in the left hand side of the statement and $d = d(x)$ as consequent. Though such rules can represent correctly each case of a given decision system, they do not bother about the length and redundancy of attributes used in such decision rules. In order to generate the decision rules of minimum description length usually the decision rules are generated based on the available decision reducts for a given decision system. As a decision reduct represents the smallest set of attributes describing the nature of the whole decision system the decision rules obtained from a decision reduct guarantees rules of minimum description length.

So, in reference to Table 4.2, considering ∂-decision reduct it is enough to concentrate on the attribute a_3 for generating the decision rules for the decision system. Thus, the decision rules are namely, $(a_3 = moderate) \Rightarrow (d = high)$, $(a_3 = high) \Rightarrow (d = moderate) \vee (d = low)$, and $(a_3 = low) \Rightarrow (d = high) \vee (d = low)$.

Finding decision rules for a given decision system opens up a different branch of study. Abstraction of a proper set of decision rules leads to a proper modelling of the nature of the decision problem, and in turns helps to classify newly appeared cases significantly well. Apart from finding rules with minimum description length, one of the important matters of concern is how the search algorithm should be designed for a given decision system so that (i) decision rules are of minimum description length, (ii) all the objects are covered under the rules, (iii) number of rules is reasonably small, and (iv) complexity of finding the rules is not high.

In literature [**?**, **?**], there are several existing methods for finding decision rules for a given decision system. Sequential covering, Exhausting set of rules, Learn from examples by modules, Decision tree etc are a few to name. However, often in practice the methods for finding decision rules [**?**] starts with a random selection of single condition attribute and then adding further attribute one by one in order to reach a suitably small set of attributes covering all the objects of a given decision system.

For instance, in sequential covering [**?**] the method starts with searching rules of length one containing only one atomic formula with one condition attribute in the antecedent. Let us illustrate the method in the context of Table 4.1. For objects o_1, o_2, o_3, o_7 one possible rule can be $(a_1 = average) \Rightarrow (d = high)$. To be noted, that instead of the above rule either of $(a_2 = close) \Rightarrow (d = high)$ or $(a_3 = moderate) \Rightarrow (d = high)$ also can be considered as a valid rule describing the objects o_1, o_2, o_3, o_7. On the other hand, neither of a_1, a_2, a_3 can alone describe the other objects. The elements, which are already covered by a rule of length one, are thrown away from the next level of the search mechanism. Now, the algorithm must search for rules of length two. It can be noticed that no two-element subset of $\{a_1, a_2, a_3\}$ can unambiguously describe the decision classes corresponding to the uncovered elements. Even with respect to the whole set of attributes also it is difficult to find a unique decision class corresponding to the available vectors of values describing the uncovered cases. In such situations, there is a trade off between support (or coverage) of a rule and accuracy (or confidence) of a rule.

Let us explain the above mentioned notions in the context of Table 4.1. The description $(a_2 = far) \wedge (a_3 = low)$ matches three cases, namely o_6, o_8 and o_9. So, it has support 3. On the other hand, for $a_2 = far$ the support is 5. Now the accuracy of the rule $(a_2 = far) \Rightarrow (d = low)$ is $\frac{3}{5}$ as it correctly classifies three out of five cases belonging to the support of its antecedent. Whereas the accuracy of the rule $(a_2 = far) \wedge (a_3 = low) \Rightarrow (d = low)$ is $\frac{2}{3}$. So, comparing these rules we can observe that adding more attributes to the antecedent we have less number of support but it guarantees more accuracy.

Moreover, the problem of resolving the conflict with contradictory rules, such as rules with same antecedent and different decisions, may be handled based on their accuracy measures; for instance both the rules $(a_2 = far) \wedge (a_3 = low) \Rightarrow (d = low)$ and $(a_2 = far) \wedge (a_3 = low) \Rightarrow (d = high)$ have the same support but their respective accuracy measures are $\frac{2}{3}$ and $\frac{1}{3}$. Such kind of rule generation techniques along with the accuracy measures help in assigning decisions for unseen or test examples. However, there are several other issues [**?**] that often come across in the process of rule generation and assigning decisions to the newly appeared cases.

One such point leads to the situations where a newly appeared case does not match to any of the antecedent of the rules, generated from a given decision system or a set of training examples. Putting in formal terms, we start with a decision system (U_{trn}, C, d) where U_{trn} can be regarded as a set of training examples for which vectors of attribute values and corresponding decision values are known. However, in real-life decision problems a training set U_{trn} can be only a small sample of possible set U^* of objects that can occur in reality and hence the decision value is unknown for them. This is usually known as *classification problem* or *concept learning* [**?**]. In concept classification, there are different strategies to combine multiple rules in order to justify the new instances. One such very well known method is k-nearest neighbours, in short KNN [**?**, **?**]. For KNN the first requirement is to introduce a metric on the set of vectors of values for the set of condition attributes. Now based on that metric, for a given instance u_{ts} from $U^* \setminus U_{trn}$ the vector of values of that instance is compared with the vector of values for each $u \in U_{trn}$. Thus, a set $NN(u_{tst}, k)$ of k nearest neighbours close to u_{tst} is formed. Then the decision of u_{tst} is estimated based on $|D_i \cap NN(u_{ts}, k)|$ for the decision class D_i corresponding to each $i = 1, \ldots r$. Here, to be noted that there are variety of metrics, combining both real and symbolic values, are available in the literature. In this regard, the readers are referred to [**?**] where the author proposed different combinations for constructing a suitable metric from the set of given examples, based on the available *city-block distance* for numeric values and *Hamming distance* for symbolic values. Moreover, for a single metric there can be different estimation strategies as well. So, this leads to different possibilities for proposing a good hypothesis that nicely generalizes the concept learned based on the training set U_{trn}.

Another very important issue is the choice of a particular attribute in the process of generating rules of minimum length. For example, in case of Table 4.1 we already have discussed that for the objects o_1, o_2, o_3, o_7 one can choose either of the attributes a_1, a_2, a_3 in the antecedent of a rule that can unambiguously describe the decision value for all of them. So, one can choose randomly one of the attributes to describe the objects o_1, o_2, o_3, o_7. However, for generating further rules to cover rest of the objects it may happen that choice of one of the attributes among a_1, a_2, a_3 is more advantageous than others. Apart from that it is also important for designing a search algorithm to have a particular mechanism and reason behind choosing a particular attributes among several other possible ones. In this regard, let us introduce the notion of *entropy* [**?**, **?**] of an attribute which helps to measure how much information can be gained about the decision attribute by analysing a particular condition attribute. In this regard, a brief introduction to decision tree learning, one of the very well known decision rule learning methods, would be helpful.

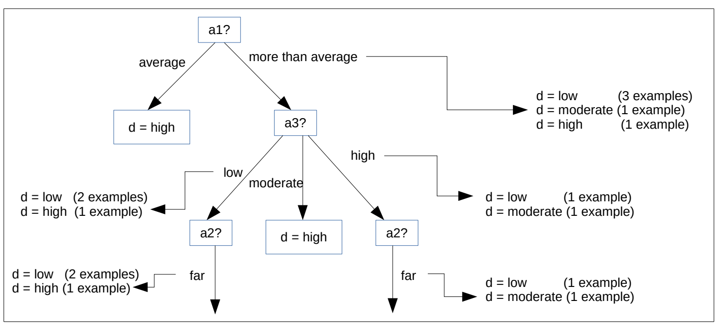
A decision tree learning method helps to generate a good hypothesis that generalizes a given decision table (U_{trn}, C, d). It represents a function that takes as input a vector of attribute values and returns a value for d. A decision tree is obtained by performing a sequence of tests. Each internal node in the tree corresponds to a test of the value of one of the condition attributes from C, and the branches from the node are labeled with the possible values of v_{ik} for that particular attribute, say a_i. Each leaf node in the tree specifies a decision value. For example, let us find a decision tree for the decision table described in Table 4.1.

Here one needs to decide which attribute to choose first to do the test. It is visible that by testing either of a_1, a_2, a_3 at the first node it can immediately lead to the decision node $d = high$ for objects o_1, o_2, o_3, o_7. For rest of the objects one needs to choose the next attribute to test. As shown in Figure 4.3, if we choose a_1 as the starting node then the branch corresponding $a_1 = average$ immediately reaches to the leaf node $d = high$ covering the objects o_1, o_2, o_3, o_7. The branch corresponding to *more than average* contains mixed examples classified by different decisions. As shown in the right side of the branch, it classifies three examples with $d = low$ and one for each of $d = high$ and $d = moderate$. So, the method goes forward for next level of testing attributes. Here also, among the two remaining attributes a_3

is chosen. The question is what should be the guiding principle for choosing an attribute to test at a particular level.

It can be noticed that in the decision tree presented in Figure 4.3 at the first level there are three possible choices of attributes, each of which immediately can classify four objects with $d = high$. However, among them selecting a_1 or a_2 leads to generation of two branches and for a_3 there are three branches. So, at this stage selecting an attribute with less number of branches may be preferred. So, among a_1, a_2 the left most attribute of Table 4.1 is chosen. However, at the next level with $a_1 =$ *more than average*, testing a_2 generates only one branch and testing a_3 generates three branches. While the disadvantage of testing a_2 is that $a_2 = far$ cannot classify the remaining objects by a single decision. Whereas, if a_3 is selected for testing then one object can be classified under one of its branches, namely $a_3 = moderate$. So, at this level number of objects classified by a test is preferred over number of branches generated by a test. Combining these factors the notion of Entropy is designed to measure how much *information is gained* by testing a particular attribute, and that in turn helps to select an attribute with maximum information gain.

Fig. 4.1 Steps for finding a decision tree for Table 4.1



The notion of entropy was proposed by Shannon and Weaver [**?**] as the fundamental quantity of information theory. Entropy aims to measure the uncertainty of a random variable and it is considered that acquisition of information corresponds to a reduction in entropy. A random variable with only one value has no uncertainty and thus its entropy is defined as zero. So, by selecting such a variable to test we gain no information. In general, the entropy of a randomly chosen attribute a_i, denoted as $H(a_i)$, with all possible values v_{ik} and respective probability of occurrence $P(v_{ik})$, is defined as follows.

$$H(a_i) = \Sigma_{v_{ik}} P(v_{ik}) \frac{1}{log_2(P(v_{ik}))} = -\Sigma_{v_{ik}} P(v_{ik}) log_2(P(v_{ik}) \tag{4.1}$$

So, clearly for the decision attribute d corresponding to the decision system presented in Table 4.1 $H(d)$ = -($\frac{5}{9}log_2(\frac{9}{5})+\frac{1}{9}log_2 9+\frac{1}{3}log_2 3$). This indicates that the decision attribute d has $H(d)$ amount of information uncertainty. Now question comes how the testing of a condition attribute influences in decreasing the information uncertainty of $H(d)$; or in other words, how we can gain information about d by testing some condition attribute a_i.

In this regard, first we need to calculate the entropy for a particular condition attribute a_i, and this information content is aggregated based on each random variable $a_i = v_{ik}$ for each value v_{ik} of a_i. For example, based on Table 4.1 we can say $H(d|a_1 = average)$ is 0 as it always leads to a single decision $d = high$. On the other hand $H(d|a_1 = more\ than\ average)$ = $(\frac{3}{5}log_2(\frac{3}{5})+\frac{1}{5}log_2(\frac{1}{5})+\frac{1}{5}log_2(\frac{1}{5}))$. Moreover, the probability that a randomly chosen object from U satisfies $a_1 = average$ is $\frac{4}{9}$ and that of $a_1 = more\ than\ average$ is $\frac{5}{9}$. So, when testing a_1 along the branch of $a_1 = average$ the amount of information learnt is $H(d|a_1 = average)P(a_1 = average)$ and that of for the branch $a_1 = more\ than\ average$ is $H(d|a_1 = more\ than\ average)P(a_1 = more\ than\ average)$. Thus, by testing a_1 we learnt the following amount of information.

$$\Sigma_{v\in V_{a_1}} P(v)H(d|a_1 = v) = \frac{4}{9}H(d|a_1 = average)+\frac{5}{9}H(d|a_1 = more\ than\ average)$$
$$= \frac{5}{9}(\frac{3}{5}log_2(\frac{3}{5})+\frac{1}{5}log_2(\frac{1}{5})+\frac{1}{5}log_2(\frac{1}{5})) = \frac{1}{3}log_2(\frac{3}{5})+\frac{2}{9}log_2(\frac{1}{5})$$

So, clearly information uncertainty in d, denoted by $H(d)$ is decreased by information content $\Sigma_{v\in V_{a_1}} P(v).H(d|a_1 = v)$ which is learned by testing the condition attribute a_1. So, the formula for *information gain* about the decision attribute d after testing a condition attribute a_i is as follows.

$$Gain(d,a_i) = H(d) - \Sigma_{v\in V_{a_i}} P(v)H(d|a_i = v) \tag{4.2}$$

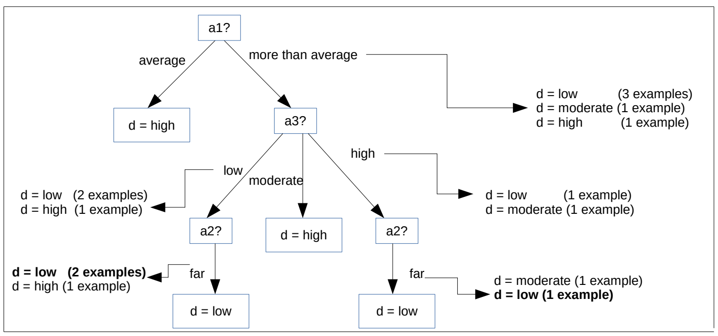
For Figure 4.3 we already have calculated $H(d) = -(\frac{5}{9}log_2(\frac{9}{5})+\frac{1}{9}log_2 9+\frac{1}{3}log_2 3)$. So, we can make the following observations.

$$Gain(d,a_1) = H(d) - (\frac{1}{3}log_2(\frac{3}{5})+\frac{2}{9}log_2(\frac{1}{5})) \tag{4.3}$$

The best attribute to test at the root node is selected as the condition attribute a_i if $Gain(d,a_i)$ is maximum. In the next level, the process is repeated to choose the next attribute to test over the set of objects that are not classified in the previous level. This forms a greedy search for an acceptable decision tree, in which the algorithm never backtracks to reconsider earlier choices. Though it provides a reasonable way to select a *fairly good* attribute to test at each level and generate a significantly smaller tree that is consistent with the examples, finding the smallest consistent tree still an intractable problem.

At the end let us return to Figure 4.3. It can be noticed that Figure 4.3 does not represent a decision tree for Table 4.1 as not all its branches are closed by leaf nodes. The reason is that the original decision system presented in Table 4.1 is an inconsistent decision system, and the algorithm for finding a decision tree, as described above, works well when we have a consistent decision system. So, in order to create a decision tree for inconsistent decision system one may first convert it into a consistent table following the techniques discussed in Section 4.2. Moreover, there are other interesting ways to construct a consistent decision tree out of an inconsistent decision system as well. For example, following [**?**], a possible decision tree for Table 4.1 can be as follows. The important point to notice here that as for $a_2 = far$ both the combinations with $a_3 = low$ and $a_3 = high$ have objects satisfying decision $d = low$ the method picks up $d = low$ as the leaf node for those branches. The inductive reasoning strategy, here, can be justified by focusing on common instances or larger support.

Fig. 4.2 A possible decision tree for Table 4.1



References

1. Analogy-Based Reasoning in Classifier Construction. Transactions on Rough Sets IV (LNCS 3700), 277–374 (2005)
2. Azad, M., Moshkov, M.: Classification and optimization of decision trees for inconsistent decision tables represented as mvd tables. In: Proceedings of the Federated Conference on Computer Science and Information Systems
3. Dash, M., Liu, H.: Consistency-based search in feature selection. Artif. Intell. (151(1–2)), 155–176 (2003)
4. Juszczuk, P., Kozak, J., Dziczkowski, G., Głowania, S., Jach, T., Probierz, B.: Real-world data difficulty estimation with the use of entropy. Entropy (LNCS 4100), 1–36 (2021)
5. Kryszkiewicz, M.: Comparative study of alternative types of knowledge reduction in inconsistent systems. Int. J. Intell. Syst. (16(1)), 105–120 (2001)
6. Mitchell, T.M.: Machine Learning. McGraw-Hill Science/Engineering/Math, New York (1997)

7. Polikar, R., DePasquale, J., Mohammed, H.S., Brown, G., Kuncheva, L.I.: Learn++.MF: A Random Subspace Approach for the Missing Feature Problem. Pattern Recognition **43**(11), 3817–3832 (2010)
8. Russell, S., Norvig, P.: Artificial Intelligence: A Modern Approach, Third Edition. Prentice Hall, Upper Saddle River, New Jersey (2010)
9. Shannon, C.E., Weaver, W.: The Mathematical Theory of Communication. University of Illinois Press (1949)
10. Skowron, A., Wang, H., Wojna, A., Bazan, J.: Multimodal classification: Case studies. Transactions on Rough Sets V (LNCS 4100), 224–239 (2006)
11. Ślęzak, D., Dutta, S.: Dynamic and discernibility characteristics of different attribute reduction criteria. In: H. S. Nguyen et al. (Eds.): IJCRS 2018,
12. Tran, T.N., Afanador, N.L., Buydens, L.M., Blanchet, L.: Interpretation of Variable Importance in Partial Least Squares with Significance Multivariate Correlation (SMC). Chemometrics and Intelligent Laboratory Systems **138**, 153–160 (2014)

Chapter 5
Rough Set Based Networks

Nowadays, issues related pertaining to data analysis related to on the basis of different kinds of networks are becoming more and more important for real-life applications [**?**, **?**, **?**, **?**, **?**, **?**, **?**, **?**].

In this chapter, we discuss examples of such approaches. The first one is based on multi-granulation rough sets. The second one is based on relationships of rough sets with the information flow approach. The third one is related to process mining based on rough sets.

5.1 Multi-granulation rough sets

In contrary to classical Pawlak's model of approximation space with single indiscernibility relation multi-relational approximation spaces consist of many relations rather than a single indiscernibility relation as it is in the classical Pawlak model of approximation space.

In this section, we present examples showing that many problems to be solved by Intelligent Systems (IS) are optimization problems based on searching for the (semi-) optimal spaces in large families of approximation spaces generated from multi-relational approximation spaces by applying different aggregation operations. Among them are problems of data reduction, attribute (feature) selection, and feature extraction (feature engineering) as well as developing learning algorithms in Machine Learning (ML) [**?**, **?**, **?**, **?**, **?**]. We emphasize the close relationships of information systems [**?**, **?**] and/or multi-granulation systems. Moreover, we show an important role of families of multi-granulation systems generated by aggregation operations over multi-granular spaces as the basis in searching for the relevant approximation spaces. This searching can be based on the space of the information (decision) systems representing the multi-relational-approximation spaces.

The beginning of multi-granulation rough set approach is usually referred to usually refers to the papers from 90-ties of the XX century by Cecylia Rauszer and Helena Rasiowa with Victor Marek (see, *e.g.*, [**?**, **?**, **?**, **?**]). They considered a team of agents having at their disposal indiscernibility relations and considered, in particular, for any object aggregation of their voting *for* and *against* of a particular decision aggregating their votes *for and against* of a particular decision corresponding to an object.

A formal definition of multi-relational approximation space is as follows.

Definition 5.1. A multi-relational approximation space is any tuple

$$AS = (U, \{r\}_{r \in R}),$$

where R is a set of binary relations over a set U.

One can consider generalizations of this definition to the case where R is by considering R as a set of fuzzy binary relations, in particular fuzzy equivalence relations. Then, it is possible to use method based on combination of the rough set and fuzzy set approaches (see Chapter **??**). Another possibility is to use methods developed using the covering based approach (see Chapter **??**).

The Pawlak's model of rough sets [**?**, **?**, **?**] is defined using an approximation space

$$AS = (U, r),$$

where U is a finite set and r is an equivalence relation over U.

Then, for any $X \subseteq U$ is defined its lower $LOW(r,X)$ and upper approximation $UPP(r,X)$ by $\{x \in U : [x]_r \subseteq X\}$ and $\{x \in U : [x]_r \cap X \neq \emptyset\}$, respectively. Moreover, the boundary region $BN(r,X)$ is defined by $UPP(r,X) \setminus LOW(r,X)$. - IS THIS PART REQUIRED TO BE REPEATED? WE MAY REFER TO THE RESPECTIVE CHAPTER.

Let us now consider a simple illustrative example related to the case of multi-relational approach.

Example 5.1. In Figure 5.1 is illustrated a family of agents $\{ag_1, \ldots, ag_k\}$ approximating a concept $X \subseteq U$ using attribute sets $\{A_1, \ldots, A_k\}$. SHOULD WE USE HERE THE SAM SUFFIX k? The classification (voting) of X relative to these sets of attributes are represented by decisions from $\{d_1, \ldots, d_k\}$. Values of these decisions are pointing out to approximation regions of X relative to these sets of attributes: the lower approximation (1), boundary region (0.5) and complement to the upper approximation (0). The concept X may be now approximated relative to the set of attributes $\{d_1, \ldots, d_k\}$. In particular, one can consider voting functions based on $\{d_1, \ldots, d_k\}$ or classifiers induced from $\{d_1, \ldots, d_k\}$ for d_X. One can observe that the considered family of agents defines a multi-relational approximation space $(U, \{IND(A_1), \ldots, IND(A_k)\})$.

One should observe that, in general approximation of X relative to the union $A_1 \cup \ldots A_k$ can be of the higher quality than approximation of this concept relative to $\{d_1, \ldots, d_k\}$ (see Problem 5.1.) NOT VERY CLEAR

From these definitions, one can observe that in Pawlak's model, the approximation space is treated as given a priori, and approximations are defined relative to this approximation space. However, in general, we search for approximations of concepts

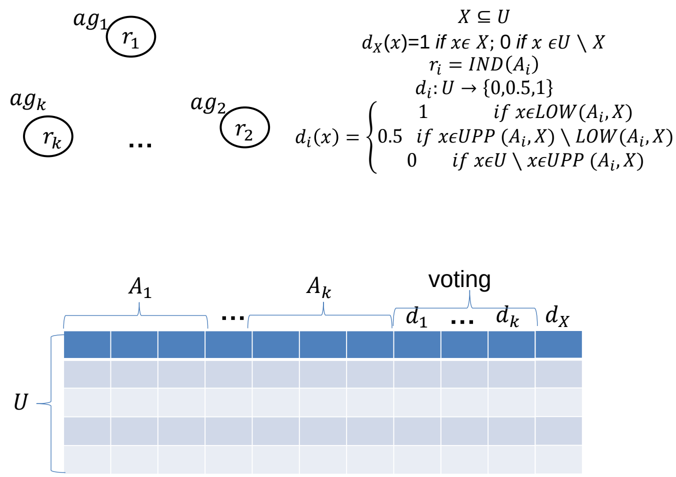


Fig. 5.1 Concept approximation in multi-relational approximation space.

over an extension of U. This requires developing reasoning techniques to support optimization in searching for the relevant approximation spaces from different, often huge, families of approximation spaces (see, *e.g.*, [**?**, **?**, **?**, **?**]). We discuss the role of information systems in this searching process.

One can observe that the Pawlak model of rough sets based on information systems is directly related to multi-relational approach too.

In fact, any information system IS defines a multi-relational system

$$AS_{IS} = (U, \{r_a\}_{a \in A}),$$

where $r_a = IND(\{a\})$ for $a \in A$.

Also any multi-relational approximation system $(U, \{r\}_{r \in R})$, where R is a finite family of equivalence relations with functions f_r for $r \in R$ labelling relations $r \in R$, *i.e*, bijections of the partition U/r of U defined by r onto $\{1, \ldots, | U/r |\}$, define an information system $IS^* = (U, A^*)$, where $A^* = \{a_r : r \in R\}$, and $a_r(x) = f_r([x]_r)$ for $x \in U$.

I THINK THE WHOLE PART BELOW WOULD NOT BE MUCH CLEAR TO THE READERS. CAN THIS BE MORE ELABORATELY EXPLAINED? One can observe that $IND(a_r) = r$ and the indiscernibility relation $IND(A^*)$ of IS^* is equal to

$$\bigcap_{r \in R} r.$$

Let us note that the indiscernibility relation $IND(A)$ of IS is invariant to renaming of attribute values. More formally, $IND(A) = IND(F(A))$, where $F(A) = \{f_a \circ a : a \in A \ \&\ f_a$ is a bijection of V_a onto $V_a\}$ and $(f_a \circ a)(x) = f_a(a(x))$ for $x \in U$.

The idea of the multi-relational approach to rough sets was further developed by other researchers (see, *e.g.*, [**?**, **?**, **?**, **?**, **?**, **?**, **?**]). In particular, the approach has been extended to covering based approach [**?**, **?**], where in multi-relational approximation spaces are considered *e.g.*, tolerance, similarity or even arbitrary binary relations.

Now we present several examples of optimization problems related to searching for optimal multi-relational approximation spaces in large families of such spaces. These examples are illustrating the importance of multi-relational approximation spaces in different areas of applications such as data reduction [**?**], construction of learning algorithms (*e.g.*, decision trees [**?**, **?**, **?**], rule-based classifiers [**?**, **?**, **?**]), discretization or symbolic value grouping [**?**, **?**], distributed learning [**?**, **?**, **?**, **?**, **?**].

In multi-relational approximation spaces with labelling functions, we represent objects based on their signatures in information systems representing them. These signatures are used to capture the relationships between objects and a set of attributes. For single attributes, equivalence classes are represented by descriptors. These descriptors take the form (a, v), where a is the attribute and v is the value of that attribute for a specific object x. The intersection of equivalence classes for single attributes is then described by combining their corresponding descriptors using conjunctions. The situation becomes different when we consider fuzzy sets (or rough fuzzy sets) as semantics for signatures of objects. First, these fuzzy sets are defined (induced) over equivalence classes of the original attributes. Then, descriptors representing the fuzzy sets are connected by fuzzy connectives in constructing formulas from signatures. These connectives, being generalization of conjunction, determine how the fuzzy sets corresponding to aggregations of descriptors are defined. One should also note that the IcS control - ??? may use some strategies for discovery of these connectives from data [**?**]. In this way, one can create expressive languages to describe concepts (c-granules) that serve as building blocks for understanding perceived situations. In the existing solutions, these languages are proposed by experts in dialogues of IS with them.

5.1.1 Multi-relational approximation spaces in optimization

In this subsection, we discuss several illustrative examples demonstrating importance of multi-relational approximation spaces in searching for (semi-)optimal solutions of different problems. Developing parallel/distrubued algorithms supporting efficient searching for such solutions is a challenging problem.

Let us first consider the information reduction problem in multi-relational approximation space $AS = (U, \{r\}_{r \in R})$ in the case when R is a finite set [**?**]. This is a searching problem for minimal subsets R' of the finite set R preserving discernibility of objects from U, *i.e.*, minimal subsets $R' \subseteq R$ (called reducts of R) such that

$$\bigcap_{r \in R} r = \bigcap_{r \in R'} r.$$

The considered problem is the problem of searching in a given family of equivalence relations for (semi-)minimal subset of equivalence relations preserving some given conditions. These problem are of high computational complexity. Hence, efficient heuristics have been developed to provide acceptable solutions. In this case Boolean reasoning methods were successfully supporting construction of efficient heuristics searching for semi-optimal solutions of the considered problem. This case is related to feature selection in Machine Learning (see, *e.g.*, [**?**, **?**] and Chapter **??**).

Our second example concerns decision trees. The aim is to give an illustrative example of importance of the optimization problems concerning searching for (semi-)optimal partitions of a given universe of objects in large families of partitions generated from some given partitions (or equivalently searching for (semi-)optimal equivalence relations in large families of equivalence relations generated from some given equivalence relations). In the considered example, these resulting partitions are approximations of a given partition defined by a given decision attribute.

One can consider different problems related to construction of decision trees. For example, the problem of constructing of the decision tree with the minimal depth. This problem is of the high computational complexity (NP-hard problem). Hence, in applications are used efficient heuristics constructing semi-optimal solutions (see *e.g.*, [**?**]). For more details on theoretical aspects of decision trees the reader is referred, *e.g.*, to [**?**, **?**, **?**].

Let us assume that is given a decision system $DS = (U, A, d)$ and the corresponding to DS multi-relational approximation space $AS = (U, \{r_a\}_{a \in A} \cup \{r_d\})$, where $r_a = IND(a)$ and $r_d = IND(d)$. Decision trees for AS are aiming to approximate the partition created by d by means of partitions corresponding to indiscernibility relations r_a for $a \in A$. A construction of a decision tree for AS is based on constructions of partitions of the universe U on the basis of partitions corresponding to r_a for $a \in A$. Below we outline an illustrative example of one of the heuristics.

The initial state of construction is created by a pair (U, ε) that can be called an initial node of the constructed tree. At each next step of construction is given a family of pairs, called nodes, of the form (X, des), where $X \subseteq U$ and des is a sequence of descriptors over some attributes $a \in A$. For any $a \in A$ in any seq there is at most one descriptor with a. In the subsequent step are considered nodes with longest sequences des. Such a node is called terminal if X is is included in one of the decision class of d or any attempt which will be described below is not improving the impurity of X measured relative to partition created by the decision attribute d. There can be used different impurity measures, *e.g.*, based on entropy or Gini index (see, *e.g.*, [**?**]). If such a node (X, des) is not terminal there are considered partitions of X defined by attributes not appearing in seq of the following form

$$\{X \cap \parallel a = v_1 \parallel, \ldots, X \cap \parallel a = v_i \parallel, \ldots, X \cap \parallel a = v_k \parallel\},$$

where $\parallel a = v_i \parallel = \{x \in U : a(x) = v_i\}$ for $i = 1, \ldots, k$ and $v_1, \ldots, v_k$ are all values from V_a for which $X \cap \parallel a = v_i \parallel$ is nonempty. Next, the considered node (X, des) is substituted by a family of nodes

$$(X \cap \parallel a = v_1 \parallel, (des, a = v_1)), \ldots, (X \cap \parallel a = v_k \parallel, (des, a = v_k)),$$

where $a \in A$ is the attribute for which the value of the used impurity measure (relative to the partition of d) is optimal. One should note that the node (X, des) is not extended if splitting this node by any attribute not leads to better value of the impurity measure.

Figure 5.2 illustrates a sequence of partitions of an exemplary universe, obtained by successively selecting a region to split within the current partition and choosing an attribute (binary attribute in the example) for the split. As this sequence of partitions is constructed, the decision tree is developed. The final partition in the sequence defines the partition used by the decision tree for classification. Based on this partition, different regions corresponding to objects marked by distinct colors representing decisions are approximated. A (semi-)optimal partition is one for which the corresponding decision tree is (semi-)optimal according to the specified metrics, such as the minimal depth of the decision tree or its overall size. Various criteria exist for selecting attributes for splitting (see, *e.g.*, [**?**]). It is important to note that the resulting decision tree classifies objects based on their features relative to the given attribute set (in this case, $\{a, b, c, e\}$). Consequently, test objects not included in the initial partition can also be classified. The quality of decision trees as classifiers is assessed using different metrics [**?**]. In particular, metrics related to Minimum Description Length (MDL) Principle [**?**] such as pruning decision trees and various measures based on the confusion matrix [**?**] are well-known for estimating the quality of decision trees for preventing overfitting, improving generalization, and enhancing model interpretability.

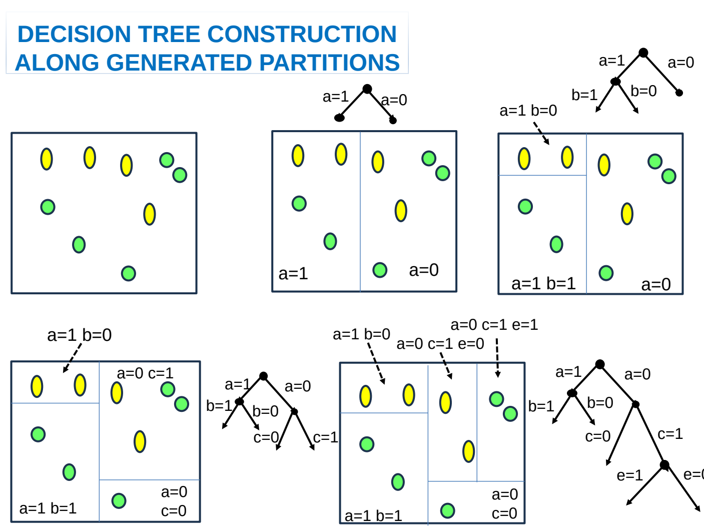


Fig. 5.2 Construction of decision tree based on generation of partitions.

It is important to note that, alongside the calculus of partitions in the discussed generalization of approximation spaces, we should provide reasoning techniques to support the search for relevant partitions that allow for high-quality approximations of given concepts, relative to specified quality metrics. Typically, reasoning begins with some basic partitions and gradually supports the generation of additional partitions aimed at achieving a (semi-)optimal partition.

In more advanced applications, it is essential to provide reasoning techniques that support the adaptation of partition generation in response to perceived changes in the physical environment.

In Chapter **??**, we discussed the covering rough set approach, which focuses on coverings of the universe of objects instead of partitions (see Figure 5.3). In this context, the family of definable sets is more complex, as they are defined by Boolean set operations rather than solely by unions. Languages that define subfamilies of definable sets within the covering rough set approach enable us to express more intricate properties. However, the computational complexity of searching for relevant definable sets increases as a result. Furthermore, the covering-based rough set approach is particularly applicable in certain contexts, such as in the inducing of rule-based classifiers.

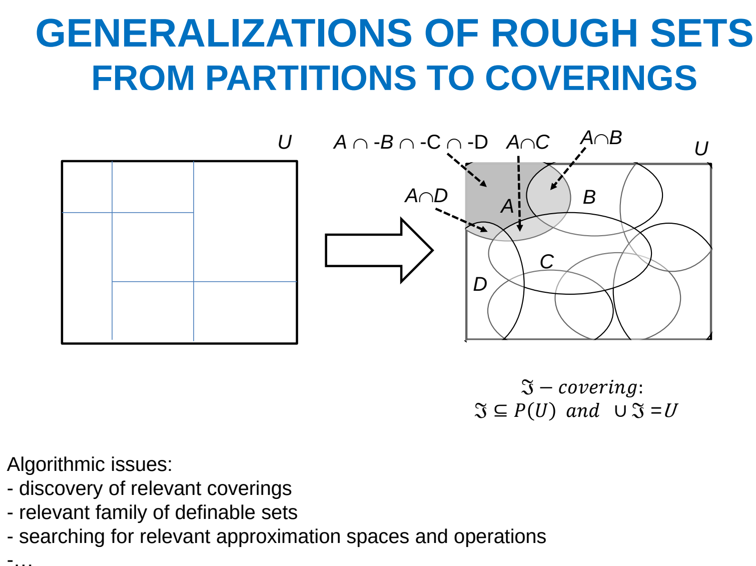


Fig. 5.3 From partitions to coverings

We would like to comment on the relationship of multi-relational approximation spaces with finite universes U and arbitrary binary relations $r \in R$ in the context of information systems. Each relation r in such a system can be represented by a family of neighborhoods defined as $r(x) = \{y \in U : xry\}$, where $x \in U$. In this case, definable sets are Boolean combinations of these neighborhoods. To approximate concepts within U, it is essential to select relevant attributes from the characteristic functions of these definable sets.

It is important to note that constructing such information systems can pose challenges due to the large number of definable sets. For example, one should note that from a given covering of U one can define a partition created by non-empty definable sets which are intersections of elements of the covering and their complements (see Figure 5.3). However, taking characteristic functions of such definable sets as attributes may lead to overfitting in approximation of concepts. Hence, for practical applications, it is necessary to seek constructive methods for discovering the relevant definable sets to achieve the high-quality approximations of concepts defined in extensions of U relative to given quality metrics. Note that these definable sets can be viewed as examples of computational building blocks essential for cognition (using the terminology of Valiant [**?**]) or as c-granules in the framework of IGrC.

In applications, the search for relevant attributes is often restricted to a language in which only certain definable sets can be expressed for making the search feasible within that language. It is also crucial to support the search for relevant computational building blocks with reasoning that considers the risks of overfitting and the description length of the blocks. This is related to the Minimum Description Length (MDL) principle [**?**].

An illustration of this can be seen when extending a given multi-relational approximation space $AS = (U, \{r\}_{r \in R})$, where R is a set of equivalence relations by adding to R new equivalence relations that are coarser than some relations from R. The goal is to approximate with high quality the partition of the universe of objects based on a specific decision attribute, utilizing this newly extended multi-relational approximation space and a quality measure grounded in MDL. The search for semi-optimal solutions within this typically large multi-relational approximation space can be effectively supported by Boolean reasoning (see, *e.g.*, [**?**, **?**]).

It's important to remember that there are many ways to define families of partitions based on equivalence relations relevant in searching for solution of a specific problem. For example, in binary decision tree construction [**?**] (see Figure 5.2), these partitions are defined using single equivalence classes and their complements. At each step of decision tree construction, searching is based on choosing the 'best' (using specific criteria like entropy) equivalence class among such partitions. Another example, can be related to classification by ensembles (see, *e.g.*, [**?**]). In this case, searching for information systems or family of equivalence relations on the basis of which members of ensembles are constructed plays the important role.

One should note that instead of binary relations over U one can consider fuzzy relations or relations obtained by combination of rough and fuzzy approaches. This approach based on combination of rough and fuzzy approaches defines important spaces of computational building blocks for cognition. In particular, these blocks may be defined by rough-fuzzy (or fuzzy-rough) aggregation of neighbourhoods. This approach led to success in many projects (see *e.g.*, [**?**, **?**, **?**, **?**, **?**, **?**, **?**, **?**] and Section 10.2 in Chapter 10).

The discussed above simple multi-relational models $(U, \{r\}_{r \in R})$ generated by information systems were used with a special kind of reasoning, called Boolean reasoning, in searching for solutions of many problems related, *e.g.*, to reduction of attributes, discretization or symbolic value grouping (see, *e.g.*, [**?**, **?**]). Discretization or symbolic value grouping concerns searching for the optimal transformation of a given multi-granular system AS_{IS} to a multi-granular system $AS_{IS'} = (U, \{r'_a\}_{a \in A'})$, where

- $A' \subseteq A$;
- for any $a \in A'$, r'_a is coarser than r_a, *i.e.*, $[x]_{r_a} \subseteq [x]_{r'_a}$ for $x \in U$;
- indiscernibility provided by A is preserved, *i.e.*, $IND(A) = IND(A')$ and
- description length of all indiscernibility classes of r'_a for $a \in A'$ is minimal, *i.e.*, the sum
$$\sum_{a \in A'} | \, U/_{r'_a} \, |$$
is minimal.

Usually, this problem is considered for decision systems and then the formulation should be accordingly changed (see [**?**, **?**] and Chapter **??**). One should note the (semi-)optimal approximation spaces are related to a proper generalization. For example, in the case of discretization 'linked' real values defining the original equivalence classes are generalized to intervals.

One should also bear in mind that:

- Quite often we deal with optimization problems related to searching for the optimal approximation space; hence, reasoning techniques supporting searching for the (semi-)optimal solutions are of great importance.
- In the case of multi-relational approximation spaces with relations different from equivalence relations, the definition of concept approximation is not unique (see, *e.g.*, [**?**, **?**]) and in applications one should provide reasoning tools supporting searching for the relevant schemes of concept approximation.
- In the case of multi-relational approximation spaces the sets U as well as the set of relations R are not necessarily finite. Such a situation is typical for problems of feature extraction (feature engineering). For example, one can consider as the set of attributes the characteristic functions of half-spaces defined by hyperplanes defined by some real-value attributes. Any such a binary attribute defines a partition of objects in a given finite universe of objects. In optimization, we are searching for the (semi-)minimal number of hyperplanes creating a partition of objects in which objets with different decisions are separated (see, *e.g.*, methods of Boolean reasoning [**?**] or more advanced criteria concerning Support Vector Machines (SVM) [**?**].)

It is also worthwhile mentioning here the relationships of the covering rough set approach with information systems. First of all, let us observe that in the definition of information systems by Pawlak together with the value sets V_a is considered the equality $=$, *i.e.*, the relational structure $(V_a, =)$. In discussion on discretization problem, we consider the relational structures $(V_a, \leq)$, where V_a is a subset of the set of reals and $\leq$ is a linear order. Considering similarity relations over V_a leads to relational structures of the form (V_a, ρ_a). Together with these relational structures over the value sets of attributes are considered languages of formulas with semantics expressed by subsets of the value sets or their Cartesian products. The characteristic functions of these sets can be considered as possible new attributes (over tuples of objects). Moreover, they can be used as constraints in aggregation of information systems for filtering tuples of objects satisfying these constraints (see, *e.g.*, [**?**, **?**]). This can also be used in definition of types of information systems [**?**].

The discussed approach allows us to generalize the indiscernibility relation defined in information systems as an equivalence relation to the indiscernibility being tolerance, similarity relation (see *e.g.*, [**?**]) or even general binary relation over signatures of objects (see, *e.g.*, [**?**, **?**]). More formally, in generalized information systems $IS_\tau = (U, A, \tau)$, where τ is a similarity relation over signatures of objects from U, objects $x, y \in U$ are τ-indiscernible in symbols

$$xIND(IS_\tau)y \text{ if and only if } inf_A(x)\ \tau\ inf_A(y).$$

In one of the above discussed cases we deal with the optimization problem in infinite space. Moreover, one can also consider another important problem for ML related to discovery of languages from which the relevant attributes should be extracted [**?**, **?**].

Let us note that searching for approximation of concepts in the space of all definable sets *i.e.*, arbitrary unions of indiscernibility classes) may be infeasible from the point of view of computational complexity. Hence, these methods are restricted to searching in subspaces of this space, *e.g.*, defined by definable sets determined by intersection of some equivalence classes from R.

I have read up to this section. I feel many related issues are discussed without going much deep into them - I mean, different concepts are discussed just giving a short description linguistically. As a result a mathematical step-by-step visualization of constructing one notion from another or their interrelation is not clear. If we consider this as a text book, then I feel maybe some of the concepts/notions must be developed step-by-step and then some related issues can be discussed without going into detail. It will be easy to connect for the readers.

5.1.2 Distributed multi-relational approximation spaces

In this subsection, we discuss applications of multi-relational approximation spaces in distributed systems. One can consider different multi-relational approximation spaces assigned to different agents. We discuss a problem of aggregation of approximations generated by these systems allocated to different agents for producing the global decisions. This is related to different problems considered in applications. For example, one can consider aggregation of decisions in ensemble learning of classification [**?**] or problems in federated learning [**?**, **?**, **?**]. The main constraint is that it is not possible to induce approximations of concepts on the basis of the global decision system obtained by aggregation of decision systems allocated to different agents.

In the sequel we consider a simple illustrative example.

We consider a family of local Pawlak's approximation spaces $\mathscr{AS} = \{AS_i : i = 1,\ldots,N\}$, where $AS_i = (U,R_i)$ for $i = 1,\ldots,N$ is a local approximation space with the universe of objects U. A given concept $\subseteq U$ is approximated relative to each local approximation space. In the result, for any object $x \in U$ and any local approximation space from the considered family is provided one of the following vote:

- 1 - if x belongs to the lower approximation of X relative to the considered local approximation space;
- 0.5 - if x belongs to the boundary region of X relative to the considered local approximation space;
- 0 - if x belongs to the complement of upper approximation of X relative to the considered local approximation space;

In this way, for any object $x \in U$ the considered family of approximation spaces $\mathscr{AS}$ provides a sequence of votes. Now, to this sequence of votes is applied a voting function returning the global vote of the family concerning x.

Let us start from definition of voting function related to a family of local Pawlak's approximation spaces aiming to resolve conflicts between votes of local approximation spaces concerning membership of a given object x to approximation regions of X constructed relative to local approximation spaces.

More formal description is as follows. By Σ is denoted a set $\{1,0.5,0\}$ and by Σ^N is denoted the set of sequences over Σ of the length N.

Definition 5.2. Let $\mathscr{AS} = \{AS_i : i = 1,\ldots,N\}$ where $AS_i = (U,R_i)$ for $i = 1,\ldots,N$ are Pawlak's approximation spaces and let $X \subseteq U$ be a concept in U. Any function

$$vote : \Sigma^N \longrightarrow \Sigma$$

is called a *voting function* for the family $\mathscr{AS}$ of approximation spaces and concept X. The pair $AS = (\mathscr{AS}, vote)$ is called a *generalized approximation space*.

The intuition behind the above defined voting function is the following. This function aggregates the votes on membership of x coming from approximation spaces from $\mathscr{AS}$ and returns an aggregated vote for exactly one of the region: lower approximation, boundary region or complement of the upper approximation of X. Each vote denotes the membership to exactly one of the regions: 1 for the lower approximation of X, 0.5 for the boundary region of X and 0 - for the complement of the upper approximation of X.

We define for $i \in \{1,\ldots,N\}$ the i-the voting function by

$$vote_i(x) = \begin{cases} 1 & \text{for } x \in LOW(AS_i,X) \\ 0.5 & \text{for } x \in Bd(AS_i,X) \\ 0 & \text{for } x \in U \setminus UPP(AS_i,X), \end{cases}$$

where $x \in U$.

Then we have for any $i \in \{1,\ldots,N\}$ and $x \in U$

$$vote_i(x) \in \{1, 0.5\} \text{ iff } x \in UPP(AS_i, X).$$

Now, we can define approximation of concepts relative to such generalized approximation spaces.

Definition 5.3. Let $GAS = (\mathscr{A}\mathscr{S}, vote)$ be a generalized approximation space. The *lower approximation* of $X \subseteq U$ (relative to *AS*) is defined by

$$LOW(GAS,X) = \{x \in U : vote(vote_1(x),\ldots,vote_i(x),\ldots,vote_N(x)) = 1\}.$$

The *upper approximation* of $X \subseteq U$ is defined by

$$UPP(GAS,X) = \{x \in U : vote(vote_1(x),\ldots,vote_i(x),\ldots,vote_N(x)) \in \{1, 0.5\}\}.$$

The *boundary region* of $X \subseteq U$ is defined by

$$Bd(GAS,X) = \{x \in U : vote(vote_1(x),\ldots,vote_i(x),\ldots,vote_N(x)) = 0.5\}.$$

From this definition we have:

$$U \setminus UPP(GAS,X) = \{x \in U : vote(vote_1(x),\ldots,vote_i(x),\ldots,vote_N(x)) = 0\},$$

and also:

$$Bd(GAS,X) = UPP(GAS,X) \setminus LOW(GAS,X)$$

and

$$LOW(GS,X) \subseteq UPP(GS,X).$$

However, in general, the following inclusions are not true:

$$LOW(GS,X) \subseteq X \subseteq UPP(GS,X).$$

Example 5.2. Let us consider an example of voting function satisfying $vote(k_1,\ldots,k_N) = 1$ iff for some i $k_i = 1$ and $vote(k_1,\ldots,k_N) \in \{1, 0.5\}$ iff $k_i \in \{1, 0.5\}$ for each i. One can check that for such voting function hold the following facts:

- $LOW(GAS,X) = \bigcup_{i=1}^{N} LOW(AS_i,X)$,
- $UPP(GAS,X) = \bigcap_{i=1}^{N} UPP(AS_i,X)$,

Many other cases of approximation spaces considered in papers on multi-granulation can be introduced using relevant voting functions. For example, one can consider the case from the mentioned paper by Rasiowa related to Problem 5.10.

One can ask for some 'natural' conditions that voting functions should satisfy. Examples of such conditions are as follows.

- $value(1,\ldots,1) = 1$.
- $value(1,\ldots,0) = 0$.
- For all $k_1,\ldots,k_N,k'_1,\ldots,k'_N \in \Sigma$

  (a) If $value(k_1,\ldots,k_N) = 1$ then for some $i \in \{1,\ldots,N\}$ $k_i = 1$.
  (b) If $value(k_1,\ldots,k_N) = 0$ then for some $i \in \{1,\ldots,N\}$ $k_i = 0$.
  (c) If $value(k_1,\ldots,k_N) = 1$ and $(k'_1,\ldots,k'_N)$ consists more occurrences of 1 than $(k_1,\ldots,k_N)$ then $value(k'_1,\ldots,k'_N) = 1$.

The reader is invited to elaborate in more detail this issue together with consequences of them on properties of approximations (see, *e.g.*, Problem 5.9).

It is worthwhile mentioning that different families of approximation spaces may be generated from a given decision system (decision table). A typical examples may concern ensembles of decision systems generated from a given decision systems using different (approximate) reducts (see, *e.g.*, [**?**, **?**, **?**, **?**]).

The discussed above case concerns the situation where the votes of approximation spaces are aggregated globally using voting functions. In real-life applications hierarchical aggregation of votes may be more efficient (see, e.g., [**?**, **?**]). In the case of [**?**] is used an ontology of concepts delivered by the domain expert. Next these concepts are approximated gradually from the lowest level up to the highest level. The characteristic functions of approximation regions of concepts from the lower level are added as new features to the higher level of the hierarchy. In the case of [**?**], using analogy to the brain hierarchical structure, it is suggested that the hierarchical structures are much more efficient. For illustration, let us consider a family of approximation spaces. For a given test case (object), each approximation space is returning a probability distribution over decision classes in the approximated classification. The aim is to aggregate these probability distributions into the resulting classification. In the aggregation process, one can first tray to discover clusters of similar probability distributions and next aggregate probability distributions corresponding to such clusters. The reader is asked to elaborate the details of this idea.

In the discussed example multi-relational approximation spaces from a given family of approximation spaces are returning their classifications of objects from the universe U concerning a given concept $X \subseteq U$. Another important case is considered in federated learning [**?**, **?**, **?**]. Federated learning is a machine learning approach where multiple entities collaborate in solving a machine learning problem, under the coordination of a central server or service provider. Each client's raw data is stored locally and not exchanged or transferred; instead, focused updates intended for immediate aggregation are used to achieve the learning objective [**?**]. Federated learning involves training statistical models over remote devices or siloed data centers (*e.g.*, mobile phones or hospitals), while keeping data localized. Training in heterogeneous and potentially massive networks introduces novel challenges that require a fundamental departure from standard approaches for large-scale machine learning, distributed optimization, and privacy-preserving data analysis [**?**]. Referring to our example, in federated learning the local agents equipped with multi-relational approximation spaces are submitting to the central agent constructed by them models and the central agent, *e.g.*, by aggregating them is able to return improved models to local agents having at the disposal the local multi-rational approximation spaces. The reader is invited to further develop the rough set based approach to problems and challenges of federated learning. The rough set approach grounded on the Interactive Granular Computing (IGrC) (see Chapter 12) model can be a good starting point in developing foundations for this approach.

In the next section, we discuss the information flow approach aiming to build the logical foundations for logic of distributed systems. We emphasize some relationships with the rough set approach.

Problem 5.1. Provide an example of concept X and attribute sets $A_1,\ldots,A_k$ justifying the claim formulated at the end of Example 5.1.

Problem 5.2. Show that $IND(a_r) = r$ and the indiscernibility relation $IND(A^*)$ of IS^* is equal to to

$$\bigcap_{r\in R} r.$$

Problem 5.3. Show that the indiscernibility relation $IND(A)$ of IS is invariant to renaming of attribute values.

Problem 5.4. Develop a heuristic supporting searching for the relevant partition for approximation of classification by the decision attribute from a given decision table with real value conditional attributes.

Problem 5.5. Develop an efficient heuristic for the information reduction problem in the case of fuzzy multi-relational approximation space.

Problem 5.6. Justify that the presented procedure for generation from a given decision system a family of nodes can be illustrated as a procedure generating a decision tree, *i.e.*, tree providing a procedure for decision making for objects represented by their signatures.

Problem 5.7. Check that the following equalities hold:

$$Bd(GAS,X) = UPP(GAS,X) \setminus LOW(GAS,X),$$

$$LOW(GS,X) \subseteq UPP(GS,X).$$

Problem 5.8. Provide and example showing that in general the following inclusions are not true:

$$LOW(GS,X) \subseteq X \subseteq UPP(GS,X).$$

Problem 5.9. If for all $k_1,\ldots,k_N \in \Sigma$ the voting function satisfies the following conditions:

$$\text{If } vote(k_1,\ldots,k_N) = 1 \text{ then } k_i = 1 \text{ for some } i$$

and

$$\text{If } vote(k_1,\ldots,k_N) = 0 \text{ then } k_i = 0 \text{ for some } i,$$

then

$$LOW(GAS,X) \subseteq X \subseteq UPP(GAS,X).$$

Problem 5.10. Develop a voting function for which the following properties hold:

- $LOW(GAS,X) = \bigcap_{i=1}^{N} LOW(AS_i,X)$,
- $LOW(GAS,U \setminus X) = \bigcup_{i=1}^{N} LOW(AS_i,U \setminus X)$.

Please present for the designed voting function an intuitive interpretation of the boundary region.

Problem 5.11. Develop a concept of voting function for non-binary classifications, *i.e.*, decision systems with more than two decision classes. Consider links with the Dempster-Shafer theory of evidence (see, [**?**] and Section 10.1 in Chapter 10).

Problem 5.12. Generalize the discussed above multi-relational approach for generalized approximation spaces with fuzzy equivalence relations (see [**?**] for definition of fuzzy equivalence relation).

Problem 5.13. Develop methods for feature selection based on ensembles of reducts.

5.2 Rough sets and information flow

This chapter can be treated as a short introduction to study of possible applications of the rough set approach to reasoning in distributed networks.

One of the main aims of the information flow approach due to Barwise and Seligman [**?**] was to develop a logic for a distributed system of agents equipped with local logics and linked by constraints represented by so called infomorphisms. In this section, we outline some relationships of the rough set approach with the information flow approach.

In [**?**] is discussed an approach to distributed decision making where different agents assigned to different nodes of a network may gain information from an event occurred in a remote node and make decision in a distributed fashion. For any agent ag from a given set of agents Ag is assigned a classification Cl_{ag} representing the information base of an agent. The flow of information between two agents is formalized by the notion of infomorphism.

Definition 5.4. A classification $Cl_{ag} = \langle Tok(ag), Typ(ag), \models_{ag} \rangle$ consists of

(i) $Tok(ag)$ – a set of objects to be classified, called tokens of Cl_{ag},
(ii) $Typ(ag)$ – a set of properties used to classify the tokens, called the types of Cl_{ag}, and
(iii) $\models_{ag} \subseteq Tok(ag) \times Typ(ag)$ – a binary relation between $Tok(ag)$ and $Typ(ag)$.

If $a \models_{ag} \alpha$, then a is said to be of type α in Cl_{ag}. That is, $\models_{ag}$ basically specifies which token is of which type.

Let us observe that any information system $\mathbb{A} = (U, At)$ defines a classification

$$Cl(\mathbb{A}) = (U, \{a = v : a \in At \text{ and } v \in V_a\}, \models_{Cl(\mathbb{A})}),$$

where $u \models_{Cl(\mathbb{A})} a = v$ iff $a(u) = v$ for $a \in At$, $v \in V_a$ and $u \in U$.

One can also consider as the set of types the set $Bool(\mathbb{A})$ of all Boolean combinations of descriptors of the form $a = v$, where $a \in At$, $v \in V_a$ and define another classification

$$ClB(\mathbb{A}) = (U, Bool(\mathbb{A}), \models_{ClB(\mathbb{A})}),$$

where $u \models_{ClB(\mathbb{A})} \alpha$ iff $u \in \|\alpha\|_{\mathbb{A}}$ for $\alpha \in Bool(\mathbb{A})$ and $u \in U$.

Any pair (Γ, Δ) of subsets of a given set of types Typ is called Genzen sequent over Typ. For a given classification $Cl_{ag} = \langle Tok(ag), Typ(ag), \models_{ag} \rangle$, and a token $u \in Tok(ag)$, the sequent (Γ, Δ) over $Typ(ag)$ is true at u, what is denoted by

$$u \models_{ag} (\Gamma, \Delta)$$

iff from the truth of the sentence:

$$u \models_{ag} \gamma \text{ for all } \gamma \in \Gamma$$

it follows the truth of the sentence:

$$u \models_{ag} \delta \text{ for some } \delta \in \Delta.$$

The sequent (Γ, Δ) is true in a given classification Cl_{ag}, what is denoted by $Clag \models_{ag} (\Gamma, \Delta)$ iff (Γ, Δ) is true in all tokens from $Tok(ag)$. The theory of classification $Clag$, denoted by $Th(ag)$, is the set of all sequents over $Typ(ag)$ true in $Clag$, *i.e.*,

$$Th(Cl_{ag}) = \{(\Gamma, \Delta) : \ Cl_{ag} \models_{ag} (\Gamma, \Delta), \text{ where } \Gamma, \Delta \subseteq Typ(ag)\}).$$

One can observe that elements of $Th(Cl(\mathbb{A}))$ correspond to nondeterministic rules (consisting of conjunction of descriptors on the left hand side and disjunction of descriptors on the right hand side) true in the information system $\mathbb{A}$.

For given two agents and their respective classifications a notion of infomorphism between two classifications is defined in the following way.

Definition 5.5. Let $Cl = \langle Tok(Cl), Typ(Cl), \models_{ClA} \rangle$ and $Cl' = \langle Tok(Cl'), Typ(Cl'), \models_{Cl'} \rangle$ be two classifications. An infomorphism $f : CL \rightleftarrows Cl'$ from Cl to Cl' is a contravariant pair of functions $f = (\hat{f}, \check{f})$ such that $\hat{f} : Typ(ClA) \mapsto Typ(Cl')$ and $\check{f} : Tok(Cl') \mapsto Tok(Cl)$ satisfying the following fundamental property of infomorphism:

$$\check{f}(b) \models_{Cl} \alpha \text{ iff } b \models_{Cl'} \hat{f}(\alpha)$$

for each $b \in Tok(Cl')$ and $\alpha \in Typ(Cl)$.

The notion of infomorphism thus allows the information to flow from one agent to another.

From the definition of infomorphism it follows that if $\mathbb{A}, \mathbb{B}$ are information systems and

$$(\hat{f}, \check{f}) : Cl(\mathbb{A}) \rightleftarrows CL(\mathbb{B})$$

then

$$a(\check{f}(u')) = v \text{ iff } a'(u') = v', \text{ where } (a' = v') = \hat{f}(a = v),$$

for any token u' of $Cl(\mathbb{B})$, any attribute a of $Cl(\mathbb{A})$, and $v \in V_a$ (see Figure 5.4).

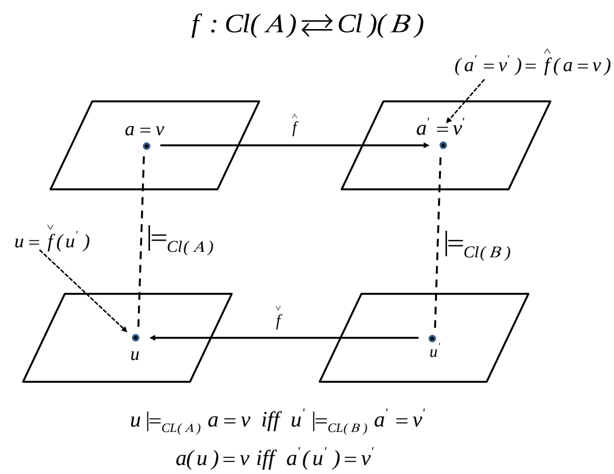


Fig. 5.4 Infomorphism between $Cl(\mathbb{A})$ and $Cl(\mathbb{B})$.

Based on the notion of classification of an agent ag, a notion of local logic has been defined as

$$(Tok(ag), Typ(ag), \models_{ag}, \vdash_{ag}, N_{ag}),$$

where the classification of ag, *i.e.*, $Cl(ag) = (Tok(ag), Typ(ag), \models_{ag})$ represents the agent's knowledge-base, a theory $(Typ(ag), \vdash_{ag},)$ where $\vdash_{ag} \subseteq 2^{Typ(ag)} \times 2^{Ty(ag)}$, presents the agent's reasoning base, and N_{ag} presents some objects/situations from $Tok(ag)$ of $Cl(ag)$ which supports the agent's theory.

Next, the definition of regular theory is provided is parallel to the notion of deductive logical consequence

Definition 5.6. A theory $T = (\Sigma, \vdash)$ is regular if it satisfies the properties of identity, weakening, and global cut for all types α, and all set $\Gamma, \Gamma', \Delta, \Delta', \Sigma', \Sigma_0, \Sigma_1$ of types.
Identity $\quad \alpha \vdash \alpha$
Weakening $\quad$ If $\Gamma \vdash \Delta$, then $\Gamma, \Gamma' \vdash \Delta, \Delta'$.
Global cut $\quad$ If $\Gamma, \Sigma_0 \vdash \Delta, \Sigma_1$ for each partition $\langle \Sigma_0, \Sigma_1 \rangle$ of Σ', then $\Gamma \vdash \Delta$.

Definition 5.7. Given two theories $T_1 = (Typ(T_1), \vdash_{T_1})$ and $T_2 = (Typ(T_2), \vdash_{T_2})$, a (regular theory) interpretation $f : T_1 \mapsto T_2$ of T_1 in T_2 is a function from $Typ(T_1)$ to $Typ(T_2)$ such that for each $\Gamma, \Delta \subseteq Typ(T_1)$ if $\Gamma \vdash_{T_1} \Delta$, then $f(\Gamma) \vdash_{T_2} f(\Delta)$.

The notion of local logic $\mathscr{L}_{ag}$ assigned to $ag \in Ag$ combines the idea of a classification together with that of a regular theory. Local logic basically formalizes the reasoning counterpart of an agent ag. Moreover, the concept of normal tokens also allows an agent to have reasonable but unsound inferences as there might be some sequents in the agent's theory which are not satisfied by every token.

Definition 5.8. A local logic $\mathscr{L}_{ag} = (Tok(ag), Typ(ag), \models_{ag}, \vdash_{ag}, N_{ag})$ of agent $ag \in Ag$ consists of
(i) a classification $Cl(ag) = (Tok(ag), Typ(ag), \models_{ag})$,
(ii) a regular theory $Th(\mathscr{L}_{ag}) = (Typ(ag), \vdash_{ag})$, and
(iii) a subset $N_A \subseteq Tok(ag)$, called the normal tokens of $\mathscr{L}_{ag}$, which satisfies all the constraints of $Th(\mathscr{L}_{ag})$.

In the case of information system $\mathbb{A}$ we use the following notation for the local logic corresponding to it:

$$\mathscr{L}_{\mathbb{A}} = (Tok(\mathbb{A}), Typ(\mathbb{A}), \models_{\mathbb{A}}, \vdash_{\mathbb{A}}, N_{\mathbb{A}}).$$

Let us assume that $f: \ Cl(\mathbb{A}) \rightleftarrows CL(\mathbb{B})$ for information systems $\mathbb{A}, \mathbb{B}$ and let us consider the following rule:

$$f - intro : \frac{\Gamma^{-f} \vdash_{\mathbb{A}} \Delta^{-f}}{\Gamma \vdash_{\mathbb{B}} \Delta}, \tag{5.1}$$

where Γ, Δ are subsets of $typ(\mathbb{B})$ and $Y^{-f} = \{\gamma \in Typ(\mathbb{A}) : \hat{f}(\gamma) \in Y\}$ for $Y \subseteq Typ(\mathbb{B})$. One can prove that this rule preserves validity, *i.e.*, if $(\Gamma^{-f}, \Delta^{-f})$ is true in $Cl(\mathbb{A})$ then (Γ, Δ) is true in $Cl(\mathbb{B})$.

Under the assumptions as above we consider the following rule:

$$f - elim : \frac{\Gamma^{f} \vdash_{\mathbb{B}} \Delta^{f}}{\Gamma \vdash_{\mathbb{A}} \Delta}, \tag{5.2}$$

where Γ, Δ are subsets of $typ(\mathbb{A})$ and $X^f = \{\hat{f}(\gamma) : \gamma \in X\}$ for $X \subseteq Typ(\mathbb{A})$. One can prove that this rule does not preserve validity.

The above presented rules are examples of rules allowing the agents to reason about validity of some rules in other agents linked to them by infomorphisms.

The central notion of information flow is the logical infomorphism, which links two agents and makes it possible to reason in a distributed network about information flow from one agent to the other.

Definition 5.9. A logic infomormhism $f : \mathscr{L}_1 \rightleftarrows \mathscr{L}_2$ consists of a contravariant pair $f = (\hat{f}, \check{f})$ of functions such that
(i) $f : cl(\mathscr{L}_1) \rightleftarrows cl(\mathscr{L}_2)$ is an infomorphism of classifications,

(ii) $\hat{f} : Th(\mathscr{L}_1) \mapsto Th(\mathscr{L}_2)$ is a theory interpretation, and
(iii) $\check{f}(N_{\mathscr{L}_2}) \subseteq N_{\mathscr{L}_1}$.

The central problem solved in the book [**?**] is related to the question how to construct a global logic representing in 'the best way' reasoning in the network of agents equipped with local logics and linked by different infomorphisms.

Problem 5.14. How well the concept of infomorphism fits to your understanding of concept of communication?

Problem 5.15. Is it possible for any information systems $\mathbb{A}, \mathbb{B}$ to extend the isomorphism $f : CL(\mathbb{A}) \rightleftarrows Cl(\mathbb{B})$ to isomorphism $f^* : CLB(\mathbb{A}) \rightleftarrows ClB(\mathbb{B})$?

Problem 5.16. Show that elements of $Th(Cl(\mathbb{A}))$ correspond to nondeterministic rules (consisting of conjunction of descriptors on the left hand side and disjunction of descriptors on the right hand side) true in the information system $\mathbb{A}$.

Problem 5.17. Develop methods for aggregation of information systems based on the concepts of classification and isomorphism.

Problem 5.18. Prove that the $f - intro$ rule from Eqn. 5.1 preserves the validity.

Problem 5.19. Prove that the $f - elim$ rule from Eqn. 5.2 does not preserve the validity.

Problem 5.20. Let us consider a set of agents Ag organised into a distributed network with agents ag_i labelled by classifications defined by some information systems and linked by infomporphisms between assigned to them classifications. Is it possible to construct an information system representing globally this network?

Problem 5.21. Let us consider a set of agents $Ag = \{ag_1, ag_2, ag_3\}$ organised into a distributed network with three nodes labelled by local logic of agents from Ag and linked by some infomorphisms. Can you design 'the best logic' in your opinion representing reasoning in this distributed network?
Hint: See the main theorem in [**?**].

5.3 Process Mining

Mining temporal or complex data streams is on the agenda of many research centers and companies worldwide [**?**, **?**, **?**, **?**, **?**]. In the data mining community, there is a rapidly growing interest in developing methods for process mining, *e.g.*, for discovery of structures of temporal processes from observations (recorded data). Works on process mining [**?**, **?**, **?**, **?**, **?**, **?**, **?**, **?**, **?**, **?**] have been undertaken by many renowned centers worldwide[1]. This research is also related to functional data analysis [**?**] , cognitive networks [**?**] , and dynamical system modeling in biology [**?**].

[1] `http://www.isle.org/~langley/`, `http://soc.web.cse.unsw.edu.au/bibliography/discovery/index.html`, `http://www.processmining.org/`

Table 5.1 The information system $U,A)$ considered in the example [**?**].

U/A	a	b	c
u_1	1	1	0
u_2	0	2	0
u_3	0	0	2

Process mining is aiming to extract process models out of event logs, to monitor deviations via comparison of models as well as logs of events, describing the structure of organization, simulation models building in an automated manner, extension of models as well as retrieval, to forecast the behavior of a process so that a suggestion list may be built on the behalf of history of processes. Primarily it involves extracting process models from the event logs [**?**].

There are many papers on discovery of concurrent processes from data based on rough sets [**?**, **?**]. This research was initiated by the idea of Professor Zdzisław Pawlak [**?**]. Data tables representing information systems are treated as partial specifications of concurrent processes. Rows in a given data table are examples of global states and columns represent local processes. Values of attributed are interpreted as states of local processes. Different data tables may represent different logs. The order in which objects appear in the data table can be interpreted as the order in time they are perceived.

For illustrating the idea by Pawlak, let us consider a simple example from [**?**] of a system of traffic lights (see Figure 5.5) for a T-intersection.

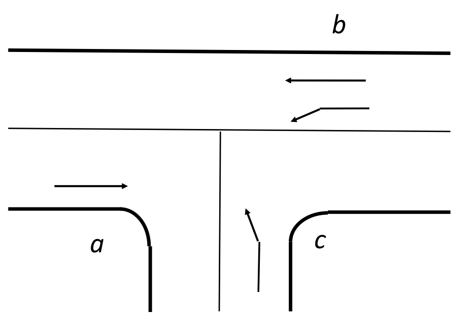


Fig. 5.5 T-intersection [**?**].

The information systems presented in Table 5.1 is characterizing the system specification.

In this information system (U,A) attributes a,b,c from A denote the traffic signals, objects u_1,u_2,u_3 from U denote the possible states of the system, and values 0,1,2 denote colours of the traffic lights, red, green, and green arrow, respectively. The attributes a,b,c can be interpreted as local processes (agents) and the values 0,1,2 as the local states of these processes. These local agents should be coordinated to obey some constraints represented by the relevant driving rules for providing the correct control of lights. These rules can be extracted from data table with objects representing global states (situations) of the discussed crossing system. In the illustrative example, the behavior of the global system is interpreted as behaviour of system controlling changes of lights on the crossing. It is assumed that each local process is cyclic,*e.g.*, changes of local process c are described by the sequence $0,2,0,2,\ldots$.

The row corresponding to object u_1 consists of vector of values (local states) 1,1,0 of attributes (local processes) a,b,c, respectively and is characterising the configuration of traffic lights where at a and b are green lights and at c – red light (*i.e.*, value 1 of attributes a and b indicate green light and value 0 of c – red light).

The information system (U,A) is describing the set of all correct configurations of lights in the considered example.

Let us observe, that in the system (U,A) the following rule obtained from the row corresponding to u_3 holds:

$$a = 0 \wedge b = 0 \Longrightarrow c = 2.$$

This rule has an intuitive interpretation:

red lights at a & *red lights at b open a possibility to turn left from c*.

This rule is not minimal but dropping the condition $a = 0$ from the left hand side of this rule leads to a minimal rule $b = 0 \Longrightarrow c = 2$. Another minimal rule obtained from the first row is: $b = 0 \Longrightarrow a = 0$.

For the considered table we have the following minimal rules: $b = 0 \Longrightarrow c = 2$, $b = 0 \Longrightarrow a = 0$, $b = 2 \Longrightarrow a = 0$, $b = 2 \Longrightarrow c = 0$, $b = 1 \Longrightarrow a = 1$, $b = 1 \Longrightarrow c = 0$, $a = 1 \wedge c = 0 \Longrightarrow b = 1$, $a = 0 \wedge c = 0 \Longrightarrow b = 2$, $a = 0 \wedge c = 2 \Longrightarrow b = 0$. One can check that they correspond to the reducts $\{b\}$ and $\{a,c\}$ of the information system represented by the considered table. Any situation (objects) from the table is satisfying all these rules – is consistent with these rules. In the considered example, these are all situations characterized by signatures over attributes a,b,c with values in $\{0,1\}$, $\{0,1,2\}$ and $\{0,2\}$, respectively and consistent with these rules.

In general, one can consider the set of all minimal rules true in (U,A) and define as admissible situations all situations described by vectors of values from $\{0,1\} \times \{0,1,2\} \times \{0,2\}$, consistent with this set of rules (*i.e.*, situations in which all such minimal rules are true). On the basis of this set of rules it is possible to constrcut a concurrent model (in the form of Petri net) synchronizing the local processes in such a way that global states appearing in its behaviour, called reachable states, are exactly the admissible situations [**?**].

One of the solutions for discovery of process models from data presented in the above cited papers was based on decomposition of data tables into modules defined by reducts of data tables (see, *e.g.*, [**?**]). The modules are linked by constraints defined by rules extracted from data. In another approach, first from a given data table decision rules are extracted (*e.g.*, a set of minimal decision rules) and such a set of decision rules is used as knowledge encoded in the data table or theory defined by data table. Next, the set of all global states is defined as equal to the maximal set of objects (global states) consistent with the theory. There were proposed methods for automatic generation from a given data table a (colored) Petri net with the reachability set equal to the maximal consistent set of states consistent with the theory generated from the data table. The reader is referred to [**?**, **?**] for information on the developed software (ROSECON) for inducing Petri nets from data tables. An important role in discovering Petri nets play the inhibitory rules [**?**]. The reader interested in complexity results related to such rules is referred to [**?**].

The presented approach can be extended by considering, *e.g.*: (ii) specification by data table (information system) as a partial representation of perceived processes in different moments of time, (ii) necessity od developing adaptive strategies for constructed data models, (ii) structural objects in considered data tables (situations)represented by time series or multi-time series, (ii) many data tables (information systems) representing different perceived processes, (iii) different methods of aggregation of data tables (information systems) leading to hierarchical learning of process models, (iv) a partial specification by data tables (information systems) changes in perceived situation, *e.g.*, by approximation transitions between perceived states, (v) necessity of dialogues with domain experts for better understanding the perceived situation. Below we are adding some comments on these issues.

In general, a specification of the concurrent system by information systems may be partial and one should develop some adaptive strategies to modify the current generators of such sets by expending the set of current situations by a new ones which should be added. On the basis of the proposed by Pawlak partial specification of concurrent processes and developed methods it was already possible to develop some of such strategies. However, much more should be done.

There are a lot of challenges related to learning models of concurrent processes from data tables occurring in hierarchical modeling which should often lead to the high quality of approximation of very complex vague concepts. This can be well illustrated by the following citation [**?**]:

> [...] *One of the fascinating goals of natural computing is to understand, in terms of information processing, the functioning of a living cell. An important step in this direction is understanding of interactions between biochemical reactions.] [... the functioning of a living cell is determined by interactions of a huge number of biochemical reactions that take place in living cells.*

On higher levels of hierarchy, the structure of objects becomes complex, *e.g.*, indiscernibility classes of data tables considered on higher level of the hierarchy can be equal to sets of paths of structural states. The theories of such data tables are much more complex than before. The rules in such a theory discovered from data may require extension to (spatio-)temporal decision rules or temporal association rules or even more complex rules defined by different temporal logics. The challenges are related to discovery of relevant rules and theories over such rules as well as to inducing, *e.g.*, Petri nets consistent with theories defined by such constraints. In real-life application hierarchical learning requires support from domain experts (see, *e.g.*, [**?**, **?**, **?**]). Further progress is needed in developing methods based on reasoning about changes [**?**, **?**].

It is worthwhile to mention that in process mining we are aiming at discovery of complex granules (see Chapter 12) representing processes and their properties. The granules leading to such granules may have complex structure. For example, they can represent properties of (i) time windows with time points (or more complex structures) labelled by value vectors of sensory attributes (or attributes over more complex granules already defined), (ii) similarity classes of already constructed granules, (iii) aggregations of more complex granules over already constructed granules, (iv) complex objects such as (parameterised) learning algorithms etc. One should note the importance of reasoning methods supporting searching discovery of relevant granules what requires, in particular discovery of relational structures over already defined granules as well as languages of formulas for expressing their properties. This is contrary to methods usually used in mathematical logic (see, *e.g.*, [**?**, **?**, **?**]), where these relational structures, language of formulas and the satisfiability relation establishing their semantics over relational structures are assumed to be given.

Let us consider an illustrative example explaining motivation for discovery of process models from data. In the example, models of concurrent processes are in the form of Petri Nets. However, one can look for other models, *e.g.*, in the form of differential equations [**?**].

This problem is illustrated in Figure 5.6. It is assumed that from granules G, G_1, G_2 representing the sets of the paths of the processes, their models in the form of *e.g.*, Petri nets PN, PN_1, PN_2, respectively, were induced. Then, the structure of interaction between PN_1 and PN_2 can be described by an operation transforming PN_1, PN_2 into PN.

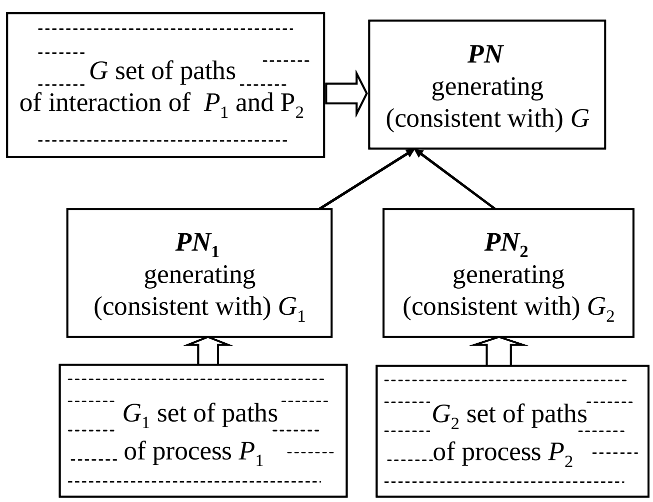


Fig. 5.6 Discovery of interaction structure

The discovery of relevant attributes on each level of the hierarchy can be supported by domain knowledge (see, *e.g.*, [**?**, **?**, **?**]) provided *e.g.*, by concept ontology together with the illustration of concepts by means of the samples of objects taken from this concepts and their complements [**?**]. Such application of domain knowledge often taken from human experts serves as another example of the interaction of a system (classifier) with its environment. Additionally, such support of relevant attributes discovery on given level of the hierarchy, as well as on other levels, can be found using different ontologies. These ontologies can be described by different sets of formulas and possibly by different logics. Thus, the description of such discovery of relevant attributes in interaction, as well as its support give a good reason for applying fibring logics methods [**?**]. Note that in the hierarchical modeling of relevant complex patterns also top-down interactions of the higher levels of the hierarchy with the lower levels should be considered, *e.g.*, if the patterns constructed on higher levels are not relevant for the target task, the top-down interaction should inform lower levels about the necessity of searching for new patterns. The key role in this process is played by information granulation [**?**, **?**]. In papers [**?**, **?**] is outlined an approach to discovery of processes from data and domain knowledge which is based on Rough Granular Computing philosophy. This is related to the domain of Process Intelligence [**?**].

Problem 5.22. Please, justify that the system (U,A) is representing all correct situations of traffic lights at the considered T-intersection.

Problem 5.23. In the considered example the set of traffic light situations consistent with the set of minimal rules true in (U,A) is equal to U. Is this true in general?

One of the problem considered in [**?**] is related to construction of Petri nets generating sets of configurations consistent with a given set of minimal rules.

Problem 5.24. Please construct the minimal set of rules of (U,A) consistent with U.

Problem 5.25. *Research problem.*
Develop a method for process mining based on an extension of the mentioned above approach by Pawlak assuming that the values of attributes are also representing information about the control of local processes as well as some behavioral constraints. Moreover, it is assumed that changes of local states in a given local process may cause changes of local states of some local processes which are defined as neighbours of this local process.

Problem 5.26. *Research problem.*
Develop a method for the above problem combining it with the idea of AlphaGo[2].

[2] see, *e.g.*, `https://deepmind.com/research/case-studies/alphago-the-story-so-far`)

References

1. Aggarwal, C.: Data Streams: Models and Algorithms. Springer, Berlin (2007). doi:10.1007/978-0-387-47534-9
2. Alam, M., Shakil, K.A., Khan, S. (eds.): Internet of Things (IoT): Concepts and Applications. Springer, Cham, Switzerland (2020). doi:10.1007/978-3-030-37468-6
3. Barwise, J., Seligman, J.: Information Flow: The Logic of Distributed Systems. Cambridge University Press (1997). doi:10.1023/A:1008350722315
4. Bazan, J.: Hierarchical classifiers for complex spatio-temporal concepts. In: J.F. Peters, A. Skowron, H. Rybiński (eds.) Transactions on Rough Sets IX: Journal Subline, *Lecture Notes in Computer Science*, vol. 5390, pp. 474–750. Springer, Heidelberg (2008). doi:10.1007/978-3-540-89876-4_26
5. Borrett, S.R., Bridewell, W., Langely, P., Arrigo, K.R.: A method for representing and developing process models. Ecological Complexity **4**, 1–12 (2007). doi:10.1016/j.ecocom.2007.02.017
6. Buyya, R., Srirama, S.N. (eds.): Fog and Edge Computing. Principles and Paradigms. John Wiley & Sons, Hoboken, NJ 07030 (2019). doi:10.1002/9781119525080
7. Cattaneo, G.: Abstract approximation spaces for rough theories. In: L. Polkowski, A. Skowron (eds.) Rough Sets in Knowledge Discovery 1: Methodology and Applications, *Studies in Fuzziness and Soft Computing*, vol. 18, pp. 59–98. Physica-Verlag, Heidelberg (1998)
8. Clark, P.G., Grzymała-Busse, J.W., Rząsa, W.: Generalizations of approximations. In: P. Lingras, M. Wolski, C. Cornelis, S. Mitra, P. Wasilewski (eds.) Rough Sets and Knowledge Technology, pp. 41–52. Springer, Berlin, Heidelberg (2013). doi:10.1007/978-3-642-41299-8_5
9. D'eer, L., Cornelis, C., Yao, Y.: A semantically sound approach to Pawlak rough sets and covering-based rough sets. International Journal of Approximate Reasoning **78**, 62–72 (2016). doi:10.1016/J.IJAR.2016.06.013
10. Delimata, P., Moshkov, M.J., Skowron, A., Suraj, Z.: Inhibitory Rules in Data Analysis: A Rough Set Approach, *Studies in Computational Intelligence*, vol. 163. Springer, Heidelberg (2009). doi:10.1007/978-3-540-85638-2
11. Dubois, D., Prade, H.: Rough fuzzy sets and fuzzy rough sets. International Journal of General Systems **17**, 191–209 (1990). doi:10.1080/03081079008935107
12. Durdymyradov, K., Moshkov, M., Ostonov, A.: Decision Trees Versus Systems of Decision Rules. A Rough Set Approach, *Studies in Big Data*, vol. 160. Springer, Cham (2025)
13. Ehrenfeucht, A., Rozenberg, G.: Reaction systems: A model of computation inspired by biochemistry. In: Y. Gao, H. Lu, S. Seki, S. Yu (eds.) Developments in Language Theory, 14th International Conference, DLT 2010, London, ON, Canada, August 17-20, 2010. Proceedings, *Lecture Notes in Computer Science*, vol. 6224, pp. 1–3. Springer (2010). doi:10.1007/978-3-642-14455-4_1. URL `https://doi.org/10.1007/978-3-642-14455-4_1`
14. Feng, J., Jost, J., Qian, M. (eds.): Network: From Biology to Theory. Springer, Berlin (2007). doi:10.1007/978-1-84628-780-0
15. Fernandez-Llatas, C. (ed.): Interactive Process Mining in Healthcare. Springer Nature, Cham (2021). doi:10.1007/978-3-030-53993-1
16. Gabbay, D. (ed.): Fibring Logics. Oxford University Press (1998)
17. Góra, G.: Combining instance-based learning and rule-based methods for imbalanced data. Phd thesis, University of Warsaw, Warsaw (2022). `https://www.mimuw.edu.pl/sites/default/files/gora_grzegorz_rozprawa_doktorska.pdf`
18. Góra, G., Wojna, A.G.: RIONA: A new classification system combining rule induction and instance-based learning. Fundamenta Informaticae **51**(4), 369–390 (2002)
19. Hastie, T., Tibshirani, R., Friedman, J.H.: The Elements of Statistical Learning: Data Mining, Inference, and Prediction. Springer-Verlag, Heidelberg (2009). doi:10.1007/b94608
20. Inuiguchi, M., Wu, W., Cornelis, C., Verbiest, N.: Fuzzy-rough hybridization. In: J. Kacprzyk, W. Pedrycz (eds.) Springer Handbook of Computational Intelligence, Springer Handbooks, pp. 425–451. Springer (2015). doi:10.1007/978-3-662-43505-2_26

21. Japkowicz, N., Shah, M.: Evaluating Learning Algorithms: A Classification Perspective. Cambridge University Press, New York, NY (2011)
22. Järvinen, J., Kovács, L., Radeleczki, S.: Rough sets defined by multiple relations. In: T. Mihálydeák, F. Min, G. Wang, M. Banerjee, I. Düntsch, Z. Suraj, D. Ciucci (eds.) Rough Sets - International Joint Conference, IJCRS 2019, Debrecen, Hungary, June 17-21, 2019, Proceedings, *Lecture Notes in Computer Science*, vol. 11499, pp. 40–51. Springer (2019). doi:10.1007/978-3-030-22815-6_4
23. Järvinen, J., Radeleczki, S.: The structure of multigranular rough sets. Fundamenta Informaticae **176**(1), 17–41 (2020). doi:10.3233/FI-2020-1961
24. Jensen, R., Shen, Q.: Computational Intelligence and Feature Selection: Rough and Fuzzy Approaches. IEEE Press Series on Cmputationa Intelligence. IEEE Press and John Wiley & Sons, Hoboken, NJ (2008)
25. Kacprzyk, J., Pedrycz, W.: Springer Handbook of Computational Intelligence. Springer, Heidelberg (2015). doi:10.1007/978-3-662-43505-2
26. Kairouz, P., McMahan, H.B., Avent, B., Bellet, A., et al: Advances and Open Problems in Federated Learning. Now Foundations and Trends, IEEE (2021)
27. Khan, M.A., Banerjee, M.: Formal reasoning with rough sets in multiple-source approximation systems. International Journal of Approximate Reasoning **49**(2), 466–477 (2008). doi:10.1016/j.ijar.2008.04.005
28. Li, T., Sahu, A., Talwalkar, A., Smith, V.: Federated learning: Challenges, methods, and future directions. IEEE Signal Processing Magazine **37**(3), 50–60 (2020). doi:10.1109/msp.2020.2975749
29. Liljeberg, A.P., Jantsch, J.S.P.A. (eds.): Fog Computing in the Internet of Things. Intelligence at the Edge. Springer, Cham, Switzerland (2018). doi:10.1007/978-3-319-57639-8
30. Lin, T.Y.: The discovery, analysis and representation of data dependencies in databases. In: L. Polkowski, A. Skowron (eds.) Rough Sets in Knowledge Discovery 1: Methodology and Applications, *Studies in Fuzziness and Soft Computing*, vol. 18, pp. 107–121. Physica-Verlag, Heidelberg (1998)
31. Ma, X., Zhan, J., Sun, B., Alcantud, J.C.R.: Novel classes of coverings based multigranulation fuzzy rough sets and corresponding applications to multiple attribute group decision-making. Artificial Intelligence Review **53**(8), 6197–6256 (2020). doi:10.1007/s10462-020-09846-1
32. Maji, P., Pal, S.K.: Rough-Fuzzy Pattern Recognition: Application in Bioinformatics and Medical Imaging. Wiley Series in Bioinformatics. John Wiley & Sons, Hoboken, NJ (2012)
33. Marquis, O., Papini, O., Prade, H. (eds.): A Guided Tour of Artificial Intelligence Research vol. 1. Knowledge Representation, Reasoning and Learning. Springer, Cham (2020). doi:10.1007/978-3-030-06164-7
34. Marquis, O., Papini, O., Prade, H. (eds.): A Guided Tour of Artificial Intelligence Research vol.2. AI Algorithms. Springer, Cham (2020). doi:10.1007/978-3-030-06167-8
35. Marquis, O., Papini, O., Prade, H. (eds.): A Guided Tour of Artificial Intelligence Research vol.3. Interfaces and Applications of AI. Springer, Cham (2020). doi:10.1007/978-3-030-06170-8
36. de Medeiros, A.K.A., Weijters, A.J.M.M., van der Aalst, W.M.P.: Genetic process mining: An experimental evaluation. Data Mining and Knowledge Discovery **14**, 245–304 (2007). doi:10.1007/s10618-006-0061-7
37. Monarch, R.M.: Human-in-the-Loop Machine Learning. Active Learning and Annotation for Human-Centered AI. MANNING, Shelter Island, NY (2021)
38. Moshkov, M.: Time and space complexity of deterministic and nondeterministic decision trees. Annals of Mathematics and Artificial Intelligence **91**(1), 45–74 (2023). doi:10.1007/S10472-022-09814-1
39. Moshkov, M., Skowron, A., Suraj, Z.: On testing membership to maximal consistent extensions of information systems. In: S. Greco, Y. Hata, S. Hirano, M. Inuiguchi, S. Miyamoto, H.S. Nguyen, R. Słowiński (eds.) Proceedings of the 5th International Conference on Rough Sets and Current Trends in Computing (RSCTC'2006), Kobe, Japan, November 6-8, 2006,

Lecture Notes in Artificial Intelligence, vol. 4259, pp. 85–90. Springer-Verlag, Heidelberg (2006). doi:10.1007/11908029_10
40. Moshkov, M., Skowron, A., Suraj, Z.: On irreducible descriptive sets of attributes for information systems. In: C.C. Chan, J.W. Grzymała-Busse, W. Ziarko (eds.) Proceedings of the 6th International Conference on Rough Sets and Current Trends in Computing (RSCTC'2008), Akron, OH, USA, October 23-25, 2008, *Lecture Notes in Artificial Intelligence*, vol. 5306, pp. 21–30. Springer-Verlag, Heidelberg (2008). doi:10.1007/978-3-540-88425-5
41. Moshkov, M.J., Skowron, A., Suraj, Z.: On minimal rule sets for almost all binary information systems. Fundamenta Informaticae **80**(1-3), 247–258 (2007)
42. Moshkov, M.J., Skowron, A., Suraj, Z.: On minimal inhibitory rules for almost all k-valued information systems. Fundamenta Informaticae **93**(1-3), 261–272 (2009). doi:10.3233/FI-2009-0101
43. Ngu, A.H., Gutierrez, M., Metsis, V., Nepal, S., Sheng, Q.Z.: Iot middleware: A survey on issues and enabling technologies. IEEE Internet Things Journal **41**(1), 1–20 (2017). doi:10.1109/JIOT.2016.2615180
44. Nguyen, H.S.: Approximate boolean reasoning: Foundations and applications in data mining. Transactions on Rough Sets V: Journal Subline **5**, 344–523 (2006). doi::10.1007/11847465_16
45. Nguyen, H.S., Jankowski, A., Skowron, A., Stepaniuk, J., Szczuka, M.: Discovery of process models from data and domain knowledge: A rough-granular approach. In: J.T. Yao (ed.) Novel Developments in Granular Computing: Applications for Advanced Human Reasoning and Soft Computation, pp. 16–47. IGI Global, Hershey, PA (2010)
46. Nguyen, H.S., Skowron, A.: A rough granular computing in discovery of process models from data and domain knowledge. Journal of Chongqing University of Post and Telecommunications **20**(3), 341–347 (2008)
47. Nguyen, H.S., Skowron, A.: Rough sets: From rudiments to challenges. In: A. Skowron, Z. Suraj (eds.) Rough Sets and Intelligent Systems. Professor Zdzislaw Pawlak in Memoriam, *Series Intelligent Systems Reference Library*, vol. 42, pp. 75–173. Springer, Heidelberg (2013). doi:10.1007/978-3-642-30344-9_3
48. North, J.S., Wikle, C.K., Schliep, E.M.: A review of data-driven discovery for dynamic systems. International Statistical Review **91**(3), 464–492 (2023). doi:10.1111/insr.12554
49. Pal, S.K., Peters, J.F. (eds.): Rough Fuzzy Image Analysis Foundations and Methodologies. Chapman & Hall/CRC, Boca Raton, Fl (2010)
50. Pal, S.K., Skowron, A. (eds.): Rough Fuzzy Hybridization: A New Trend in Decision-Making. Springer-Verlag, Singapore (1999)
51. Pancerz, K., Suraj, Z.: Modelling concurrent systems specified by dynamic information systems: A rough set approach. Electronic Notes in Theoretical Computer Science **82**(4), 206–218 (2003). doi:10.1016/S1571-0661(04)80719-9
52. Pancerz, K., Suraj, Z.: Discovering concurrent models from data tables with the ROSECON system. Fundamenta Informaticae **60**(1-4), 251–268 (2004)
53. Pancerz, K., Suraj, Z.: Discovery of asynchronous concurrent models from experimental tables. Fundamenta Informaticae **61**(2), 97–116 (2004)
54. Pancerz, K., Suraj, Z.: Restriction-based concurrent system design using the rough set formalism. Fundamenta Informaticae **67**(1-3), 233–247 (2005)
55. Pancerz, K., Suraj, Z.: Reconstruction of concurrent system models described by decomposed data tables. Fundamenta Informaticae **71**(1), 121–137 (2006)
56. Pancerz, K., Suraj, Z.: Towards efficient computing consistent and partially consistent extensions of information systems. Fundamenta Informaticae **79**(3-4), 553–566 (2007)
57. Papageorgiou, E.I., Stylios, C.D.: Fuzzy cognitive maps. In: W. Pedrycz, S. Skowron, V. Kreinovich (eds.) Handbook of Granular Computing, pp. 755–774. John Wiley & Sons, Hoboken, NJ (2008). doi:10.1002/9780470724163
58. Pawlak, Z.: Information systems - theoretical foundations. Information Systems **6**, 205–218 (1981). doi:10.1016/0306-4379(81)90023-5

59. Pawlak, Z.: Rough sets. International Journal of Computer and Information Sciences **11**, 341–356 (1982). doi:10.1007/BF01001956
60. Pawlak, Z.: Rough Sets: Theoretical Aspects of Reasoning about Data, *System Theory, Knowledge Engineering and Problem Solving*, vol. 9. Kluwer Academic Publishers, Dordrecht, The Netherlands (1991). doi:10.1007/978-94-011-3534-4
61. Pawlak, Z.: Concurrent versus sequential - the rough sets perspective. Bulletin of the EATCS **48**, 178–190 (1992)
62. Pawlak, Z., Skowron, A.: Rudiments of rough sets. Information Sciences **177**(1), 3–27 (2007). doi:10.1016/j.ins.2006.06.003
63. Peters, J.F., Skowron, A., Suraj, Z.: An application of rough set methods in control design. Fundamenta Informaticae **43**(1-4), 269–290 (2000). doi:10.3233/FI-2000-43123414
64. Peters, J.F., Skowron, A., Suraj, Z.: An application of rough set methods in control design. Fundamenta Informaticae **43**(1-4), 269–290 (2000). doi:10.3233/FI-2000-43123414
65. Poggio, T., Smale, S.: The mathematics of learning: Dealing with data. Notices of the AMS **50**(5), 537–544 (2003)
66. Polkowski, L., Skowron, A.: Rough mereological approach to knowledge- based distributed AI. In: J.K. Lee, J. Liebowitz, J.M. Chaee (eds.) Critical Technology, Proceedings of the Third World Congress on Expert Systems, February 5-9, Soeul, Korea, pp. 774–781. Cognizant Communication Corporation, New York (1996)
67. Qian, Y., Liang, J., Yao, Y., Dang, C.: MGRS: A multi-granulation rough sets. Information Sciences **180**, 949–970 (2010). doi:10.1016/j.ins.2009.11.023
68. Ramsay, J.O., Silverman, B.W.: Applied Functional Data Analysis. Springer, Berlin (2002). doi:10.1007/b98886
69. Rasiowa, H.: Mechanical proof systems for logic: Reaching consensus by groups of intelligent agents. International Journal of Approximate Reasoning **5**(4), 415–432 (1991). doi:10.1016/0888-613X(91)90020-M
70. Rasiowa, H., Marek, V.W.: Mechanical proof systems for logic ii, consensus programs and their procedding (extended abstract). In: H.J. Komorowski, Z.W. Ras (eds.) Methodologies for Intelligent Systems, 7th International Symposium, ISMIS '93, Trondheim, Norway, June 15-18, 1993, Proceedings, *Lecture Notes in Computer Science*, vol. 689, pp. 142–151. Springer (1993). doi:10.1007/3-540-56804-2_14
71. Rasiowa, H., Marek, V.W.: Mechanical proof systems for logic ii, consensus programs and their processing. Journal of Intelligent Information Systems **2**(2), 149–164 (1993). doi:10.1007/BF00965875
72. Rauszer, C.: Rough logic for multi-agent systems. In: M. Masuch, L. Pólos (eds.) Knowledge Representation and Reasoning Under Uncertainty, Logic at Work [International Conference Logic at Work, Amsterdam, The Netherlands, December 17-19, 1992], *Lecture Notes in Computer Science*, vol. 808, pp. 161–181. Springer (1992). doi:10.1007/3-540-58095-6_12
73. Reinkemeyer, L. (ed.): Process Mining in Action Principles. Use Cases and Outlook. Springer, Cham (2020). doi:10.1007/978-3-030-40172-6
74. Reinkemeyer, L. (ed.): Process Intelligence in Action. Taking Process Mining to the Next Level. Springer Nature, Cham (2024). doi:10.1007/978-3-031-61343-2
75. Rissanen, J.: Minimum-description-length principle. In: S. Kotz, N. Johnson (eds.) Encyclopedia of Statistical Sciences, pp. 523–527. John Wiley & Sons, New York, NY (1985)
76. Roddick, J.F., Hornsby, K., Spiliopoulou, M.: An updated bibliography of temporal, spatial, and spatio-temporal data mining research. In: J.F. Roddick, K. Hornsby (eds.) First International WorkshopTemporal, Spatial, and Spatio-Temporal Data Mining (TSDM 2000) Lyon, France, September 12, 2000, Revised Papers, *Lecture Notes in Computer Science*, vol. 2007, pp. 147–164. Springer, Heidelberg (2001). doi:10.1007/3-540-45244-3_12
77. Rokach, L.: Ensemble Learning: Pattern Classification Using Ensemble Methods, *Machine Perception and Artificial Intelligence*, vol. 85. World Scientific, Singapore (2019). (2nd edition)
78. Rokach, L., Maimon, O. (eds.): Data Mining with Decision Trees: Theory and Applications, *Series in Machine Perception and Artificial Intelligence*, vol. 69. World Scientific, Singapore (2008)

79. Russell, S.J., Norvig, P.: Artificial Intelligence A Modern Approach. Pearson Education, Inc., Upper Saddle River, NJ (2021). 4th edition
80. de Saint-Cyr, F.D., Herzig, A., Lang, J., Marquis, P.: Reasoning about action and change. In: O. Marquis, O. Papini, H. Prade (eds.) A Guided Tour of Artificial Intelligence Research. Knowledge Representation, Reasoning and Learning, pp. 487–518. Springer, Cham (2020). doi:10.1007/978-3-030-06164-7_15
81. Shafer, G.: Mathematical theory of evidence. Princeton University Press, Princeton, New Jersey (1976)
82. Shneiderman, B.: Human – Centered AI. Oxford University Press, Oxford, UK (2022). doi:10.1093/oso/9780192845290.001.0001
83. Shoham, Y.: Reasoning about change: Time and causation from the standpoint of artificial intelligence. MIT Press, Cambridge, MA (1988)
84. Skowron, A.: Rough–fuzzy computing. In: G. Rozenberg, T. Bäck, J.N. Kok (eds.) Handbook of Natural Computing, pp. 1921–1948. Springer Berlin Heidelberg, Berlin, Heidelberg (2012). doi:10.1007/978-3-540-92910-9_57
85. Skowron, A., Ślęzak, D.: Rough sets turn 40: From information systems to intelligent systems. In: M. Ganzha, L.A. Maciaszek, M. Paprzycki, D. Ślęzak (eds.) Proceedings of the 17th Conference on Computer Science and Intelligence Systems, FedCSIS 2022, Sofia, Bulgaria, September 4-7, 2022, *Annals of Computer Science and Information Systems*, vol. 30, pp. 23–34 (2022). doi:10.15439/2022F310
86. Skowron, A., Ślęzak, D.: A rough set perspective on intelligence systems (2024). (submitted)
87. Skowron, A., Stepaniuk, J.: Tolerance approximation spaces. Fundam. Informaticae **27**(2/3), 245–253 (1996). doi:10.3233/FI-1996-272311
88. Skowron, A., Stepaniuk, J.: Information granules: Towards foundations of granular computing. International Journal of Intelligent Systems **16**(1), 57–86 (2001). doi:10.1002/1098-111X(200101)16:1<57::AID-INT6>3.0.CO;2-Y
89. Skowron, A., Stepaniuk, J.: Toward rough set based insightful reasoning in intelligent systems p. 2 (2024). (submitted)
90. Skowron, A., Stepaniuk, J., Peters, J.F.: Rough Sets and Infomorphisms: Towards Approximation of Relations in Distributed Environments. Fundamenta Informaticae **54**(2-3), 263–277 (2003)
91. Skowron, A., Suraj, Z.: A rough set approach to real-time state identification. Bulletin of the EATCS **50**, 264–275 (1993)
92. Skowron, A., Suraj, Z.: Rough sets and concurrency. Bulletin of the Polish Academy of Sciences, Technical Sciences **41**, 237–254 (1993)
93. Skowron, A., Suraj, Z.: Discovery of concurrent data models from experimental tables: A rough set approach. In: Proceedings of the First International Conference on Knowledge Discovery and Data Mining (KDD 1995), Montreal, Canada, August 20-21, 1995, pp. 288–293. AAAI Press, Menlo Park, CA (1995)
94. Slezak, D.: On generalized decision functions: Reducts, networks and ensembles. In: Y. Yao, Q. Hu, H. Yu, J.W. Grzymala-Busse (eds.) Rough Sets, Fuzzy Sets, Data Mining, and Granular Computing - 15th International Conference, RSFDGrC 2015, Tianjin, China, November 20-23, 2015, Proceedings, *Lecture Notes in Computer Science*, vol. 9437, pp. 13–23. Springer (2015). doi:10.1007/978-3-319-25783-9_2
95. Slezak, D., Janusz, A.: Ensembles of bireducts: Towards robust classification and simple representation. In: T. Kim, H. Adeli, D. Slezak, F.E. Sandnes, X. Song, K. Chung, K.P. Arnett (eds.) Future Generation Information Technology - Third International Conference, FGIT 2011 in Conjunction with GDC 2011, Jeju Island, Korea, December 8-10, 2011. Proceedings, *Lecture Notes in Computer Science*, vol. 7105, pp. 64–77. Springer (2011). doi:10.1007/978-3-642-27142-7_9
96. Slezak, D., Stawicki, S.: The problem of finding the simplest classifier ensemble is np-hard - A rough-set-inspired formulation based on decision bireducts. In: R. Bello, D. Miao, R. Falcon, M. Nakata, A. Rosete, D. Ciucci (eds.) Rough Sets - International Joint Conference, IJCRS 2020, Havana, Cuba, June 29 - July 3, 2020, Proceedings, *Lecture Notes in Computer Science*, vol. 12179, pp. 204–212. Springer (2020). doi:10.1007/978-3-030-52705-1_15

97. Slezak, D., Widz, S.: Evolutionary inspired optimization of feature subset ensembles. In: H. Takagi, A. Abraham, M. Köppen, K. Yoshida, A.C.P.L.F. de Carvalho (eds.) Second World Congress on Nature & Biologically Inspired Computing, NaBIC 2010, 15-17 December 2010, Kitakyushu, Japan, pp. 437–442. IEEE (2010). doi:10.1109/NABIC.2010.5716365
98. Ślęzak, D., Grzegorowski, M., Janusz, A., Stawicki, S.: Toward interactive attribute selection with infolattices - A position paper. In: L. Polkowski, Y. Yao, P. Artiemjew, D. Ciucci, D. Liu, D. Ślęzak, B. Zielosko (eds.) Rough Sets - International Joint Conference, IJCRS 2017, Olsztyn, Poland, July 3-7, 2017, Proceedings, Part II, *Lecture Notes in Computer Science*, vol. 10314, pp. 526–539. Springer (2017). doi:10.1007/978-3-319-60840-2_38
99. Suraj, Z.: Discovery of concurrent data models from experimental tables: A rough set approach. Fundamenta Informaticae **28**(3-4), 353–376 (1996). doi:10.3233/FI-1996-283411
100. Suraj, Z.: Rough set methods for the synthesis and analysis of concurrent processes. In: L. Polkowski, T.Y. Lin, S. Tsumoto (eds.) Rough Set Methods and Applications: New Developments in Knowledge Discovery in Information Systems, *Studies in Fuzziness and Soft Computing*, vol. 56, pp. 379–488. Springer-Verlag/Physica-Verlag, Heidelberg (2000). doi:10.1007/978-3-7908-1840-6
101. Suraj, Z.: Discovering concurrent process models in data: A rough set approach. In: H. Sakai, M.K. Chakraborty, A.E. Hassanien, D. Ślezak, W. Zhu (eds.) Proceedings of the 12th International Conference on Rough Sets, Fuzzy Sets, Data Mining and Granular Computing (RSFDGrC'2009), Delhi, India, December 15-18, 2009, *Lecture Notes in Artificial Intelligence*, vol. 5908, pp. 12–19. Springer-Verlag, Heidelberg (2009). doi:I:10.1007/978-3-642-10646-0_2
102. Suraj, Z., Pancerz, K.: A synthesis of concurrent systems: A rough set approach. In: G. Wang, Q. Liu, Y. Yao, A. Skowron (eds.) Proceedings of the 9-th International Conference on Rough Sets, Fuzzy Sets, Data Mining, and Granular Computing (RSFDGrC'2003), Chongqing, China, May 26-29, 2003, *Lecture Notes in Artificial Intelligence*, vol. 2639, pp. 299–302. Springer-Verlag, Heidelberg (2003). doi:10.1007/3-540-39205-X_43
103. Suraj, Z., Pancerz, K.: The ROSECON system - a computer tool for modelling and analysing of processes. In: 2005 International Conference on Computational Intelligence for Modelling Control and Automation (CIMCA 2005), International Conference on Intelligent Agents, Web Technologies and Internet Commerce (IAWTIC 2005), 28-30 November 2005, Vienna, Austria, pp. 829–834. IEEE Computer Society (2005). doi:10.1109/CIMCA.2005.1631571
104. Suraj, Z., Pancerz, K.: Some remarks on computing consistent extensions of dynamic information systems. In: Proceedings of the Fifth International Conference on Intelligent Systems Design and Applications (ISDA 2005), September 8-10, 2005, Wrocław, Poland, pp. 420–425. IEEE Computer Society (2005). doi:10.1109/ISDA.2005.84
105. Suraj, Z., Pancerz, K., Owsiany, G.: On consistent and partially consistent extensions of information systems. In: D. Ślęzak, G. Wang, M. Szczuka, I. Düntsch, Y. Yao (eds.) Proceedings of the 10th International Conference on Rough Sets, Fuzzy Sets, Data Mining, and Granular Computing (RSFDGrC'2005), Regina, Canada, August 31-September 3, 2005, Part I, Lecture Notes in Artificial Intelligence, pp. 224–233. Springer-Verlag, Heidelberg (2005). doi:10.1007/11548669_24
106. Swiniarski, R.W., Pancerz, K., Suraj, Z.: Prediction of model changes of concurrent systems described by temporal information systems. In: Proceedings of The 2005 International Conference on Data Mining (DMIN 2005), Las Vegas, Nevada, USA, June 20-23, 2005, pp. 51–57. CSREA Press (2005)
107. Unnikrishnan, K.P., Ramakrishnan, N., Sastry, P.S., Uthurusamy, R.: Network reconstruction from dynamic data. SIGKDD Explorations **8**(2), 90–91 (2006). doi:10.1145/1233321.1233335
108. Valiant, L.: Research directions. `https://people.seas.harvard.edu/~valiant/researchinterests.htm` (2024). Accessed: 2024.05.28
109. van der Aalst, W.: Process Mining Discovery. Data Science in Action. Springer, Heidelberg (2016). doi:10.1007/978-3-662-49851-4. (second edition)
110. van der Aalst, W.M.P.: Process Mining Discovery, Conformance and Enhancement of Business Processes. Springer, Heidelberg (2011). doi:0.1007/978-3-642-19345-3

111. Vapnik, V.: Statistical Learning Theory. John Wiley & Sons, New York, NY (1998)
112. Wu, F.X.: Inference of gene regulatory networks and its validation. Current Bioinformatics **2**(2), 139–144 (2007). doi:10.2174/157489307780618240
113. Wu, H.C.: Mathematical Foundation of Fuzzy Sets. John Wiley & Sons, Hoboken, NJ (2023). doi:10.1002/9781119981558
114. Xu, W., Li, W., Zhang, X.: Generalized multigranulation rough sets and optimal granularity selection. Granular Computing **2**, 271–288 (2017). doi:10.1007/s41066-017-0042-9
115. Yadav, V., Dubey, A.K., Singh, H.P., Dubey, G., Suryani, E. (eds.): Process Mining Techniques for Pattern Recognition. Concepts, Theory, and Practice. CRC Press, Boca Raton, FL (2022)
116. Yang, Q., Liu, Y., Cheng, Y., Kang, Y., Chen, T., Yu, H.: Federated Learning, *Synthesis Lectures on Artificial Intelligence and Machine Learning*, vol. 43. Morgan & Claypool, Williston, VT (2020). doi:10.2200/S00960ED2V01Y201910AIM043
117. Yao, Y.: Combination of rough and fuzzy sets based on α-level sets. In: Rough Sets and Data Mining: Analysis of Imprecise Data, pp. 301–321. Springer US, Boston, MA (1997). doi:10.1007/978-1-4613-1461-5_15
118. Zadeh, L.A.: Foreword. In: S.K. Pal, L. Polkowski, A. Skowron (eds.) Rough-Neural Computing: Techniques for Computing with Words, pp. ix–xi. Springer, Heidelberg (2004). doi:10.1007/978-3-642-18859-6
119. Zadeh, L.A.: Computing with Words: Principal Concepts and Ideas, *Studies in Fuzziness and Soft Computing*, vol. 277. Springer, Heidelberg (2012). doi:10.1007/978-3-642-27473-2
120. Zhan, J., Xu, W.: Two types of coverings based multigranulation rough fuzzy sets and applications to decision making. Artificial Intelligence Review **53**(1), 167–198 (2020). doi:10.1007/s10462-018-9649-8
121. Zhan, J., Zhang, X., Yao, Y.: Covering based multigranulation fuzzy rough sets and corresponding applications. Artificial Intelligence Review **53**(2), 1093–1126 (2020). doi:10.1007/S10462-019-09690-Y. URL `https://doi.org/10.1007/s10462-019-09690-y`

Chapter 6
Algebras based on Rough Sets

6.1 Introduction

The algebraic study of rough set theory began soon after the proposal of the notion of rough sets by Pawlak [**?**]. The research focussed on the kind of algebraic structures that rough sets formed, and a study of the properties. The result is a list of algebras, both new and well-known, the common feature being that each has a 'rough set' as an instance. In this chapter, we present an updated account of the developments that have taken place in the area over the last two decades. Our main references are [**?**, **?**].

In order to develop an algebra an instance (or model) of which would be rough sets, it is clearly necessary to specify the definition of a rough set. As we have observed in Section **??** of Chapter 2, there are four set-based definitions of a (Pawlak) rough set itself, viz. Definitions 2.8, 2.9, 2.10, 2.11. Apart from these, we also have the 'operator-oriented' view [**?**] reflected by Definitions 2.6, 2.7.

Let us fix an approximation space (U,R). We denote by $\mathscr{F}_i,\ i=1,\ldots,5$, the collection of all rough sets corresponding to (U,R), given by Definition 2.6, 2.8,2.9, 2.10, 2.11 respectively. The version of Definition 2.10 where (U,R) is fixed in all the quadruples, is considered to get $\mathscr{F}_4$. We had observed that Definitions 2.8, 2.9, 2.10 and 2.11 are equivalent to each other for any given (U,R), in the sense that there is a one-one correspondence between families $\mathscr{F}_2,\ \mathscr{F}_3,\ \mathscr{F}_4$ and $\mathscr{F}_5$. As we shall see in this chapter, these equivalent definitions yield different (though related) algebras, by taking different definitions of union, intersection, complementation and other algebraic operations.

Apart from the above families $\mathscr{F}_1,\ldots,\mathscr{F}_5$, another collection that makes an appearance in the algebraic studies is $\mathscr{R} \equiv \{(D_1,D_2),\ D_1 \subseteq D_2,\ D_1,D_2 \in \mathscr{D}\}$, where $\mathscr{D}$ denotes the collection of all definable sets of a fixed approximation space (U,R). Notice that $\mathscr{R}$ is a generalization of $\mathscr{F}_2$: $\mathscr{F}_2 \subseteq \mathscr{R}$. Further, since $D_2 \setminus D_1$ may include singleton equivalence classes, $\mathscr{F}_2$ may be a proper subset of $\mathscr{R}$.

A summary of the structures obtained by considering the families above are presented in Section 6.2. Section 6.3 presents the algebras obtained from the operator-based definitions. Both Sections 6.2 and 6.3 give the main results proved about the algebras as well. In Section 6.4, we outline the various relationships that surface among these algebras.

6.2 Algebras from Set-based Definitions

One finds that the different algebras emerging from the set-based definitions of rough sets are instances of :

1. quasi-Boolean algebras [**?**, **?**];
2. (a) topological quasi-Boolean algebras [**?**, **?**];
   (b) pre-rough algebras [**?**];
   (c) rough algebras [**?**];
3. regular double Stone algebras [**?**, **?**];
4. complete atomic Stone algebras [**?**];
5. semi-simple Nelson algebras [**?**];
6. 3-valued Łukasiewicz algebras [**?**, **?**].

In the sequel, we sketch how exactly these algebras come about, starting from the definitions of rough sets. Broadly, the scheme is of the following nature. As remarked in the Introduction, the primary task is to fix the definition of a rough set and therefore the corresponding family. The next step is to take an appropriate operation of union / intersection / complementation / interior etc. on the family, to give rise to a class of algebraic structures, say $\mathscr{RS}$. This leads, on abstraction (according to the properties of the operations in $\mathscr{RS}$), to one of the classes of algebras in the preceding list. In many cases, the connection between $\mathscr{RS}$ and the corresponding class (say $\mathscr{A}$) of abstract algebras, is formalized by establishing a *representation* result. One demonstrates a correspondence $c : \mathscr{A} \to \mathscr{RS}$ such that any element $A \in \mathscr{A}$ is *isomorphic* to a subalgebra of $c(A)$. In some cases, a reverse representation is proved.

We note that when the family $\mathscr{F}_2$, or the more general $\mathscr{R}$ is considered, a natural definition for the operations of union and intersection of the members would be the following.

Definition 6.1.

- $(D_1, D_2) \sqcup (D'_1, D'_2) \;\equiv\; (D_1 \cup D'_1, D_2 \cup D'_2)$,
- $(D_1, D_2) \sqcap (D'_1, D'_2) \;\equiv\; (D_1 \cap D'_1, D_2 \cap D'_2)$.

A restriction of these operations to the subclass $\mathscr{F}_2$ would give:

Definition 6.2.

- $(\underline{A}, \overline{A}) \sqcup (\underline{B}, \overline{B}) \;\equiv\; (\underline{A} \cup \underline{B}, \overline{A} \cup \overline{B})$,
- $(\underline{A}, \overline{A}) \sqcap (\underline{B}, \overline{B}) \;\equiv\; (\underline{A} \cap \underline{B}, \overline{A} \cap \overline{B})$.

One needs to ensure closure here, i.e. to check whether the right-hand entities in the above do belong to $\mathscr{F}_2$. However, we recall the sets $(C_i, D_i), i = 1,2,3$ from Section 2.4 of Chapter 2 – these precisely serve the purpose.

6.2.1 Quasi-Boolean algebras

Definition 6.3. [**?**] A quasi-Boolean algebra (or a De Morgan lattice) is a bounded distributive lattice $(A, \leq, \vee, \wedge, 0, 1)$ with a unary operation $\neg$ that satisfies involution ($\neg\neg a = a$, for each $a \in A$), and makes the De Morgan identities hold.

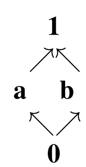
Iwiński [**?**] and Pomykała [**?**] show that rough sets form structures that are quasi-Boolean algebras. Iwiński, presenting a 'rough algebra' for the first time, follows Definition 2.8. The general collection $\mathscr{R}$ instead of $\mathscr{F}_2$ is considered. Operations of join ($\sqcup$) and meet ($\sqcap$) on $\mathscr{R}$ are given by Definition 6.1. It may be noted that any definable set A of (U,R) is identifiable with the pair (A,A) of $\mathscr{R}$. Further,

Definition 6.4. $\neg(D_1, D_2) \equiv ({D_2}^c, {D_1}^c)$, where ^{c} denotes ordinary set-theoretic complementation.

$\neg$ satisfies the De Morgan identities, and when restricted to definable sets, is the usual complement. But it does not satisfy the laws of Boolean complementation in general.

Proposition 6.1. *[?]* $(\mathscr{R}, \sqcup, \sqcap, \neg, 0, 1)$ *is a complete atomic quasi-Boolean algebra, where* $0 \equiv (\emptyset, \emptyset)$ *and* $1 \equiv (U,U)$*. Atoms are of the form* $(\emptyset, A)$*, A being an elementary set of* (U,R)*. The definable sets form a maximal Boolean subalgebra of* $(\mathscr{R}, \sqcup, \sqcap, \neg, 0, 1)$.

Does the converse of this proposition hold? Let us see.
A basic finite quasi-Boolean algebra is $\mathscr{U}_0 \equiv (\{0, a, b, 1\}, \vee, \wedge, \neg, 1)$. It is a diamond as a lattice, viz.



and $\neg$ is given by the equations :

$\neg 0 = 1,\ \neg 1 = 0,\ \neg a = a,\ \neg = b.$

It is known [**?**] that any quasi-Boolean algebra is isomorphic to a subalgebra of the product $\prod_{i \in I} \mathscr{U}_i$, where I is a set of indices, and $\mathscr{U}_i = \mathscr{U}_0$.

Hence, to address the converse of proposition 6.1, it seems natural to ask if $\mathscr{U}_0$ is isomorphic to some $\mathscr{R}$. The answer is in the negative, since for any member (D_1, D_2) of $\mathscr{R}$ other than $(\emptyset, X)$, $\neg(D_1, D_2) \neq (D_1, D_2)$, whereas in $\mathscr{U}_0$, $\neg a = a,\ \neg b = b$ and $a \neq b$. So the class $\mathscr{R}$ is a proper subclass of the class of quasi-Boolean algebras.

No representation result is proved in [**?**]. However, if one considers the family $\mathscr{F}_2$, such a result is obtained in [**?**].

Clearly, $(\mathscr{F}_2, \sqcup, \sqcap, \neg, 0, 1)$ is also a quasi-Boolean algebra, the operations being restrictions of those in $\mathscr{R}$. But [**?**] says more. An important notion involved here is that of an 'individual atom' – a singleton elementary class. Let us denote by S, the collection of all individual atoms in the approximation space $< X, R >$.
Two simple examples of quasi-Boolean algebras are the two and three element chains $\mathscr{B}_0 \equiv (\{0,1\}, \vee, \wedge, \neg, 1)$ and $\mathscr{C}_0 \equiv (\{0,a,1\}, \vee, \wedge, \neg, 1)$ respectively.
$\neg$ is defined as: $\neg 0 \equiv 1,\ \neg 1 \equiv 0,\ \neg a \equiv a$.

Theorem 6.1. $(\mathscr{F}_2, \sqcup, \sqcap, \neg, 0, 1)$ *is isomorphic to a subalgebra of the product* $\prod_{i \in I} \mathscr{U}_i$*, where I is a set of indices, and* $\mathscr{U}_i = \mathscr{B}_0$ *or* $\mathscr{U}_i = \mathscr{C}_0$*, for each* $i \in I$.

For the proof, I is considered to be the quotient set U/R, i.e. the family of all elementary sets in (U,R). Further, if $i (\in I)$ is an individual atom, i.e. $i \in S$, then $\mathscr{U}_i = \mathscr{B}_0$, and $\mathscr{U}_i = \mathscr{C}_0$ otherwise. The isomorphism f between $\mathscr{RS}$ and $\prod_{i \in I} \mathscr{U}_i$ is defined as follows. Let $(\underline{A}, \overline{A}) \in \mathscr{RS}$.
$f((\underline{A}, \overline{A})) \equiv (x_i)_{i \in I}$, $(x_i)_{i \in I} \in \prod_{i \in I} \mathscr{U}_i$, if and only if

1. $i \in S$ and $i \subseteq \underline{A}$ imply $x_i = 1$,
2. $i \in S$ and $i \not\subseteq \underline{A}$ imply $x_i = 0$,
3. $i \notin S$, $i \not\subseteq \underline{A}$ and $i \not\subseteq \overline{A}$ imply $x_i = 0$,
4. $i \notin S$, $i \not\subseteq \underline{A}$ and $i \subseteq \overline{A}$ imply $x_i = 1$,
5. $i \notin S$ and $i \subseteq \underline{A}$ imply $x_i = 1$.

It should be mentioned that Pomykała came up with a number of algebraic structures that have $\mathscr{F}_2$ as domain. These differ from each other with respect to the complementation and implication operations chosen.***

6.2.2 Topological quasi-Boolean algebras

Definition 6.5. [**?**, **?**] A topological quasi-Boolean algebra (tqBa) is a quasi-Boolean algebra $(A, \leq, \vee, \wedge, \neg, 0, 1)$ with an interior (unary) operation L that satisfies

L1 $La \leq a$,
L2 $L(a \wedge b) = La \wedge Lb$,
L3 $LLa = La$,
L4 $L1 = 1$ and
L5 $MLa = La$,

where M is the closure operation, viz. $Ma \equiv \neg L \neg a, a, b \in A$.

Proceeding from Section 6.2.1, one may define an interior operation L on $(\mathscr{R}, \sqcup, \sqcap, \neg, 0, 1)$:

Definition 6.6. $L(D_1, D_2) \equiv (D_1, D_1)$, $D_1, D_2 \in \mathscr{D}$.
Thus, the closure M on $\mathscr{R}$ is given by $M(D_1, D_2) = (D_2, D_2)$.

On the other hand, one may start from Definition 2.11, and define, for $[A], [B] \in \mathscr{R}U)/\approx$,

- $[A] \sqcap [B] \equiv [A \sqcap B]$,
- $\neg[A] \equiv [A^c]$,
- $L[A] \equiv [\underline{A}]$,

where $A \sqcap B$ $(= D_1)$ is as given in Section 2.4.1 of Chapter 2.
One then obtains

Proposition 6.2. *L as in Definition 6.6 gives the tqBa* $(\mathscr{R}, \sqcup, \sqcap, \neg, L, 0, 1)$. *Restricting L to* $\mathscr{F}_2$ *makes* $(\mathscr{F}_2, \sqcup, \sqcap, \neg, 0, 1)$ *form a tqBa. The quotient set* $\mathscr{R}U)/\approx$ *yields a tqBa structure as well, with the preceding definitions of* $\sqcap, \neg$ *and L.*

The tqBa on $\mathscr{R}U)/\approx$ is isomorphic to that on $\mathscr{F}_2$. It is also isomorphic to that on $\mathscr{R}$, provided of course, no definable set in (U, R) is a singleton.

No representation result of rough structures with respect to tqBa's have been proved. As a matter of fact, the class of tqBa's itself is open to investigation. Algebraically, the following is the only known result so far.

Proposition 6.3. *[?] TqBa's form a variety that is not a discriminator variety.*

The tqBas on $\mathscr{R}U)/\approx$ and $\mathscr{F}_2$ satisfy more properties, as we shall see in Sections 6.2.3 and 6.2.4.

6.2.3 Pre-rough algebras

The following are added to the definition of a tqBa to get a pre-rough algebra.

Definition 6.7. [?] A *pre-rough algebra* is a tqBa $(A, \leq, \vee, \wedge, \neg, L, 0, 1)$ in which

- $\neg La \vee La = 1$,
- $L(a \vee b) = La \vee Lb$,
- $La \leq Lb, Ma \leq Mb$ imply $a \leq b$.

One may define an 'implication' operation in this structure as

$$a \Rightarrow b \equiv (\neg La \sqcup Lb) \sqcap (\neg Ma \sqcup Mb).$$

Observation 6 *In a pre-rough algebra* $\mathscr{P} \equiv (A, \leq, \vee, \wedge, \neg, L, 0, 1)$, $\mathscr{L}(A) \equiv (L(A), \leq, \vee, \wedge, \neg, 0, 1)$ *is a Boolean algebra (using the first axiom in Definition 6.7). It may also be noticed that* $L(A) = M(A)$.

Example 6.1. Let $\mathscr{T} \equiv (A, \leq, \vee, \wedge, \neg, L, 0, 1)$, where $A \equiv \{0, a, 1\}$,

$$\begin{array}{c} 1 \\ \uparrow \\ a \\ \uparrow \\ 0 \end{array}$$

with $\neg 0 \equiv 1$, $\neg a \equiv a$, $\neg 1 \equiv 0$, $L0 \equiv 0 \equiv La$, $L1 \equiv 1$. T is the smallest non-trivial pre-rough algebra.

The tqBa's on $\mathscr{R}U)/\approx$ and $\mathscr{T}_2$ (and also on $\mathscr{R}$) are pre-rough algebras.

A representation result [**?**] shows that any pre-rough algebra is, in fact, an algebra of pairs of Boolean elements.

Theorem 6.2. *Any pre-rough algebra* $(A, \leq, \sqcap, \sqcup, \neg, L, 0, 1)$ *is isomorphic to the pre-rough algebra formed by the set* $B \equiv \{(La, Ma) : a \in A\}$. *The operations on* B *are defined just by abstracting those on* $\mathscr{T}_2$.

6.2.4 Rough algebras

Definition 6.8. [**?**] A *rough algebra* $\mathscr{P} \equiv (A, \leq, \sqcap, \sqcup, \neg, L, \Rightarrow, 0, 1)$ is a pre-rough algebra such that the subalgebra $(L(A), \leq, \sqcap, \sqcup, \neg, 0, 1)$ of $\mathscr{P}$, where $L(A) \equiv \{La : a \in A\}$, is

- complete and
- completely distributive, i.e. $\sqcup_{i \in I} \sqcap_{j \in J} a_{i,j} = \sqcap_{f \in J^I} \sqcup_{i \in I} a_{i,f(i)}$, for any index sets I, J and elements $a_{i,j}, i \in I, j \in J$, of $L(A)$, J^I being the set of maps of I into J.

The pre-rough algebras on each of $\mathscr{R}U)/\approx$, $\mathscr{T}_2$ and $\mathscr{R}$, are rough algebras as well. The following representation result is then obtained.

Theorem 6.3. *Any rough algebra is isomorphic to a subalgebra of* $(\mathscr{R}, \sqcup, \sqcap, \neg, L, 0, 1)$ *corresponding to some approximation space* (U, R).

Proof. Let $\mathscr{P} \equiv (A, \leq, \sqcap, \sqcup, \neg, L, 0, 1)$ be a rough algebra.
Then $\mathscr{L}(A) \equiv (L(A), \leq, \sqcap, \sqcup, \neg, 0, 1)$ is a complete Boolean subalgebra of $\mathscr{P}$ that is completely distributive. Hence it is isomorphic to a complete field of sets [**?**],
$\mathscr{C} \equiv (C, \subseteq, \cap, \cup, ^c, \emptyset, 1)$, say.
$\mathscr{C}$ is atomic [**?**], and let X denote the union of all its atoms. The atoms induce a partition R (say), of X. Thus we have an approximation space (X, R).
It may then be noticed that $C = DS$, the collection of all definable sets of (X, R).***
So the isomorphism of $\mathscr{L}(A)$ and $\mathscr{C}$ implies the isomorphism of $TQ(\mathscr{L}(A))$ and the approximation space algebra $TQ(\mathscr{D})$ of $< X, R >$. Now there is an isomorphic copy of $\mathscr{P}$ in $TQ(\mathscr{L}(A))$ (cf. Theorem 2.1), and hence in $TQ(\mathscr{D})$, and this is the required subalgebra of $TQ(\mathscr{D})$. □

In fact, one can show [**?**] that

Corollary 6.1. *Any rough algebra is isomorphic to a subalgebra of* $\mathscr{R}U')/\approx$ *for some approximation space* (U',R').

6.2.5 Complete atomic Stone algebras

Definition 6.9. (**Ref: Birkhoff?) A *Stone algebra* is a bounded distributive lattice $(A,\leq,\vee,\wedge,0,1)$ which has a *pseudo-complement* * on A, i.e. $y \leq x^*$ if and only if $y \wedge x = 0$, and which satisfies the *Stone identity*, viz. $x^* \vee x^{**} = 1$.

In [**?**], Pomykała defines * on $\mathscr{F}_2$ as:

Definition 6.10. $(\underline{A},\overline{A})^* \equiv (\overline{A}^c,\overline{A}^c)$, $(\underline{A},\overline{A}) \in \mathscr{F}_2$.

Then one obtains (with $\sqcup,\sqcap$ as in Definition 6.2, and $0,1$ as in Proposition 6.1)

Proposition 6.4. $(\mathscr{F}_2,\sqcup,\sqcap,^*,0,1)$ *is a Stone algebra.*

However, no representation is obtained.

Starting from Definition 2.11, [**?**] arrives at an enhanced rough structure. $(\mathscr{R}U)/\approx,\leq)$ is clearly a partially ordered set, $\leq$ being defined in terms of rough inclusion, i.e. $[A] \leq [B]$, if and only if A is roughly included in B, $[A],[B] \in \mathscr{R}U)/\approx$. Now operations of join ($\cup_{\approx}$), meet ($\cap_{\approx}$) on $\mathscr{R}U)/\approx$ and ('exterior') complementation ($^{\text{ex}}$) are defined.

For a subset A of U, an *upper sample* P is such that $P \subseteq A$ and $\overline{P} = \overline{A}$. An upper sample P of A is *minimal*, if there is no upper sample Z of A with $Z \subseteq P$. Then

Definition 6.11.
- $[A] \cup_{\approx} [B] \equiv [\underline{A} \cup \underline{B} \cup P]$, where P is a minimal upper sample of $\overline{A} \cup \overline{B}$, and
- $[A] \cap_{\approx} [B] \equiv [(\underline{A} \cap \underline{B}) \cup P]$, where P is a minimal upper sample of $\overline{A} \cap \overline{B}$.
- $[A]^{\text{ex}} \equiv [(\overline{A})^c]$.

One may note that $\emptyset$ is included among elementary sets. For a finite domain U,

Proposition 6.5. $(\mathscr{R}U)/\approx,\cup_{\approx},\cap_{\approx},^{\text{ex}},[\emptyset],[U])$ *is a complete atomic Stone algebra, where the atoms are determined by proper subsets of the elementary sets or by singleton elementary sets in* (U,R).

Again, no representation is obtained. Such a result is found though, on introducing a further operation on the family of rough sets $\mathscr{F}_2$.

6.2.6 Regular double Stone algebras

Definition 6.12. (**Ref) A *double Stone algebra* (dSa) is a Stone algebra $(A,\vee,\wedge,^*,0,1)$ which has a *dual pseudo-complement* $^+$, i.e. $y \geq x^+$ if and only if $y \vee x = 1$, and

which satisfies $x^+ \wedge x^{++} = 0$.
The dSa is *regular* if, for all $x, y \in A$, $x \wedge x^+ \leq y \vee y^*$
holds. (This is equivalent to requiring that $x^* = y^*,\ x^+ = y^+$ imply $x = y$, for all $x, y \in A$.)

[**?**] introduces a dual pseudo-complementation $^+$ on $\mathscr{F}_2$ and gets, further to Proposition 6.4,

Proposition 6.6. $(\mathscr{F}_2, \sqcup, \sqcap, ^*, ^+, 0, 1)$*, for a given approximation space* (U, R)*, is a regular dSa, where* $(\underline{A}, \overline{A})^+ \ \equiv\ (\underline{A}^c, \underline{A}^c)$.

As a representation result, Comer obtains

Theorem 6.4. *Any regular dSa is isomorphic to a subalgebra of* $(\mathscr{F}_2, \sqcup, \sqcap, ^*, ^+, 0, 1)$ *for some approximation space* (U, R).

6.2.7 Semi-simple Nelson algebras

Definition 6.13. [**?**] A *Nelson algebra* is a quasi-Boolean algebra $(A, \wedge, \vee, \neg, 0, 1)$ equipped with a unary operation $\sim$ and a binary operation $\to$ such that, for any $a, b, x \in A$,

- $a \wedge \neg a \leq b \vee \neg b$,
- $a \wedge x \leq \neg a \vee b$ if and only if $x \leq a \to b$,
- $a \to (b \to c) = (a \wedge b) \to c$,
- $\sim a = a \to \neg a = a \to 0$.

A Nelson algebra A is *semi-simple*, if $a \vee \sim a = 1$, for all $a \in A$.

$\neg$ and $\sim$ are the ‘strong’ and ‘weak’ negations on A respectively.

These algebras are discussed in the context of rough sets in [**?**], which considers finite domains, and adopts Definition 2.9. It is observed that

Proposition 6.7. $(\mathscr{F}_3, \sqcap, \sqcup, \neg, \sim, \to, 0, 1)$ *is a semi-simple Nelson algebra, the operations being defined as:*

- $(\underline{A_1}, \overline{A_1}^c) \sqcap (\underline{A_2}, \overline{A_2}^c) \ \equiv\ (\underline{A_1} \cap \underline{A_2}, \overline{A_1}^c \cup \overline{A_2}^c)$,
- $(\underline{A_1}, \overline{A_1}^c) \sqcup (\underline{A_2}, \overline{A_2}^c) \ \equiv\ (\underline{A_1} \cup \underline{A_2}, \overline{A_1}^c \cap \overline{A_2}^c)$,
- $(\underline{A_1}, \overline{A_1}^c) \to (\underline{A_2}, \overline{A_2}^c) \ \equiv\ (\underline{A_1}^c \cup \underline{A_2}, \underline{A_1} \cap \overline{A_2}^c)$,
- $\neg(\underline{A_1}, \overline{A_1}^c) \ \equiv\ (\overline{A_1}^c, \underline{A_1})$, *and*
- $\sim (\underline{A_1}, \overline{A_1}^c) \ \equiv\ (\underline{A_1}^c, \underline{A_1})$.

The representation theorem is as follows.

Theorem 6.5. *Any finite semi-simple Nelson algebra is isomorphic to* $(\mathscr{F}_3, \sqcap, \sqcup, \neg, \sim, \to, 0, 1)$ *for some approximation space* (U, R).

$\mathscr{F}_3$ also forms a Stone as well as a regular double Stone algebra with suitable operations. The operations are derived from those which make $\mathscr{F}_3$ a Nelson algebra (cf. proposition 6.7). $\sqcap, \sqcup$ remain the same, while the pseudo-complementation * is taken as $\neg \sim \neg$, and the dual pseudo-complementation $^+$ as $\sim$.

6.2.8 3-valued Łukasiewicz algebras

Definition 6.14. (cf. [**?**]) A 3-valued Łukasiewicz (Moisil) algebra $(A, \leq, \wedge, \vee, \neg, M, 0, 1)$ is such that $(A, \leq, \wedge, \vee, \neg, 0, 1)$ is a quasi-Boolean algebra and M is a unary operator on A satisfying, for all $a, b \in A$

- $M(a \wedge b) = Ma \wedge Mb$,
- $M(a \vee b) = Ma \vee Mb$,
- $Ma \wedge \neg Ma = 0$,
- $MMa = Ma$,
- $M\neg Ma = \neg Ma$,
- $\neg M \neg a \leq Ma$, and
- $Ma = Mb, M\neg a = M\neg b$ imply $a = b$.

A direct representation result concerning this class of algebras has been obtained in [**?**]. However, the same has been concluded through relationships of 3-valued Łukasiewicz algebras with other algebras, in both [**?**] and [**?**]. We shall elaborate on this in Section 6.4.

[**?**] considers Definition 2.8, i.e. the family $\mathscr{F}_2$, and also finite domains.

With the operator M as in Definition 6.6 (restricted to $\mathscr{F}_2$), $\sqcup, \sqcap$ as in Definition 6.2, $\neg$ as in Definition 6.4, $0 \equiv (\emptyset, \emptyset)$, $1 \equiv (U, U)$ (cf. Proposition 6.1), one finds that

Proposition 6.8. $(\mathscr{F}_2, \sqcup, \sqcap, \neg, M, 0, 1)$ *is a 3-valued Łukasiewicz algebra.*

The representation theorem is as follows.

Theorem 6.6. *Every 3-valued Łukasiewicz algebra is isomorphic to a subalgebra of* $(\mathscr{F}_2, \sqcup, \sqcap, \neg, M, 0, 1)$ *corresponding to some approximation space* (U, R).

6.2.9 Other algebras

In the special situation when the approximation space has no singleton elementary sets in it, [**?**] observes that $\mathscr{F}_2$ with its pairs in reverse order, viz. the collection of pairs $(\overline{A}, \underline{A}), A \subseteq U$, turns out to be a *Post algebra* of order three [**?**]. Therefore [**?**], it is a 3-valued Łukasiewicz algebra with a centre (i.e. an element c such that $\neg c = c$).

In the general situation (with no restriction on the approximation space), the same structure can be made into an algebra that is a generalization of a Post algebra, viz. a certain chain-based lattice of order three [**?**].

6.3 Algebras from Operator-based Definitions

6.3.1 Boolean algebras with operators

**Yiyu

Definition 6.15. [**?**] A Boolean algebra with operators is a Boolean algebra $(A, \vee, \wedge, \sim, 0, 1,)$ along with a collection $\{f_i\}_{i \in I}$ of operators on A, where I is an index set. Each n-ary operator $f_i, i \in I$, satisfies the two following properties:
O1. $f_i(a_1, ..., 0, ..., a_n) = 0$ (normality) and
O2. $f_i(a_1, ..., a_i \vee a_{i'}, ..., a_n) = f_i(a_1, ..., a_i, ..., a_n) \vee f_i(a_1, ..., a_{i'}, ..., a_n)$ (additivity).

For any set U, the power set $\mathscr{P}U)$ of U forms a Boolean algebra with the standard set-theoretic operations. If (U, R) is an approximation space, $\mathscr{P}U)$ also forms a *topological Boolean algebra*. We recall that

Definition 6.16. [**?**] A topological Boolean algebra (tBa) is a Boolean algebra $(A, \vee, \wedge, \sim, 0, 1)$, that has a unary operation L satisfying the properties L1-L4 of an interior given in Definition 6.5.

The interior operation on $\mathscr{P}U)$ is nothing but the lower approximation with respect to the approximation space (U, R), regarded as an operator on U, i.e. $L(X) \equiv \underline{X}$, for any $X \subseteq U$. L, in fact, satisfies all the properties L1-L5 of Definition 6.5. The upper approximation operator, denoted by $^-$ say, would then satisfy the properties of a closure operator, which include $O1, O2$ of Definition 6.15. In other words, as observed in [**?**], $(\mathscr{P}U), \cup, \cap, ^c, ^-, \emptyset, U)$ forms a *monadic Boolean algebra* [**?**], that is an instance of a Boolean algebra with the single binary operator $^-$.

The tBa formed by the power set is called the *topological field of sets* [**?**].

6.3.2 Cylindric algebras

Definition 6.17. Cylindric algebra

[**?**] considers the following version of an *information system* [**?**]: it is a quadruple $S \equiv (U, AT, V, f)$, where U is a set, AT a finite set, V a function with domain AT, and $f : U \to \prod_{a \in AT} V_a$. Now each $P(\subseteq AT)$ induces an equivalence relation R_P on U, giving (U, R_P), an *approximation space for knowledge P*. Each R_P in turn, induces an 'upper approximation operator' $\overline{P} : \mathscr{P}U) \to \mathscr{P}U)$, i.e. $\overline{P}(A)$ is the union of equivalence classes under R_P of all elements of $A(\subseteq U)$.

Then the structure $(\mathscr{P}U), \cup, \cap, ^c, \overline{P}, \emptyset, U), P \subseteq AT$, is called a *knowledge approximation algebra of type AT derived from the information system S*. For each $P \subseteq AT$, the structure $(\mathscr{P}U), \cup, \cap, ^c,, \overline{P}, \emptyset, U)$ is called the (upper) *approximation closure algebra of P*. It may be noted that this is an instance of a monadic Boolean algebra.

Now $(\mathscr{R}U), \cup, \cap, ^c, \emptyset, U)$ is not only a Boolean algebra, as mentioned in Section 6.3.1, it is also complete, and atomic. In fact, Comer observes that a knowledge approximation algebra of type AT is an instance of a general algebraic structure which consists of a complete atomic Boolean algebra $(B, \vee, \wedge, \sim, 0, 1)$, and a family of functions $K_P : B \to B$, $P \subseteq AT$, AT being a finite set. Moreover, the functions satisfy the following, for $x, y \in B$ and $P, Q \subseteq AT$.

- $K_P(0) = 0$.
- $K_P(x) \geq x$.
- $K_P(x \wedge K_P(y)) = K_P(x) \wedge K_P(y)$.
- If $x \neq 0$ then $K_\emptyset(x) = 1$.
- $K_{P \cup Q}(x) = K_P(x) \wedge K_Q(x)$, if x is an atom of B.

This leads to

Proposition 6.9. *Approximation closure algebras are complete atomic cylindric algebras of dimension one.*

A representation theorem is subsequently obtained.

Theorem 6.7. *Every complete atomic cylindric algebra of dimension one is isomorphic to an approximation closure algebra. In fact, every cylindric algebra of dimension one is embeddable in an approximation closure algebra.*

6.4 Relationships

As observed in [**?**, **?**], one can define to and fro transformations to show that pre-rough, regular double Stone, semi-simple Nelson and 3-valued Łukasiewicz algebras are all equivalent to each other.

It is not difficult to see that the defining axioms of pre-rough and 3-valued Łukasiewicz algebras (cf. Definitions 6.7 and 6.14 respectively), are deducible from each other.

The transformations involved for a passage to and from a pre-rough algebra $(A, \wedge, \vee, \neg, L, \Rightarrow, 0, 1)$ and a regular double Stone algebra (Definition 3) $(L, \vee, \wedge, ^*, ^+, 0, 1)$ are:

DP1. $(a,b)^+ \equiv \neg L(a,b)$,
DP2. $(a,b)^* \equiv L\neg(a,b)$, and
PD. $L((a,b)) \equiv (\neg a, b)^+$.

For a semi-simple Nelson algebra (Definition 5) $\mathscr{N} \equiv (A, \wedge, \vee, \neg, \sim, \to, 0, 1)$ and a pre-rough algebra $(A, \wedge, \vee, \neg, L, \Rightarrow, 0, 1)$, the transformations are:

NP1. $La = \neg \sim a$,
NP2. $a \Rightarrow b = \neg \sim (a \hookrightarrow b)$, where $a \hookrightarrow b \equiv (\sim a \wedge \sim \neg b) \vee (\neg \sim \neg a \vee b)$, and
PN1. $\sim a = \neg La$,
PN2. $a \to b = \neg La \vee b$.

It may be noted that an equivalent axiomatization of 3-valued Łukasiewicz algebras is given by the *Wajsberg algebras*.

Definition 6.18. [**?**] A Wajsberg algebra is a structure $(A, \rightarrow, \neg, 1)$ such that

1. $a \rightarrow (b \rightarrow a) = 1$,
2. $(a \rightarrow b) \rightarrow ((b \rightarrow c) \rightarrow (a \rightarrow c)) = 1$,
3. $((a \rightarrow \neg a) \rightarrow a) \rightarrow a = 1$,
4. $(\neg a \rightarrow \neg b) \rightarrow (b \rightarrow a) = 1$,
5. If $1 \rightarrow a = 1$ then $a = 1$,
6. If $a \rightarrow b = 1 = b \rightarrow a$ then $a = b$,

where $a, b, c \in A$.

Thus Wajsberg algebras also get related to the group of algebras being considered here. The transformations involved for a 3-valued Lukasiewicz algebra $(A, \leq, \sqcap, \sqcup, \neg, M, 0, 1)$ and a Wajsberg algebra $(A, \rightarrow, \neg, 1)$ are:

LW. $a \rightarrow b \equiv (M\neg a \sqcup b) \sqcap (Mb \sqcup \neg a)$, and
WL1. $a \sqcup b \equiv (a \rightarrow b) \rightarrow b$,
WL2. $a \sqcap b \equiv \neg(\neg a \sqcup \neg b)$,
WL3. $Ma \equiv \neg a \rightarrow a$,
WL4. $0 \equiv \neg 1$.

A 3-valued Łukasiewicz algebra is cryptoisomorphic to an MV_3-algebra (**def:) [**?**] in the sense of Birkhoff [**?**]. Thus all the preceding algebras are also cryptoisomorphic to MV_3-algebras as well.

** Yiyu: Relations between the operator-based algebras

References

1. Banerjee, M.: Rough sets and 3-valued Łukasiewicz logic. Fundamenta Informaticae **32**, 213–220 (1997)
2. Banerjee, M., Chakraborty, M. K.: Rough algebra. Bull. Polish Acad. Sc.(Math.) **41**(4), 293–297 (1993)
3. Banerjee, M., Chakraborty, M. K.: Rough sets through algebraic logic. Fundamenta Informaticae **28**(3-4), 211–221 (1996)
4. Banerjee, M., Chakraborty, M. K.: Algebras from rough sets. In: Pal, S. K., Polkowski, L., Skowron, A. (eds.) Rough-neuro Computing: Techniques for Computing with Words, pp. 157–184. Springer Verlag, Berlin (2004)
5. Banerjee, M. and Pal, K.: The variety of topological quasi-Boolean algebras. In: Trends in Logic III – Int. Conf. in Memoriam A. Mostowski, H. Rasiowa, C. Rauszer, Warsaw, Poland (2005). Extended Abstract.
`http://www.studialogica.org/`.
6. Banka, H. and Mitra, S.: Feature Selection, Classification and Rule Generation Using Rough Sets. In: Peters, G., Lingras, P., Ślezak, D. and Yao, Y. (eds.) Rough Sets: Selected Methods and Applications in Management and Engineering, pp. 51–76. Springer, London (2012)

7. Bazan, J. G.: A Comparison of Dynamic and non-Dynamic Rough Set Methods for Extracting Laws from Decision Tables. In: Polkowski, L. and Skowron, A. (eds.) Rough Sets in Knowledge Discovery 1: Methodology and Applications, Series Studies in Fuzziness and Soft Computing, vol 18, pp. 321–365. Physica-Verlag, Heidelberg (1998)
*@bookpolkowski98rough-methodology, editor = Lech Polkowski and Andrzej Skowron, title = Rough Sets in Knowledge Discovery 1: Methodology and Applications, publisher = Physica-Verlag, year = 1998, address = Heidelberg, series = Studies in Fuzziness and Soft Computing, volume = 18
8. Bazan, J. G., Hung Son Nguyen, Sinh Hoa Nguyen, Synak, P. and Wróblewski, J.: Rough set algorithms in classification problems. In: Polkowski, L., Lin, T. Y. and Tsumoto, S. (eds.) Rough Set Methods and Applications: New Developments in Knowledge Discovery in Information Systems, Series Studies in Fuzziness and Soft Computing, vol 56, pp. 49–88. Springer-Verlag/Physica-Verlag, Heidelberg (2000)
9. Blackburn, P., de Rijke, M., Venema, Y.: Modal Logic. Cambridge Univ. Press (2001)
10. Boicescu, V., Filipoiu, A., Georgescu, G., Rudeano, S.: Łukasiewicz-Moisil Algebras. North Holland, Amsterdam (1991)
11. Brown, F.M.: Boolean Reasoning. Kluwer Academic Publishers, Dordrecht, The Netherlands (1990)
12. Cattaneo, G.: Abstract approximation spaces for rough theories. In: Polkowski, L. and Skowron, A. (eds.) Rough Sets in Knowledge Discovery 1: Methodology and Applications, pp. 59–98. Physica-Verlag, Heidelberg (1998)
13. Chuchro, M.: On rough sets in topological Boolean algebras. In: Ziarko, W. P. (ed.) Rough Sets, Fuzzy Sets and Knowledge Discovery (Proceedings RSKD'93), pp. 157–160. Springer Verlag, London (1994)
14. Cignoli, R.: Representation of Łukasiewicz and Post algebras by continuous functions. Colloq. Math. **24**(2), 127–138 (1972)
15. Comer, S.: An algebraic approach to the approximation of information. Fundamenta Informaticae **14**, 492–502 (1991)
16. Comer, S.: Perfect extensions of regular double Stone algebras. Algebra Universalis **34**, 96–109 (1995)
17. Düntsch, I.: Rough sets and algebras of relations. In: Orłowska, E. (ed.) Incomplete Information: Rough Set Analysis, pp. 95–108. Physica-Verlag (1998)
18. Epstein, G., Horn, A.: Chain based lattices. J. of Mathematics **55**(1), 65–84 (1974) Reprinted In: Rine, D. C. (ed.) Computer Science and Multiple-valued Logic. Theory and Applications, pp. 58–76. North Holland (1991)
19. Garcia-Molina, H., Ullman, J.D. and Widom, J.D.: Database Systems: The Complete Book. Prentice Hall, Upper Saddle River, New Jersey (2002)
20. Grzymała-Busse, J.W.: LERS – A System for Learning from Examples based on Rough Sets. In: Słowiński, R. (ed.) Intelligent Decision Support – Handbook of Applications and Advances of the Rough Sets Theory, Series System Theory, Knowledge Engineering and Problem Solving, vol 11, pp. 3–18. Kluwer Academic Publishers, Dordrecht, The Netherlands (1992)
21. Grzymała-Busse, J.W.: LERS – A Data Mining System. In: Oded Maimon and Lior Rokach (eds.) The Data Mining and Knowledge Discovery Handbook, pp. 1347–1351. Springer, Heidelberg (2005)
22. Grzymała-Busse, J. W. and Rzasa, W.: Local and global approximations for incomplete data. In: Peters, J. F. and Skowron, A. (eds.) Transactions on Rough Sets VIII: Journal Subline, Series Lecture Notes in Computer Science, vol 4100, pp. 21–34. Springer, Heidelberg (2008)
23. Hastie, T., Tibshirani, R. and Friedman, J. H.: The Elements of Statistical Learning: Data Mining, Inference, and Prediction. Springer-Verlag, Heidelberg (2001)
24. He, X. and Xu, L. and Shen, W.: Dynamic information system and its rough set model based on time sequence. In: Yan-Qing Zhang and Lin, T.Y. (eds.) Proceedings of the IEEE International Conference on Granular Computing, Atlanta, May 10-12, 2006, pp. 542-545. IEEE Press (2006)

25. Higgs, D.: A category approach to Boolean-valued set theory. Preprint, University of Waterloo (1973)
26. Iturrioz, L.: Rough sets and three-valued structures. In: Orłowska, E. (ed.) Logic at Work: Essays Dedicated to the Memory of Helena Rasiowa, pp. 596–603. Physica-Verlag (1999)
27. Iwiński, T. B.: Algebraic approach to rough sets. Bull. Polish Acad. Sc. (Math.) **35**(9-10), 673–683 (1987)
28. Jinfu Liu, Qinghua Hu and Daren Yu: A comparative study on rough set based class imbalance learning. Knowledge-Based Systems **21**, 753–763 (2008)
29. Kryszkiewicz, M.: Rough set approach to incomplete information systems. Information Sciences **112**, 39–49 (1998)
30. Kryszkiewicz, M.: Rules in incomplete information systems. Information Sciences **113**, 271–292 (1999)
31. Kuroki, N.: Rough ideals in semigroups. Information Sciences **100**, 139–163 (1997)
32. Łukasiewicz, J.: Die logischen Grundlagen der Wahrscheinlichkeitsrechnung, 1913. In: Borkowski, L. (ed.) Jan Łukasiewicz: Selected Works. pp. 16–63. North Holland Publishing Company, Amsterdam (1970)
33. Moisil, Gr. C.: Sur les idéaux des algébres łukasiewicziennes trivalentes. An. Univ. C.I. Parhon, Acta Logica **3**, 83–95 (1960)
34. Monteiro, A.: Construction des algebres de Łukasiewicz trivalentes dans les algebres de Boole monadiques. I. Math. Japonicae **12**, 1–23 (1967)
35. Mundici, D.: The C^*-algebras of three-valued logic. In: Ferro, R. et al. (eds.) Logic Colloquium '88, pp. 61–77. North Holland, Amsterdam (1989)
36. Oded Maimon and Lior Rokach (eds.): The Data Mining and Knowledge Discovery Handbook. Springer, Heidelberg (2005)
37. Nguyen, H.S.: Approximate Boolean reasoning: Foundations and Applications in Data Mining. In: Peters, J. F. and Skowron, A. (eds.) Transactions on Rough Sets V: Journal Subline, Series Lecture Notes in Computer Science, vol 4100, pp. 344–523. Springer, Heidelberg (2006)
38. Nguyen, H.S. and Nguyen, S.H.: Rough sets and association rule generation. Fundamenta Informaticae **40**(4), 383–405 (1999)
39. Nguyen, H.S. and Skowron, A.: Rough Sets: From Rudiments to Challenges. In: Skowron, A. and Suraj, Z. (eds.) Rough Sets and Intelligent Systems. Professor Zdzisław Pawlak in Memoriam, Series Intelligent Systems Reference Library, vol 42-43, pp. 75–173. Springer, Heidelberg (2013)
40. Nguyen, H.S. and Ślezak, D.: Approximate Reducts and Association Rules - Correspondence and Complexity Results. In: Skowron, A., Setsuo Ohsuga and Zhong, N. (eds.) Proceedings of the 7-th International Workshop on Rough Sets, Fuzzy Sets, Data Mining, and Granular-Soft Computing (RSFDGrC'1999), Yamaguchi, November 9-11, 1999, Series Lecture Notes in Artificial Intelligence, vol 1711, pp. 137–145. Springer, Heidelberg (1999)
41. Orłowska, E.: Dynamic information system. Fundamenta Informaticae **5**, 101–118 (1982)
42. Orłowska, E.: Introduction: what you always wanted to know about rough sets. In: Orłowska, E. (ed.) Incomplete Information: Rough Set Analysis, pp. 1–20. Physica-Verlag (1998)
43. Orłowska, E.: Many-valuedness and uncertainty. International Journal of Multiple Valued Logic **4**, 207–277 (1999)
44. Orłowska, E. and Pawlak, Z.: Representation of nondeterministic information. Theoretical Computer Science **29**, 27–39 (1984)
45. Pawlak, Z.: Information systems - theoretical foundations. Information Systems **6**, 205–218 (1981)
46. Peters, J. F. and Skowron, A. (eds.): Transactions on Rough Sets V: Journal Subline, Series Lecture Notes in Computer Science, vol 4100. Springer, Heidelberg (2006)
47. Peters, G., Lingras, P., Ślezak, D. and Yao, Y. (eds.): Rough Sets: Selected Methods and Applications in Management and Engineering. Springer, London (2012)
48. Polkowski, L., Lin, T. Y. and Tsumoto, S. (eds.): Rough Set Methods and Applications: New Developments in Knowledge Discovery in Information Systems, Series Studies in Fuzziness and Soft Computing, vol 56. Springer-Verlag/Physica-Verlag, Heidelberg (2000)

49. Pomykała, J. A.: Approximation, similarity and rough construction. Preprint, ILLC Prepublication Series CT–93–07, University of Amsterdam (1993)
50. Pomykała, J. and Pomykała, J. A.: The Stone algebra of rough sets. Bull. Polish Acad. Sc. (Math.) **36**, 495–508 (1988)
51. Rasiowa, H.: An Algebraic Approach to Non-classical Logics. North Holland, Amsterdam (1974)
52. Rissanen, J.: Modeling by shortest data description. Automatica **14**, 465–471 (1978)
53. Rissanen, J.: Minimum-description-length principle. In: Kotz, S. and Johnson, N. (eds.) Encyclopedia of Statistical Sciences, pp. 523–527. John Wiley & Sons, New York, NY (1985)
54. Roman, S. (ed.): Introduction to Information and Coding Theory. Springer, Heidelberg (1997)
55. Sikorski, R.: Boolean Algebras. Springer-Verlag, New York (1969)
56. Skowron, A.: Boolean Reasoning for Decision Rules Generation. In: Jan Komorowski, J. and Raś, Z.W. (eds.) Proceedings of ISMIS'1993, Trondheim, Norway, June 15-18, 1993, Series Lecture Notes in Artificial Intelligence, vol 689, pp. 295–305. Springer-Verlag, Heidelberg (1993)
57. Skowron, A. and Grzymała-Busse, J. W.: From Rough Set Theory to Evidence Theory. In: Yager R., Fedrizzi, M. and Kacprzyk J. (eds.) Advances in the Dempster-Shafer Theory of Evidence, pp. 193–236. John Wiley & Sons, New York, NY (1994)
58. Skowron, A.: Extracting laws from decision tables. Computational Intelligence: An International Journal **11**(2), 371–388 (1995)
59. Skowron, A. and Suraj, Z. (eds.): Rough Sets and Intelligent Systems. Professor Zdzisław Pawlak in Memoriam, Series Intelligent Systems Reference Library, vol 42-43. Springer, Heidelberg (2013)
60. Ślezak, D.: Approximate reducts in decision tables. In: Proceedings Sixth International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems IPMU'1996, Granada, Spain, July 1-5, 1996, vol III, pp. 1159–1164 (1996)
61. Ślezak, D.: Various approaches to reasoning with frequency-based decision reducts: A survey. In: Polkowski, L., Lin, T. Y. and Tsumoto, S. (eds.) Rough Set Methods and Applications: New Developments in Knowledge Discovery in Information Systems, Series Studies in Fuzziness and Soft Computing, vol 56., pp. 235–285. Springer-Verlag/Physica-Verlag, Heidelberg (2000)
62. Ślezak, D.: Normalized decision functions and measures for inconsistent decision tables analysis. Fundamenta Informaticae **44**(3), 291–319 (2000)
63. Ślezak, D.: Approximate Entropy Reducts. Fundamenta Informaticae **53**(3-4), 365–387 (2002)
64. Słowiński, R. (ed.): Intelligent Decision Support - Handbook of Applications and Advances of the Rough Sets Theory. Series System Theory, Knowledge Engineering and Problem Solving, vol 11. Kluwer Academic Publishers, Dordrecht, The Netherlands (1992)
65. Stefanowski, J. and Tsoukiàs, A.: On the Extension of Rough Sets under Incomplete Information. In: Zhong, N., Skowron, A. and Ohsuga, S. (eds.) RSFDGrC '99: Proceedings of the 7th International Workshop on New Directions in Rough Sets, Data Mining, and Granular-Soft Computing, Yamaguchi, Japan, November 9-11, 1999, Series Lecture Notes in Artificial Intelligence, vol 1711, pp. 73–81. Springer-Verlag, Berlin Heidelberg (1999)
66. Vakarelov, D.: Notes on $\mathscr{N}$-lattices and constructive logic with strong negation. Studia Logica **XXXVI**, 109–125 (1977)
67. Wasilewska, A., Vigneron, L.: Rough equality algebras. In: Wang, P. P. (ed.) Proceedings of the International Workshop on Rough Sets and Soft Computing (JCIS'95), pp. 26–30 (1995)
68. Wasilewska, A., Vigneron, L.: On generalized rough sets. Preprint
69. Wiweger, A.: On topological rough sets. Bull. Polish Acad. Sc. (Math.) **37**, 51–62 (1988)
70. Xiangjun Li and Fen Rao: An Rough Entropy Based Approach to Outlier Detection. Journal of Computational Information Systems **8**(24), 10501–10508 (2012)
71. Yao, Y.: Constructive and algebraic methods of the theory of rough sets. Information Sciences **109**, 21–47 (1998)

72. Yao, Y.: Generalized rough set models. In: Polkowski, L., Skowron, A. (eds.) Rough Sets in Knowledge Discovery 1: Methodology and Applications, pp. 286–318. Physica-Verlag, Heidelberg (1998)
73. Zheng Zheng and Guoyin Wang: RRIA: A Rough Set and Rule Tree Based Incremental Knowledge Acquisition Algorithm. Fundamenta Informaticae **59**(2-3), 299–313 (2004)

Chapter 7
Topological Aspects of Rough Sets

7.1 Some introductory remarks on topological spaces

In this section, topologies connected with rough sets are discussed. For an introduction to topology, any standard text book is recommended. However, the minimal prerequisites are re-presented here for the convenience of readers.

Definition 7.1. Let X be a non-empty set. A *topology* τ on X is a collection of subsets of X such that the following axioms are satisfied.

(i) $\emptyset, X \in \tau$

(ii) $A, B \in \tau$ implies $A \cap B \in \tau$

(iii) If $A_i \in \tau$ where $i \in I$, an index set, then $\bigcup_{i \in I} A_i \in \tau$.

In ordinary language, conditions (ii) and (iii) respectively state that intersection of finitely many members of τ belongs to τ, while union of arbitrarily many members of τ belongs to τ.
If τ is a topology on X then the pair (X, τ) is called a *topological space*.

Example 7.1. **

In the following, let (X, τ) be a topological space.

Definition 7.2. The elements of τ are called the *open sets* of (X, τ).
A subset A of X is said to be *closed* if and only if its complement A^c in X is open.

Definition 7.3. Let $A \subseteq X$. The union of all open sets contained in A is called the *interior* of A, i.e.

$$Int_\tau(A) := \bigcup \{O \subseteq X \ : \ O \in \tau \text{ and } O \subseteq A\}.$$

Definition 7.4. For $A \subseteq X$, the intersection of all closed sets containing A is called the *closure* of A, i.e.

$$Cl_\tau(A) := \bigcap \{C \subseteq X \ : \ C \text{ is a closed set and } A \subseteq C\}.$$

Note 7.1. Int_τ and Cl_τ may be considered as unary operators on $\mathscr{P}(X)$, the power set of X.

Exercise. Show that

1. $Int_\tau(A)$ is an open set.
2. $Cl_\tau(A)$ is a closed set.
3. $Cl_\tau(A^c) = (Int_\tau(A))^c$.
4. $Int_\tau(A^c) = (Cl_\tau(A))^c$.

The following properties of the closure and interior operators may be established.

Theorem 7.1. *The closure operator Cl_τ satisfies the following conditions.*

Cl1: $A \subseteq Cl_\tau(A)$.
Cl2: $Cl_\tau(A \cup B) = Cl_\tau(A) \cup Cl_\tau(B)$.
Cl3: $Cl_\tau Cl_\tau(A) = Cl_\tau(A)$.
Cl4: $Cl_\tau(\emptyset) = \emptyset$.

Theorem 7.2. *The interior operator Int_τ satisfies the following conditions.*

Int1: $Int_\tau(A) \subseteq A$.
Int2: $Int_\tau(A \cap B) = Int_\tau(A) \cap Int_\tau(B)$.
Int3: $Int_\tau Int_\tau(A) = Int_\tau(A)$.
Int4: $Int_\tau(X) = X$.

Either of the above two theorems gives a characterization of topological spaces. Let the following conditions, known as *Kuratowski's closure axioms*, be considered.

Kuratowski's closure axioms:

Let X be any set and $Cl : \mathscr{P}(X) \to \mathscr{P}(X)$ be a map satisfying the following.

Cl_1: $A \subseteq Cl(A)$.
Cl_2: $Cl(A \cup B) = Cl(A) \cup Cl(B)$.
Cl_3: $ClCl(A) = Cl(A)$.
Cl_4: $Cl(\emptyset) = \emptyset$.

Cl is called a *closure operator* on X.

Similarly, let $Int : \mathscr{P}(X) \to \mathscr{P}(X)$ be a map satisfying the following.

Int_1: $Int(A) \subseteq A$.
Int_2: $Int(A \cap B) = Int(A) \cap Int(B)$.
Int_3: $IntInt(A) = Int(A)$.
Int_4: $Int(X) = X$.

Exercise. Prove the following.

1. $A \subseteq B$ implies $Cl(A) \subseteq Cl(B)$, $Int(A) \subseteq Int(B)$.
2. $B \subseteq Cl(A)$ implies $Cl(B) \subseteq Cl(A)$.
3. $Int(A) \subseteq B$ implies $Int(A) \subseteq Int(B)$.

Theorem 7.3.
(i) If an operator Cl satisfies conditions $Cl_1 - Cl_4$ *then the operator* $Int : \mathscr{P}(X) \to \mathscr{P}(X)$ *defined as:*
$$Int(A) := (Cl(A^c))^c,$$
satisfies the conditions $Int_1 - Int_4$.
(ii) If an operator Int satisfies conditions $Int_1 - Int_4$ *then the operator* $Cl : \mathscr{P}(X) \to \mathscr{P}(X)$ *defined as:*
$$Cl(A) := (Int(A^c))^c,$$
satisfies the conditions $Cl_1 - Cl_4$.

Theorem 7.4. *Let* $Int : \mathscr{P}(X) \to \mathscr{P}(X)$ *satisfy* $Int_1 - Int_4$. *Let*
$$\tau_{Int} := \{A \in \mathscr{P}(X) \ : \ Int(A) = A\}.$$
Then τ_{Int} *is a topology on* X.

Corollary 7.1. *The closed sets of the topological space* (X, τ_{Int}) *are the sets A such that* $A^c \in \tau_{Int}$, *i.e.* $Int(A^c) = A^c$, *in other words,* $A = (Int(A^c))^c$.

Theorem 7.5. *Let* $Cl : \mathscr{P}(X) \to \mathscr{P}(X)$ *satisfy* $Cl_1 - Cl_4$. *Let*
$$\tau_{Cl} := \{A \in \mathscr{P}(X) \ : \ Cl(A^c) = A^c\}.$$
Then τ_{Cl} *is a topology on* X.

Corollary 7.2. *The closed sets of the topological space* (X, τ_{Cl}) *are the sets A such that* $Cl(A) = A$.

Exercise. Show that

1. $Int_{\tau_{Int}} = Int$.
2. $Cl_{\tau_{Cl}} = Cl$.
3. $\tau_{Int_\tau} = \tau$.
4. $\tau_{Cl_\tau} = \tau$.

Definition 7.5. A topological space (X, τ) is an *Alexandrov space* if and only if $A_i \in \tau$ implies $\cap A_i \in \tau$, where $i \in I$, any index set.

So a topological space is Alexandrov, if the intersection of an arbitrary family of open sets in the topology is also open. There are various characterizations of Alexandrov spaces available in literature. For instance, one can show the following.

Exercise. A topological space (X, τ) is Alexandrov if and only if the union of an arbitrary family of closed sets of (X, τ) is also closed.

Recall that a *0-dimensional topology* is one in which every open set is closed and every closed set is open.

Exercise.

1. Let *Int* be an interior operator defined on a set X such that it distributes over arbitrary intersection of subsets. Show that (X, Int_τ) is Alexandrov.
   Conversely, if (X, τ) is Alexandrov then show that τ_{Int} distributes over arbitrary intersection.

2. Let Cl be an closure operator defined on a set X such that it distributes over arbitrary union of subsets. Show that (X, Cl_τ) is Alexandrov.
   Conversely, if (X,τ) is Alexandrov then show that τ_{Cl} distributes over arbitrary union.
3. Prove that a 0-dimensional topological space is Alexandrov. Prove also that the converse is not true.

7.2 Binary relations and topological properties

Let R be a binary relation on a set X. Two operators $\underline{R}$ and $\overline{R}$, also called the *upper approximation* and *lower approximation* operators respectively, may be defined on $\mathscr{P}(X)$, by

$$\underline{R}(A) := \{x \in X \ : \ R_x \subseteq A\} \text{ and} \overline{R}(A) := \{x \in X \ : \ R_x \cap A \neq \emptyset\},$$

where $A \subseteq X$ and $R_x := \{y \in X \ : \ xRy\}$.

Note 7.2. Corresponding to the binary relation R on X, we have a function that, by abuse of notation, may also be denoted R, viz. $R : X \to \mathscr{P}(X)$ such that $R(x) := R_x$, for any $x \in X$.
We deviate here from the earlier notation for lower and upper approximations, viz. $\underline{A}_R, \overline{A}_R$, and write $\underline{R}(A), \overline{R}(A)$.

The following proposition states some properties of the operators $\underline{R}$ and $\overline{R}$.

Proposition 7.1.

1. $\underline{R}(A) = (\overline{R}(A^c))^c$	1′. $\overline{R}(A) = (\underline{R}(A^c))^c$
2. $\underline{R}(X) = X$	2′. $\overline{R}(\emptyset) = \emptyset$
3. $\underline{R}(\bigcap_{i\in I} A_i) = \bigcap_{i\in I} \underline{R}(A_i)$	3′. $\overline{R}(\bigcup_{i\in I} A_i) = \bigcup_{i\in I} \overline{R}(A_i)$
4. $A \subseteq B \Rightarrow \underline{R}(A) \subseteq \underline{R}(B)$	4′. $A \subseteq B \Rightarrow \overline{R}(A) \subseteq \overline{R}(B)$
5. $\underline{R}(\bigcup_{i\in I} A_i) \supseteq \bigcup_{i\in I} \underline{R}(A_i)$	5′. $\overline{R}(\bigcap_{i\in I} A_i) \subseteq \bigcap_{i\in I} \overline{R}(A_i)$,

where I is an index set.

Proof. The proofs are straightforward. Nonetheless, we show the proof of 3.

$$\begin{aligned} x \in \underline{R}(\textstyle\bigcap_{i\in I} A_i) &\text{ if and only if } R_x \subseteq \textstyle\bigcap_{i\in I} A_i \\ &\text{ if and only if } R_x \subseteq A_i \text{ for each } i \\ &\text{ if and only if } x \in \underline{R}(A_i) \text{ for each } i \\ &\text{ if and only if } x \in \textstyle\bigcap_{i\in I} \underline{R}(A_i). \end{aligned}$$

□

Exercise. Show that xRy if and only if $x \in \overline{R}(\{y\})$.

Definition 7.6. A binary relation R on a set X is called *serial* if for all x in X there is y in X such that xRy.
R is called *Euclidean* if for all x,y,z in X, xRy and xRz imply yRz.

The following proposition establishes the connection between various properties of a relation and properties of the operators $\underline{R}$ and $\overline{R}$.

Proposition 7.2. *The statements under each of (i)-(v) below, are equivalent.*
(i) (a) R is serial.
(b) $\underline{R}(\emptyset) = \emptyset$.
(c) $\overline{R}(X) = X$.
(d) $\underline{R}(A) \subseteq \overline{R}(A)$ *for all A.*
(ii) (a) R is reflexive.
(b) $\underline{R}(A) \subseteq A$.
(c) $A \subseteq \overline{R}(A)$ *for all A.*
(iii) (a) R is symmetric.
(b) $\overline{R}\,\underline{R}(A) \subseteq A$.
(c) $A \subseteq \underline{R}\,\overline{R}(A)$ *for all A.*
(iv) (a) R is transitive.
(b) $\underline{R}(A) \subseteq \underline{R}\,\underline{R}(A)$.
(c) $\overline{R}\,\overline{R}(A) \subseteq \overline{R}(A)$ *for all A.*
(v) (a) R is Euclidean.
(b) $\overline{R}\,\underline{R}(A) \subseteq \underline{R}(A)$.
(c) $\overline{R}(A) \subseteq \underline{R}\,\overline{R}(A)$ *for all A.*

Definition 7.7. A binary relation R on X is said to be a *pre-order* if R is reflexive and transitive.

Proposition 7.3. *Let X be a non-empty set and R a binary relation on X. Then the following statements are equivalent.*
(i) R is a pre-order.
(ii) $\overline{R}$ *is a closure operator.*
(iii) $\underline{R}$ *is an interior operator.*

Proof. The proof takes the route: $(i) \Rightarrow (ii) \Rightarrow (iii) \Rightarrow (i)$.
$(i) \Rightarrow (ii)$: Let $x \in A$. Then $R_x \cap A \neq \emptyset$ since R is reflexive. So, $x \in \overline{R}(A)$. Thus $A \subseteq \overline{R}(A)$.
Now we show $\overline{R}(A \cup B) = \overline{R}(A) \cup \overline{R}(B)$.

$$\begin{aligned}\overline{R}(A\cup B) &= \{x \in X \ : \ R_x \cap (A\cup B) \neq \emptyset\}\\ &= \{x \in X \ : \ R_x \cap A \neq \emptyset\} \cup \{x \in X \ : \ R_x \cap B \neq \emptyset\}\\ &= \overline{R}(A) \cup \overline{R}(B).\end{aligned}$$

Let $x \in \overline{R}(\overline{R}(A))$. Then $R_x \cap \overline{R}(A) \neq \emptyset$. Now $y \in R_x \cap \overline{R}(A)$ implies that $y \in R_x$ and $y \in \overline{R}(A)$. That means xRy and $R_y \cap A \neq \emptyset$. That is, xRy and there exists $z \in A$ such that yRz. By the transitivity of R, xRz. So, $R_x \cap A \neq \emptyset$ and hence $x \in \overline{R}(A)$. So $\overline{R}(\overline{R}(A)) \subseteq$

$\overline{R}(A)$. As R is reflexive, $\overline{R}(A) \subseteq \overline{R}(\overline{R}(A))$. Hence $\overline{R}(A) = \overline{R}(\overline{R}(A))$.
$\overline{R}(\emptyset) = \emptyset$ from the description of upper approximation operator. So, the operator $\overline{R}$ is a closure operator.

$(ii) \Rightarrow (iii)$:
This is because $\underline{R}$ is the dual operation of $\overline{R}$ (dual of a closure operator is an interior operator and vice versa.)

$(iii) \Rightarrow (i)$:
Let the lower approximation operator $\underline{R}$ be an interior operator. That is, for any $A, B \subseteq X$,

$$\begin{aligned} \underline{R}(A) &\subseteq A \\ \underline{R}(A \cap B) &= \underline{R}(A) \cap \underline{R}(B) \\ \underline{R}(\underline{R}(A)) &= \underline{R}(A). \end{aligned}$$

We have to show that R is reflexive and transitive.
Consider any $x \in X$. By definition, $\underline{R}(R_x) = \{z \in X \ : \ R_z \subseteq R_x\}$. So $x \in \underline{R}(R_x) \subseteq R_x$, the last using the first condition above on the set R_x. Thus xRx.
Let xRy and yRz. Then $y \in R_x,\ z \in R_y$. By the third condition above, for any $A \subseteq X$, $x \in \underline{R}(\underline{R}(A))$ if and only if $x \in \underline{R}(A)$. That is, $R_x \subseteq \underline{R}(A)$ if and only if $R_x \subseteq A$. In particular, for $A = R_x$, we then have $R_x \subseteq \underline{R}(R_x)$. By definition of $\underline{R}$, as $y \in R_x$, $R_y \subseteq R_x$. Finally, as $z \in R_y$, we get $z \in R_x$, i.e. xRz.
Note that the first condition is enough to prove reflexivity of R, while only the third condition is required to prove transitivity of R. □

We now observe that a topology may be generated by some binary relations.

Proposition 7.4. *Define* $\tau_R := \{A \subseteq X \ : \ \underline{R}(A) = A\}$.

(i) For any reflexive relation R, τ_R is a topology on X.
(ii) Let R be a pre-order. Then
(a) $\tau_R = \{\underline{R}(A) \ : \ A \subseteq X\}$,
(b) $Int_{\tau_R} = \underline{R}$,
(c) $Cl_{\tau_R} = \overline{R}$,
(d) τ_R is an Alexandrov topology.

Proof. (i) As R is reflexive, $\underline{R}(\emptyset) = \emptyset$. By Proposition 7.1(2), we also have $\underline{R}(X) = X$. So $\emptyset,\ X \in \tau_R$.
Let $A, B \in \tau_R$. Then $A = \underline{R}(A),\ B = \underline{R}(B)$. Using Proposition 7.1(3), $\underline{R}(A \cap B) = \underline{R}(A) \cap \underline{R}(B) = A \cap B$. Hence $A \cap B \in \tau_R$.
Let $A_i \in \tau_R,\ i \in I$, an index set. We need to show that $\bigcup_{i \in I} A_i \in \tau_R$, i.e. $\bigcup_{i \in I} A_i = \underline{R}(\bigcup_{i \in I} A_i)$. Now, as R is reflexive, by Proposition 7.2(ii), $\underline{R}(\bigcup_{i \in I} A_i) \subseteq \bigcup_{i \in I} A_i$. By Proposition 7.1(5), $\underline{R}(\bigcup_{i \in I} A_i) \supseteq \bigcup_{i \in I} \underline{R}(A_i)$, and for each $i \in I$, $\underline{R}(A_i) = A_i$ as $A_i \in \tau_R$. Hence proved.

$(ii)(a)$ $\{A \subseteq X \ : \ \underline{R}(A) = A\} \subseteq \{\underline{R}(A) \ : \ A \subseteq X\}$. Now as R is reflexive, by Proposition 7.2(2), $\underline{R}(\underline{R}(A)) \subseteq \underline{R}(A)$. Transitivity of R gives $\underline{R}(A) \subseteq \underline{R}(\underline{R}(A)$ by Proposition 7.2(iv). Thus for any $A \subseteq X$, $\underline{R}(\underline{R}(A)) = \underline{R}(A)$. This gives $\{\underline{R}(A) \ : \ A \subseteq X\} \subseteq \{A \subseteq X \ : \ \underline{R}(A) = A\}$.

$(ii)(b)$ Let $A \subseteq X$. We have to show $Int_{\tau_R}(A) = \underline{R}(A)$. By Definition 7.3, $Int_{\tau_R}(A) := \bigcup\{B \subseteq X \ : \ B \in \tau_R$ and $B \subseteq A\}$. That is,

$$Int_{\tau_R}(A) = \bigcup\{B \subseteq X \ : \ \underline{R}(B) = B \text{ and } B \subseteq A\}. \quad (1)$$

Let $x \in Int_{\tau_R}(A)$. Then for some $B \subseteq A$, $x \in B$, $\underline{R}(B) = B$. So $x \in \underline{R}(B)$, i.e. $R_x \subseteq B$, whence $R_x \subseteq A$. Thus $x \in \underline{R}(A)$, by definition of $\underline{R}$.
Conversely, let $x \in \underline{R}(A)$. As R is a pre-order, $\underline{R}(A) \subseteq A$ and $\underline{R}(\underline{R}(A)) = \underline{R}(A)$ (as shown above in $(ii)(a)$). So $\underline{R}(A)$ is one of the B's in (1), and hence $x \in Int_{\tau_R}(A)$.

$(ii)(c)$ This follows from $(ii)(b)$ above, since Cl_{τ_R} is dual of Int_{τ_R} and $\overline{R}$ is the dual of $\underline{R}$.

$(ii)(d)$ By $(ii)(a)$ of this Proposition, open sets of τ_R are of the form $\underline{R}(A)$, $A \subseteq X$. By Proposition 7.1(3), $\bigcap_{i \in I} \underline{R}(A_i) = \underline{R}(\bigcap_{i \in I} A_i)$, and the right hand side is a member of τ_R. So the intersection of an arbitrary collection of open sets is also open, making τ_R Alexandrov. □

Let (X,R) be an approximation space, i.e. R is an equivalence relation. Consider τ_R, the topology generated by R (as given in Proposition 7.4). It may be observed that $\tau_R = \mathscr{D}$, the collection of all definable sets in (X,R). Therefore, we have the following by Proposition 7.4.

Theorem 7.6. *$(X,\mathscr{D})$ is an Alexandrov space.*

Exercise. Show that the interior of a set $A(\subseteq X)$ with respect to $\mathscr{D}$ is $\underline{R}(A)$ and the closure of A with respect to $\mathscr{D}$ is $\overline{R}(A)$.

We can prove an even stronger result.

Exercise. The topological space $(X,\mathscr{D})$ or (X,τ_R) is 0-dimensional.

We next pass on to generating relations from topologies.

Proposition 7.5. *Let (X,τ) be a topological space. Let a binary relation R_τ be defined on X by: $xR_\tau y$ if and only if $x \in Cl_\tau(\{y\})$. Then*
(i) R_τ is a pre-order.
(ii) If τ is 0-dimensional then R_τ is also symmetric.

Proof. (i) $xR_\tau x$ since $x \in Cl_\tau(\{x\})$. So, R_τ is reflexive.
Similarly, let $xR_\tau y$ and $yR_\tau z$. So, $x \in Cl_\tau\{y\}$ and $y \in Cl_\tau\{z\}$. Now $y \in Cl_\tau\{z\} \Rightarrow Cl_\tau\{y\} \subseteq Cl_\tau\{z\}$. So, $x \in Cl_\tau\{z\}$ and $xR_\tau z$. In other words, R_τ is transitive.

(ii) Let $xR_\tau y$, i.e. $x \in Cl_\tau\{y\}$. Let O_y be any open neighbourhood of y. Then $\{y\} \subseteq O_y$. Now, O_y is also a closed set since τ is 0-dimensional. So, $Cl_\tau\{y\} \subseteq O_y$. Hence $x \in O_y$. This means that any open neighbourhood of y intersects $\{x\}$. So, $y \in Cl_\tau\{x\}$. Hence $yR_\tau x$, and R is symmetric. □

So Proposition 7.5 says that if τ is a 0-dimensional topology then R_τ is an equivalence relation.

By Proposition 7.5, a given topological space (X,τ) induces a pre-order R_τ. As shown by Proposition 7.4, the pre-order R_τ generates an Alexandrov topology τ_{R_τ} on

X with $\overline{R_\tau}$ as the closure operator (equivalently, $\underline{R_\tau}$ as the interior operator). We now have Proposition 7.6 below that establishes a relation between τ_{R_τ} and the original topology τ.

Proposition 7.6. *The following assertions are equivalent.*

(i) $Cl_{\tau_{R_\tau}} = \overline{R_\tau} = Cl_\tau$.

(ii) $Int_{\tau_{R_\tau}} = \underline{R_\tau} = Int_\tau$.

(iii) τ *is Alexandrov.*

Proof. By Proposition 7.5(i), R_τ is a pre-order, and Proposition 7.4(ii)(b-c) give that $Int_{\tau_{R_\tau}} = \underline{R_\tau}$, $Cl_{\tau_{R_\tau}} = \overline{R_\tau}$.
$(i) \Leftrightarrow (ii)$ because of the dualities between $\overline{R_\tau}$ and $\underline{R_\tau}$ as well as Cl_τ and Int_τ.

$(i) \Rightarrow (iii)$. As R_τ is a pre-order, τ_{R_τ} is Alexandrov by Proposition 7.4(ii)(d). Assumption (i) implies that the topology τ_{R_τ} is the same as τ. So, τ is Alexandrov.

$(iii) \Rightarrow (i)$. Suppose the topology τ is Alexandrov. We need to establish that $\overline{R_\tau}(A) = Cl_\tau(A)$ for any $A \subseteq X$.
$x \in \overline{R_\tau}(A)$ implies $R_{\tau x} \cap A \neq \emptyset$, i.e. there exists $y \in A$ such that $xR_\tau y$. So $x \in Cl_\tau(\{y\})$, by definition of R_τ. As $\{y\} \subseteq A$, $x \in Cl_\tau(A)$, and therefore $\overline{R_\tau}(A) \subseteq Cl_\tau(A)$.
Conversely, let $x \in Cl_\tau(A)$. Then for all open sets O_i, $i \in I$ (I an index set) containing x, $O_i \cap A \neq \emptyset$. Since τ is Alexandrov, $\bigcap_{i\in I} O_i$ is also open and $x \in \bigcap_{i\in I} O_i$. So there exists $y \in (\bigcap_{i\in I} O_i) \cap A$, and all the open sets O_i containing x intersect $\{y\}$. Thus x is a limit point of $\{y\}$, and $x \in Cl_\tau(\{y\})$. By definition of R_τ, $xR_\tau y$. As $y \in A$, $x \in \overline{R_\tau}(A)$ and so $Cl_\tau(A) \subseteq \overline{R_\tau}(A)$. □

For any pre-order relation R on X, by Proposition 7.4, one obtains an Alexandrov topology τ_R on X. Using Proposition 7.5, the topology τ_R induces a pre-order R_{τ_R}. The following proposition shows that these two pre-order relations are identical. On the other hand, as a corollary to Proposition 7.6, one concludes that the topologies τ_{R_τ} and τ are identical when τ is Alexandrov.

Proposition 7.7.

(i) Let R *be a pre-order relation on* X*. Then* $R_{\tau_R} = R$.

(ii) If τ *is an Alexandrov topology on* X*, then* $\tau_{R_\tau} = \tau$.

Proof. (i) By Proposition 7.4(ii), $\tau_R = \{\underline{R}(A) \; : \; A \subseteq X\}$, and for any $A \subseteq X$, $Int_{\tau_R}(A) = \underline{R}(A)$, while $Cl_{\tau_R}(A) = \overline{R}(A)$. Using the definition of the relation R_{τ_R} (see Proposition 7.5), $xR_{\tau_R}y$ if and only if $x \in Cl_{\tau_R}(\{y\})$, which is if and only if $x \in \overline{R}(\{y\})$. As $\{y\})$ is a singleton, the last is if and only if xRy.

(ii) This directly follows from Proposition 7.6. □

Exercise.

1. Show that if $x \in Cl_\tau(\{y\})$ and $y \in Cl_\tau(\{x\})$ then $x = y$.
2. Show that if (X, τ) is 0-dimensional then for all $x, y \in X$, $x \in Cl_\tau(\{y\})$ implies $y \in Cl_\tau(\{x\})$.

7.3 Topological Rough Sets

Let (X,τ) be a topological space. Then for any subset A of X, $Int_\tau(A)$ and $Cl_\tau(A)$ are obtained. Let the relation $E(\tau)$ be defined on the power set $\mathscr{P}(X)$ by:

$$A\ E(\tau)\ B \text{ if and only if } Int_\tau(A) = Int_\tau(B) \text{ and } Cl_\tau(A) = Cl_\tau(B).$$

$E(\tau)$ is an equivalence relation.

Definition 7.8. A *topological rough set* in (X,τ) is a member of the quotient set $\mathscr{P}(X)/E(\tau)$.

Observation 7 *In case of the topological space $(X,\mathscr{D})$=(X,τ_R) generated by an approximation space (X,R), the equivalence relation $E(\tau_R)$ is just the relation of rough equality (refer to Theorem 7.6 and Proposition 7.4). Thus topological rough sets in $(X,\mathscr{D})$ are rough sets according to Definition 2.11 as given in Chapter 2.*

Definition 7.9. In a topological space (X,τ), a pair (M,N), where $M \subseteq X,\ N \subseteq X$, is called a *rough pair* if and only if the following conditions are satisfied.

r_1: M is open.
r_2: N is closed.
r_3: $M \subseteq N$.
r_4: The set $N \setminus Cl_\tau(M)$ contains a subset Z which is dense in $N \setminus Cl_\tau(M)$ and co-dense in X, i.e.
 (i) $Int_\tau(Z) = \emptyset$,
 (ii) $Z \subseteq N \setminus Cl_\tau(M)$,
 (iii) $N \setminus Cl_\tau(M) \subseteq Cl_\tau(Z)$.

The set of all rough pairs in (X,τ) shall be denoted by $RP(X,\tau)$.

Proposition 7.8. *Given a topological space (X,τ), for any subset A of X, the pair $(Int_\tau(A),Cl_\tau(A))$ is a rough pair.*

Proof. Conditions (r_1), (r_2) and (r_3) follow from the definitions of interior and closure of a set. We need to prove (r_4).
Consider $Z := A \setminus Cl_\tau(Int_\tau(A))$. $Int_\tau(Z) = \emptyset$ if and only if there is no non-empty open subset of Z. Now if $G \subseteq Z$ and G is open, then $G \subseteq A$ and no element in G is in the closure of $Int_\tau(A)$. So $G \cap Int_\tau(A) = \emptyset$. This is possible only when $G = \emptyset$. Thus condition (i) holds.
Also $Z := A \setminus Cl_\tau(Int_\tau(A)) \subseteq Cl_\tau(A) \setminus Cl_\tau(Int_\tau(A))$. So condition (ii) holds.
For (iii), let $a \in Cl_\tau(A) \setminus Cl_\tau(Int_\tau(A))$, i.e. $a \in Cl_\tau(A)$, $a \notin Cl_\tau(Int_\tau(A))$. We get two cases.
Case 1: $a \in A$. Then $a \in Z$, and so $a \in Cl_\tau(Z)$.
Case 2: $a \notin A$. Consider an arbitrary open set G such that $a \in G$ (there is at least one, viz. the whole set). Now $G \setminus Cl_\tau(Int_\tau(A))$ is open, since for any sets P,Q, if P is open and Q is closed then $P \setminus Q = P \cap Q^c$ is open. As $a \in G$ and $a \notin Cl_\tau(Int_\tau(A))$, $a \in G \setminus Cl_\tau(Int_\tau(A))$ which is open. Also $a \in Cl_\tau(A)$. So $(G \setminus Cl_\tau(Int_\tau(A))) \cap A \neq \emptyset$. Thus there exists $x \in A$ such that $x \notin Cl_\tau(Int_\tau(A))$, i.e. $x \in A \setminus Cl_\tau(Int_\tau(A)) = Z$. So any open set G containing a has a non-empty intersection with Z, and thus $a \in Cl_\tau(Z)$. □

Proposition 7.9. *Given* (X,τ)*, for any rough pair* (M,N)*, there exists a subset A of X such that* $M = Int_\tau(A)$ *and* $N = Cl_\tau(A))$*.*

Proof. Let (M,N) be a rough pair. Then there exists Z satisfying $(r_4)(i)$, (ii), (iii) of Definition 7.9. Consider $A = M \cup Z$. So M is open by (r_1) and $M \subseteq A$. This implies $M \subseteq Int_\tau(A)$. (I)
If G is an open set contained in A then $G \setminus Cl_\tau(M)$ is an open set and contained in A. $G \setminus Cl_\tau(M) \nsubseteq Cl_\tau(M)$. Hence $G \setminus Cl_\tau(M) \nsubseteq M$, and so $G \setminus Cl_\tau(M) \subseteq Z$. By (i) therefore, $G \setminus Cl_\tau(M) = \emptyset$. This means $G \subseteq Cl_\tau(M)$. Since $G \subseteq M \cup Z$, by (ii) we get $G \subseteq M$. In particular,
$Int_\tau(A) \subseteq M$. (II)
From (I) and (II) we get $Int_\tau(A) = M$.
Also $Cl_\tau(A) = Cl_\tau(M \cup Z) = Cl_\tau(M) \cup Cl_\tau(Z) \subseteq N \cup Cl_\tau(Z)$, by (r_2) and (r_3). By (ii), $Z \subseteq N$ and N is closed, so $Cl_\tau(Z) \subseteq Z$. Thus
$N \cup Cl_\tau(Z) \subseteq N$. (III)
On the other hand, by (r_2) and (r_3) $N = Cl_\tau(M) \cup (N \setminus Cl_\tau(M))$. So, as N is closed, $N = Cl_\tau(N) = Cl_\tau Cl_\tau(M) \cup Cl_\tau(N \setminus Cl_\tau(M)) = Cl_\tau(M) \cup Cl_\tau(N \setminus Cl_\tau(M)) \subseteq Cl_\tau(M) \cup Cl_\tau(Z)$, by (iii). So,
$N = Cl_\tau(A)$. (IV)
From (III) and (IV), we get $N = Cl_\tau(A)$.
This completes the proof. □

Propositions 7.8 and 7.9 together imply the following for the set $RP(X,\tau)$ of all rough pairs in (X,τ).

Corollary 7.3. $RP(X,\tau) = \{(Int_\tau(A), Cl_\tau(A)) : A \subseteq X\}$.

Observation 8 *Rough pairs in the topological space* $(X,\mathscr{D})$=(X,τ_R) *corresponding to the approximation space* (X,R)*, are rough sets according to Definition 2.8 as given in Chapter 2.*

We now define a mapping $F : \mathscr{P}(X)/E(\tau) \to RP(X,\tau)$ by:

$$F([A]_{E(\tau)}) := (Int_\tau(A), Cl_\tau(A)).$$

Clearly the definition is unambiguous.

Proposition 7.10. *The mapping F defined above is a bijection.*

The proof is straightforward.

Corollary 7.4. *Pawlakian rough sets in* (X,R) *are exactly the same as topological rough sets on the 0-dimensional topological space* $(X,\mathscr{D})$*.*

7.4 Tolerance relation, tolerance topology and approximation operators

A reflexive and symmetric relation is called a *tolerance relation*. A pair (X,R) is called a *tolerance space*, if X is a non-empty set and R is a tolerance relation on X.

Definition 7.10. A subset A of X is said to be a *tolerance class* if and only if any two elements of A are mutually R-related and it is maximal in this respect. This means, for any $x \in X,\ x \notin A$, there exists at least one element $a \in A$ such that xRa does not hold.

Obviously, every element $x \in X$ is included in at least one tolerance class – there may be more than one.

Example 7.2. Consider a tolerance space (X,R), where $X := \{1,2,3,4,5,6,7,8,9\}$ and R is as given in Figure 7.1.

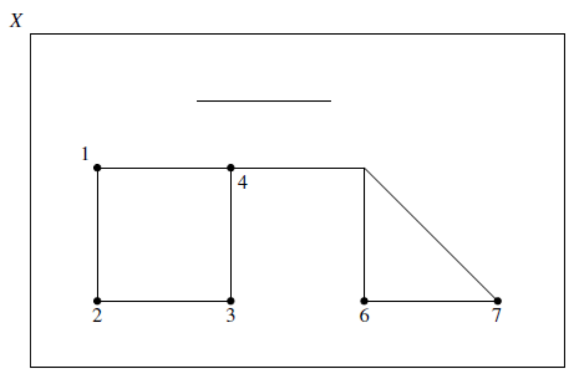


Fig. 7.1 Tolerance classes

The tolerance classes here are $\{1,2,3,4\},\{4,5\},\{5,6,7\},\{8,9\}$.

Let (X,R) be any tolerance space. We have the following.

Proposition 7.11. *For each $x \in X$, the set $R_x = \{y \in X : xRy\}$ is a τ-open set.*

Proof. R_x is the union of all tolerance classes containing x. So R_x is τ-open. □

For the tolerance relation R, consider the lower and upper approximations $\underline{R}(A)$ and $\overline{R}(A)$ for any subset A of X. We state some propositions below, the proofs of which are left as exercises.

Proposition 7.12. $\underline{R}(A) \subseteq Int_\tau(A) \subseteq A \subseteq Cl_\tau(A) \subseteq \overline{R}(A)$.

Questions: Is $\underline{R}(A)$ τ-open? Is $\overline{R}(A)$ τ-closed?
These are not yet settled. However, the following equations hold.

Proposition 7.13.
(i) $\underline{R}(A) = \underline{R}(Int_\tau(A))$.
(ii) $\overline{R}(A) = \overline{R}(Cl_\tau(A))$.

We can also establish the following.

Proposition 7.14.
(i) $\underline{R}(A) = Int_\tau(A)$ *if and only if* $\bigcup_{x \in Int_\tau(A)} R_x \subseteq Int_\tau(A)$.
(ii) $\overline{R}(A) = Cl_\tau(A)$ *if and only if* $Cl_\tau(A) \subseteq \bigcap_{x \in (Cl_\tau(A))^c} {R_x}^c$.

Let us now define *neighbourhood systems* and see the connection with tolerance spaces.

Definition 7.11 (Neighbourhood systems). [**?**](Kelley)
Let X be a non-empty set. For $x \in X$, a collection $\mathscr{N}_x$ of subsets of X is said to be a *neighbourhood system of x* if and only if the following conditions are satisfied.
(i) If $N \in \mathscr{N}_x$ then $x \in N$.
(ii) If $N_1, N_2 \in \mathscr{N}_x$ then $N_1 \cap N_2 \in \mathscr{N}_x$.
(iii) If $N_1 \in \mathscr{N}_x$ and $N_1 \subseteq N_2$ then $N_2 \in \mathscr{N}_x$.
(iv) If $N_1 \in \mathscr{N}_x$ then there is a member $N_2 \in \mathscr{N}_x$ such that $N_2 \subseteq N_1$ and for each $y \in N_2$, $N_2 \in \mathscr{N}_y$.
If for all $x \in X$ a family $\mathscr{N}_x$ is defined, the collection $\{\mathscr{N}_x : x \in X\}$ is called a *neighbourhood system*.

Note 7.3. Corresponding to a neighbourhood system $\{\mathscr{N}_x : x \in X\}$, we have a function $\mathscr{N} : X \to \mathscr{P}\mathscr{P}(X)$ such that $\mathscr{N}(x) := \mathscr{N}_x$, for any $x \in X$.

A topology may be generated in X by a neighbourhood system and vice versa. This is called the *neighbourhood definition* of a topological space. For further detail, we refer to Kelley [**?**].

Given a tolerance space (X, R), a neighbourhood system is obtained as in Theorem 7.7 below. For any $x \in X$, let us fix some notations:
K_x^R denotes the set of all tolerance classes,
Z_x^R denotes the set of all finite intersections of the elements of K_x^R, and
${\mathscr{N}_x}^R$ denotes the collection of all subsets N of X such that there exists at least one member of Z_x^R which is a subset of N.

Theorem 7.7. $\{{\mathscr{N}_x}^R\}_{x \in X}$ *forms a neighbourhood system.*

Proof. Exercise.

So a topology τ is generated in X by the neighbourhood system $\{{\mathscr{N}_x}^R\}_{x \in X}$, and is called *tolerance topology*. Again, taking the set $\bigcup_{x \in X} K_x^R$, that is the set of all tolerance classes as a subbasis and hence the set $\bigcup_{x \in X} Z_x^R$ as a basis, one can generate a topology τ' on X in which open sets are unions of elements belonging to the basis set. Obviously any tolerance class is a member of the basis.

Proposition 7.15. $\tau = \tau'$.

Proof. Exercise.

We shall see revisit the discussed notions in Section 7.6.

7.5 Connection with weaker topological structures

Let us recall the Kuratowski closure axioms (see Section 7.1). From Table **??** of Chapter **??**, we notice that covering-based systems P_4, C_2 and C_5 satisfy all the four axioms. So in these systems, the upper approximation operators are closure operators and hence generate topologies. To remark on the other systems in Table **??**, we first introduce a few definitions from [**?**] (Pagliani-Chak)**

Definition 7.12. A *pre-topological space* is a triple $(X, \bar{\cdot}, \underline{\cdot})$ such that
(i) X is a non-empty set,
(ii) $\bar{\cdot} : \mathscr{P}(X) \to \mathscr{P}(X)$ is an expansion map, i.e. $A \subseteq \overline{A}$ for all $A \subseteq X$,
(iii) $\overline{\emptyset} = \emptyset$,
(iv) $\underline{\cdot} : \mathscr{P}(X) \to \mathscr{P}(X)$ is the dual map of $\bar{\cdot} : \mathscr{P}(X) \to \mathscr{P}(X)$ and is a contraction map, i.e. $\underline{A} \subseteq A$ for all $A \subseteq X$.

One can observe that in a pre-topological space $(X, \bar{\cdot}, \underline{\cdot})$, $\underline{X} = X$ holds.

Definition 7.13. A pre-topological space $(X, \bar{\cdot}, \underline{\cdot})$ is said to be of *type T_{id}* if and only if the operator $\bar{\cdot}$ (equivalently $\underline{\cdot}$) is idempotent, i.e. $\overline{\overline{A}} = \overline{A}$ ($\underline{\underline{A}} = \underline{A}$) for all $A \subseteq X$.

Definition 7.14. A pre-topological space $(X, \bar{\cdot}, \underline{\cdot})$ is said to be of *type T_I* if and only if for all $A, B \subseteq X$, $A \subseteq B$ implies $\overline{A} \subseteq \overline{B}$ (equivalently $\underline{A} \subseteq \underline{B}$).

Definition 7.15. A pre-topological space $(X, \bar{\cdot}, \underline{\cdot})$ is said to be of *type T_D* if and only if for all $A, B \subseteq X$, $\overline{A \cup B} = \overline{A} \cup \overline{B}$ (equivalently $\underline{A \cap B} = \underline{A} \cap \underline{B}$).

Any topological space $(X, \bar{\cdot}, \underline{\cdot})$ is obviously pre-topological. Let X be a non-empty set with a covering $\mathscr{C} := \{C_i\}$. The pair $(X, \mathscr{C})$ is called a *covering system*. We observe that the covering systems $P_1, P_2, P_3, C_1, C_3, C_4$ and C_{Gr} all form pre-topological spaces. In fact, they may be characterised further.

Type T_{id} : $P_1, P_2, P_3, C_1, C_4, C_{Gr}$
Type T_I : All
Type T_D : P_1, C_4

We see that some covering systems such as P_2 are not of the type T_D since $\overline{A \cup B} \subseteq \overline{A} \cup \overline{B}$ does not hold, although the reverse, viz. $\overline{A} \cup \overline{B} \subseteq \overline{A \cup B}$ holds for all the systems. Interestingly, there exists a general treatment of such a situation by Tarski **.

In a covering system $(X, \mathscr{C})$, where $\mathscr{C} := \{C_i\}$, a subset A of X is said to be *pseudo open* if and only if $A = \bigcup_j C_{i_j}$, i.e. A is the union of some sets in $\mathscr{C}$. A is said to be *pseudo closed* if and only if $A = \bigcap_j C^c_{i_j}$, i.e. A is the intersection of the complements of some sets in $\mathscr{C}$.
Let $\mathbb{O}(X)$ and $\mathbb{C}(X)$ denote the sets of all pseudo open and pseudo closed sets respectively.
Exercise.
(i) Prove that $A \in \mathbb{C}(X)$ if and only if $A^c \in \mathbb{O}(X)$.
(ii) $\emptyset, X$ are both pseudo open and pseudo closed.

(iii) $\mathbb{O}(X)$ is closed with respect to arbitrary union, but nothing can be asserted about closure with respect to intersection (even finite intersection).
(iv) $\mathbb{C}(X)$ is closed with respect to arbitrary intersection, but nothing can be asserted about closure with respect to union (even finite union).

Let us now define lower and upper approximations of a set A relative to the covering system $(X, \mathscr{C})$ in the following way.

$$\underline{A} := \bigcup\{P \in \mathbb{O}(X) \text{ and } P \subseteq A\}$$
$$\overline{A} := \bigcap\{Q \in \mathbb{C}(X) \text{ and } A \subseteq Q\}.$$

Proposition 7.16. *For any $A, B \subseteq X$, the following hold.*
(i) (a) $\underline{X} = X$.
(b) $\underline{A} \subseteq A$.
(c) $\underline{A \cap B} \subseteq \underline{A} \cap \underline{B}$. *However, the converse does not hold in general.*
(d) $\underline{\underline{A}} = \underline{A}$.
(ii) Dually, we also have
(a) $\overline{\emptyset} = \emptyset$,
(b) $A \subseteq \overline{A}$,
(c) $\overline{A} \cup \overline{B} \subseteq \overline{A \cup B}$ *and*
(d) $\overline{\overline{A}} = \overline{A}$.
(iii) $\underline{A}$ *and* $\overline{A}$ *are duals.*
(iv) $A \subseteq B$ *implies* $\underline{A} \subseteq \underline{B}$, $\overline{A} \subseteq \overline{B}$.

So another interesting covering based rough set system is obtained by using pseudo open and pseudo closed sets. This lower-upper approximation pair satisfies the properties 1, 2, 3, 4, 7, 8, 9, 10, 11, 12, 15, 16 in Table **??** of Chapter **??**.

7.6 Interrelation of the three approaches

In Sections 7.2, 7.4 and 7.5, we have observed the roles of the mappings $R : X \to \mathscr{P}(X)$ (R being a binary relation on X), the neighbourhood system $\mathscr{N} : X \to \mathscr{P}\mathscr{P}(X)$ and the contraction map $\underline{\ } : \mathscr{P}(X) \to \mathscr{P}(X)$ (or equivalently its dual, the expansion map $\overline{\ } : \mathscr{P}(X) \to \mathscr{P}(X)$). All are connected with the generation of topologies or weaker structures on the base set X from different approaches. It may be interesting to probe into the interrelations among these maps. We deal with them with minimal conditions.

(I) The contraction-expansion function approach:
start with $(X, \underline{\ })$, where the contraction map $\underline{\ } : \mathscr{P}(X) \to \mathscr{P}(X)$ satisfies $\underline{A} \subseteq A$ for all A.
The expansion map $\overline{\ } : \mathscr{P}(X) \to \mathscr{P}(X)$ is defined dually by $\overline{A} = (\underline{A^c})^c$.
(II) The relational approach:
start with (X, R), where R is a binary relation which is reflexive. Define R_x as $R_x := \{y \in X : xRy\}$.

(III) Neighbourhood approach:
start with $(X, \mathscr{N})$, where $\mathscr{N} : X \to \mathscr{P}\mathscr{P}(X)$ is such that for all $A \in \mathscr{N}_x$ (the image of x), $x \in A$.

It is possible to establish a nice structure involving the three approaches, see Figure 7.2.

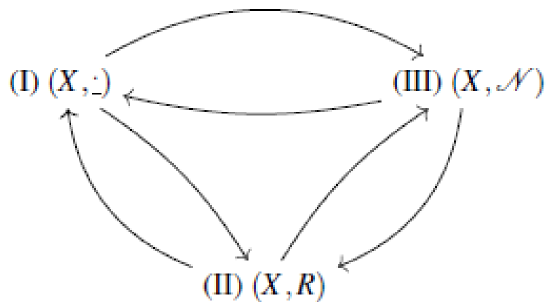


Fig. 7.2 Interrelationships between the 3 approaches

The arrows in Figure 7.2 are explained as follows.

(I) $\to$ (III): Given $(X, \underline{\cdot})$, define $\mathscr{N}$ by $\mathscr{N}_x := \{A \subseteq X : x \in \underline{A}\}$.

(III) $\to$ (I): Given $(X, \mathscr{N})$, define $\underline{A}$ by $\underline{A} := \{x \in X : x \in A \in \mathscr{N}_x\}$.

(I) $\to$ (II): Given $(X, \underline{\cdot})$, define R by xRy if and only if $y \in \bigcap\{A \subseteq X : x \in \underline{A}\}$,
i.e. $y \in \bigcap \mathscr{N}_x$.

(II) $\to$ (I): Given (X, R), define $\underline{\cdot}$ by $\underline{A} := \{x \in X : R_x \subseteq A\}$.

(III) $\to$ (II): Given $(X, \mathscr{N})$, define R by xRy if and only if $y \in A$ for some $A \in \mathscr{N}_x$.

(II) $\to$ (III): Given (X, R), define $\mathscr{N}$ by $\mathscr{N}_x := \{A \subseteq X : R_x \subseteq A\}$.

Exercise. Check that the respective required conditions are obtained.

For an elaborate discussion on these interrelations, interested readers may refer to ** [**?**]**.

7.7 Other work relating topology and rough sets

We briefly comment on other work done in connection to topological notions and rough sets. Biswas [10] introduced *rough metric spaces* and related notions such as rough diameter, and rough open balls and sets. *Nearness and proximity* structures in the context of rough set theory were studied by Peters et al. [76, 80, 81], and used extensively in application domains such as image analysis, forgery detection, analysis of microfossils. On the other hand, *uniform spaces* were related to approximation spaces by Vlach in [121].

Recently, Singh and Tiwari [thesis,TCS] have explored the notion of nearness both in the context of Pawlak's approximation space, and generalized approximation spaces considered by Yao. The notion of *rough proximity space* is defined based on Pawlak's approximation space, while a *Cech** rough proximity space* is defined on a generalised approximation space that is based on a tolerance relation. A *rough pseudo metric* is defined on an approximation space; it is shown that it induces a rough proximity on the approximation space. *Rough semi-uniformity* and *semi-linear uniformity* are also considered by the authors. Major goals are to obtain compactifications or completions of the introduced spaces that may help in applications in the area of image analysis.

Chapter 8
Inductive and Boolean Reasoning with Rough Sets

In this chapter we discuss the role of inductive reasoning in he rough set approach as well as we present applications of Boolean reasoning in the problems related to rough set applications in Machine Learning and Data Mining. This, in particular concerns data reduction by computing different kinds of reducts, discretisation or symbolic value groping, computing of different kinds of decision rules as well as association rules.

This chapter is based on several works, in particular [**?**, **?**, **?**, **?**, **?**, **?**, **?**, **?**, **?**, **?**, **?**, **?**].

8.1 Rough Sets and Induction

Granular formulas are constructed from atomic formulas corresponding to the considered attributes [**?**, **?**, **?**, **?**]. In the consequence, the satisfiability of such formulas is defined if the satisfiability of atomic formulas is given as the result of sensor measurement. Let us consider the two information systems $\mathbb{A} = (U, C, D)$ and $\mathbb{A}^* = (U^*, C)$ having the same set of attributes C, but $U \subseteq U^*$. Hence, one can consider for any constructed formula α over atomic formulas its semantics $\|\alpha\|_{\mathbb{A}} \subseteq U$ over U as well as the semantics $\|\alpha\|_{\mathbb{A}^*} \subseteq U^*$ over U^* (see Figure 8.1).

The difference between these two cases is the following. In the case of U, one can compute $\|\alpha\|_{\mathbb{A}} \subseteq U$ but in the case $\|\alpha\|_{\mathbb{A}^*} \subseteq U^*$, for any object from $U^* - U$, there is no information about its membership relative to $\|\alpha\|_{\mathbb{A}^*} - \|\alpha\|_{\mathbb{A}}$. One can estimate the satisfiability of α for objects $u \in U^* - U$ only after the relevant sensory measurements on u are performed. In particular, one can use some methods for estimation of relationships among semantics of formulas over U^* using the relationships among semantics of these formulas over U. For example, one can apply statistical methods. This step is crucial in investigation of extensions of approximation spaces relevant for inducing classifiers from data.

The rough set approach is strongly related to inductive reasoning (*e.g.*, in rough set based methods for inducing classifiers or clusters [**?**]). The general idea for in-

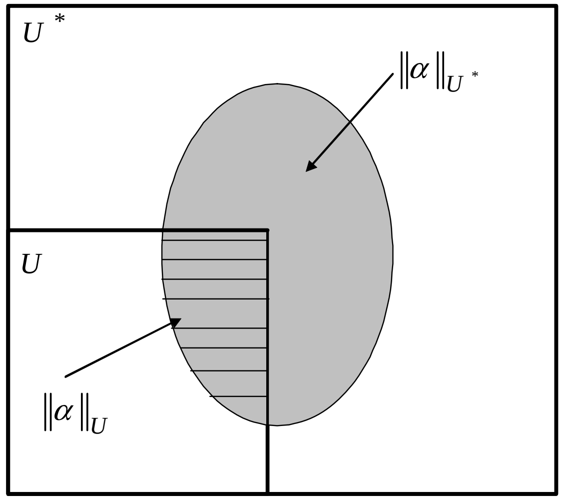


Fig. 8.1 Two semantics of α over U and U^*, respectively

ducing classifiers is as follows. From a given decision table a set of granules in the form of decision rules is induced together with arguments *for* and *against* for each decision rule and decision class. For any new object with known signature one can select rules matching this object. Note that the left hand sides of decision rules are described by formulas making it possible to check for new objects if they satisfy them assuming that the signatures of these objects are known. In this way one can consider two semantics of formulas: on a sample of objects U and on its extension $U^* \supseteq U$. Definitely, one should consider a risk related to such generalization in the decision rule inducing. Next, a conflict resolution should be applied for resolving conflicts between matched rules by new object voting for different decisions. In the rough set approach, the process of inducing classifiers can be considered as the process of inducing approximations of concepts over extensions of approximation spaces (defined over samples of objects represented by decision systems). The whole procedure can be generalized for the case of approximation of more complex information granules. It is worthwhile mentioning that there were also developed approaches for inducing approximate reasoning schemes.

A typical approach in machine learning is based on inducing classifiers from samples of objects. These classifiers are used for prediction decisions on objects unseen so far, if only the signatures of these objects are available. This approach can be called global, *i.e.*, leading to decision extension from a given sample of objects on the whole universe of objects. This global approach has some drawbacks (see Epilogue in [**?**]). Instead of this one can try to use transduction [**?**], semi-supervised learning, induced local models relative to new objects, or adaptive learning strategies. However, we are still far away from fully understanding discovery processes behind such generalization strategies [**?**].

8.1.1 Rough Sets and Classifiers

Rough sets are strongly related to inductive reasoning (*e.g.*, in rough set based methods for inducing classifiers or clusters).

In this section, we present an illustrative example of the rough set approach to induction of concept approximations. The approach can be generalized to the rough set approach to inductive extensions of approximation spaces.

Let us consider the problem of approximation of concepts over a universe U^{∞} (concepts that are subsets of U^{∞}). We assume that the concepts are perceived only through some subsets of U^{∞}, called samples. This is a typical situation in the machine learning, pattern recognition, or data mining approaches [**?**].

We assume that there is given an information system $\mathscr{A} = (U, A)$ and let us assume that for some $C \subseteq U^{\infty}$ there is given the set $\Pi_U(C) = C \cap U$. In this way we obtain a decision system $\mathbb{A}_d = (U, A, d)$, where $d(x) = 1$ if $x \in \Pi_U(C)$ and $d(x) = 0$, otherwise.

We would like to illustrate how from the decision function d may be induced a decision function μ_C defined over U^{∞} with values in the interval $[0, 1]$ which can be treated as an approximation of the characteristic function of C.

Let us assume that $RULES(\mathbb{A}_d)$ is a set of decision rules induced by some rule generation method from $\mathbb{A}_d$. For any object $x \in U^{\infty}$, let $MatchRules(\mathbb{A}_d, x)$ be the set of rules from this set supported by x.

Now, the rough membership function $\mu_C : U^{\infty} \to [0, 1]$ approximating the characteristic function of C can be defined as follows (see Figure 8.2)

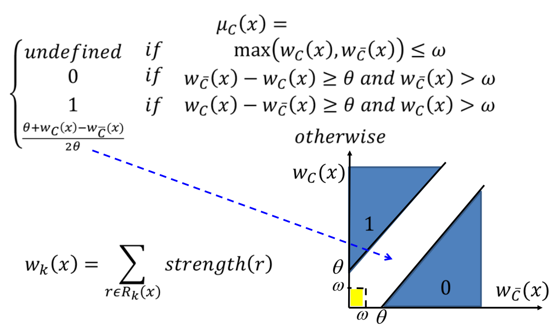


Fig. 8.2 Rough set based rule classifier for a concept C partially specified by a decision system, where Θ, ω are thresholds specified by the user, $strength(r)$ denotes the strength of the rule r (*e.g.*, defined by the support of the rule r, and $R_k(x)$ denotes the set of decision rules induced from a given decision system for the decision $k \in \{C, \overline{C}\}$ ($\overline{C} = U^* \setminus C$) matching the case x [**?**].

1. Let $R_k(x)$, for $x \in U^{\infty}$ be the set of all decision rules from $MatchRules(\mathbb{A}_d, x)$ with right hand side $d = k$, where $d = 1$ denotes that the rule r is voting for C and $d = 0$ – that the rule r is voting against C, respectively.

2. We define real values $w_k(x)$, where $w_1(x)$ is called the weight "for" and $w_0(x)$ the weight "against" membership of the object x in C, respectively, by $w_k(x) = \sum_{r \in R_k(x)} strength(r)$, where $strength(r)$ is a normalized function depending on $length$, $support$, $confidence$ of the decision rule r and on some global information about the decision system $\mathscr{A}_d$ such as the size of the decision system or the class distribution.
3. Finally, one can define the value of $\mu_C(x)$ in the following way: $\mu_C(x)$ is undefined if $\max(w_1(x), w_0(x)) < \omega$; $\mu_C(x) = 0$ if $w_0(x) - w_1(x) \geq \theta$ and $w_0(x) > \omega$; $\mu_C(x) = 1$ if $w_1(x) - w_0(x) \geq \theta$ and $w_1(x) > \omega$ and $\mu_C(x) = \frac{\theta + (w_1(x) - w_0(x))}{2\theta}$, otherwise, where ω, θ are parameters set by user.

For computing of the value $\mu_C(x)$ for $x \in U^\infty$ the user should select a strategy resolving conflicting votes "for" and "against" membership of x in C. The degree of these conflicts are represented by values $w_1(x)$ and $w_0(x)$, respectively. Note that for some cases of x due to the small differences between these values the selected strategy may not produce the definite answer and these cases will create the boundary region.

The induced membership function μ_C is illustrated in Figure 8.3. The trajectory

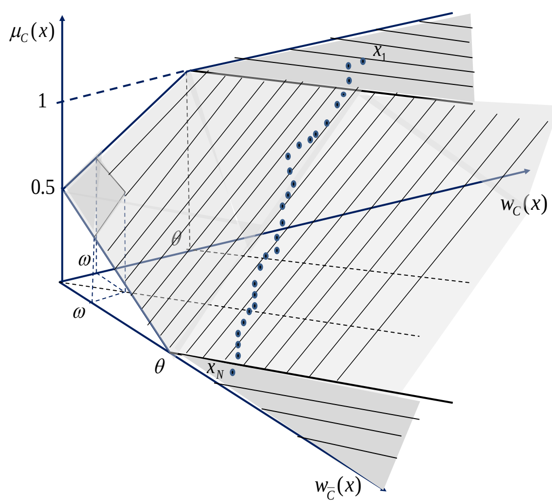


Fig. 8.3 Rough membership function μ_C. $x_1, \ldots, x_N$ is a trajectory on the surface of the membership function μ_C.

$x_1, \ldots, x_N$ in Figure 8.3 is showing changes of membership for a heap x_1 from which gradually are deleted grains of sand. The membership function μ_C is continuous and 'small' changes of the heap (caused by eliminating from the heap a single grain of sand) cause 'small' changes in the membership of function value what was used in [?] for explaining that the trajectory cannot 'jump over' the boundary region.

This leads to eliminating the contradiction related to the sorites paradox that there exists i such that x_i is a heap and x_{i+1} is not.

One can define the lower approximation, the upper approximation and the boundary region of the concept C relative to the induced rough membership function μ_C as follows

$$\begin{aligned} &\mathsf{LOW}_{\mu_C}(X) = \{x \in U^\infty :\ \mu_C(x) = 1\}, \\ &\mathsf{UPP}_{\mu_C}(X) = \{x \in U^\infty :\ \mu_C(x) > 0 \text{ or } \mu_C(x) \text{ is undefined}\}, \\ &\mathsf{BN}_{\mu_C}(X) = \mathsf{UPP}_{\mu_C}(X) \setminus \mathsf{LOW}_{\mu_C}(X). \end{aligned} \tag{8.1}$$

The whole procedure can be generalized for the case of approximation of more complex information granules than concepts.

8.1.2 Inducing Relevant Approximation Spaces

A key task in Granular Computing (GC) is the *information granulation* process that leads to the formation of information aggregates (with inherent patterns) from a set of available objects. A methodological and algorithmic issue is the formation of transparent (understandable) *information granules* inasmuch as they should provide a clear and understandable description of patterns present in sample objects [**?**, **?**]. Such a fundamental property can be formalized by a set of constraints that must be satisfied during the information granulation process. For example, in case of inducing granules such as classifiers, the constraints specify requirements for the quality of classifiers. Then, inducing of classifiers can be understood as searching for relevant approximation spaces (which can be treated as a spacial type of granules) relative to some properly selected optimization measures. Note that while there is a large literature on the covering based rough set approach (see, *e.g.*, [**?**, **?**]) still much more work should be done on (scalable) algorithmic searching methods for relevant approximation spaces in huge families of approximation spaces defined by many parameters determining neighborhoods, inclusion measures and approximation operators. The selection of the optimization measures is not an easy task because they should guarantee that the (semi-) optimal approximation spaces selected relative to these criteria should allow us to construct classifiers of the high quality.

Let us consider some examples of optimization measures [**?**]. For example, the quality of an approximation space can be measured by:

$$Quality_1 : \mathcal{SAS}(U) \times \mathcal{P}(U) \to [0,1], \tag{8.2}$$

where U is a non-empty set of objects and $\mathcal{SAS}(U)$ is a set of possible approximation spaces with the universe U.

Example 8.1. If $\mathsf{UPP}_{\mathbb{AS}}(X) \neq \varnothing$ for $\mathbb{AS} \in \mathcal{SAS}(U)$ and $X \subseteq U$ then

$$Quality_1(\mathbb{AS},X) = \nu_{SRI}(\mathsf{UPP}_{\mathbb{AS}}(X),\mathsf{LOW}_{\mathbb{AS}}(X)) = \frac{|\mathsf{LOW}_{\mathbb{AS}}(X)|}{|\mathsf{UPP}_{\mathbb{AS}}(X)|}. \quad (8.3)$$

The value $1 - Quality_1(\mathbb{AS},X)$ expresses the degree of completeness of our knowledge about X, given the approximation space $\mathbb{AS}$.

Example 8.2. In applications, we usually use another quality measure analogous to the minimum description length principle [**?**, **?**], where also the description length of approximation is included. Let us denote by $description(\mathbb{AS},X)$ the description length of approximation of X in $\mathbb{AS}$. The description length may be measured, *e.g.*, by the sum of description lengths of algorithms testing membership for neighborhoods used in construction of the lower approximation, the upper approximation, and the boundary region of the set X. Then the quality $Quality_2(\mathbb{AS},X)$ can be defined by

$$Quality_2(\mathbb{AS},X) = g(Quality_1(\mathbb{AS},X), description(\mathbb{AS},X)), \quad (8.4)$$

where g is a relevant function used for fusion of values $Quality_1(\mathbb{AS},X)$ and $description(\mathbb{AS},X)$. This function g can reflect weights given by experts relative to both criteria.

One can consider different optimization problems relative to a given class $\mathbb{S}\mathcal{AS}(U)$ of approximation spaces. For example, for a given $X \subseteq U$ and a threshold $t \in [0,1]$, one can search for an approximation space $\mathbb{AS}$ satisfying the constraint $Quality_2(\mathbb{AS},X) \geq t$.

The reader is referred to [**?**] for more details on searching for relevant approximation spaces.

8.2 Discernibility and Boolean Reasoning: Rough Set Methods for Machine Learning, Pattern Recognition, and Data Mining

Tasks collected under the labels of data mining, knowledge discovery, decision support, pattern classification, and approximate reasoning require tools aimed at discovering *templates (patterns)* in data and classifying them into certain *decision classes*. Templates are in many cases most frequent sequences of events, most probable events, regular configurations of objects, the decision rules of high quality, standard reasoning schemes. Tools for discovery and classification of templates are based on *reasoning schemes* rooted in various paradigms [**?**]. Such patterns can be extracted from data by means of methods based, *e.g.*, on Boolean reasoning and discernibility (see this section and [**?**]).

Discernibility relations belong to the most important relations considered in rough set theory. The ability to discern between perceived objects is important for constructing many entities like reducts, decision rules or decision algorithms. In the classical rough set approach, a discernibility relation $DIS(B) \subseteq U \times U$, where $B \subseteq A$ is a subset of attributes of an information system (U,A), is defined by $xDIS(B)y$ if and only if $non(x\mathcal{IND}_B y)$, where $\mathcal{IND}_B$ is the B-indiscernibility relation. However, this is, in general, not the case for the generalized approximation spaces. One can define indiscernibility by $x \in \mathcal{I}(y)$ and discernibility by $\mathcal{I}(x) \cap \mathcal{I}(y) = \varnothing$ for any objects x, y, where $\mathcal{I}(x) = B(x), \mathcal{I}(y) = B(y)$ in the case of the indiscernibility relation, and $\mathcal{I}(x), \mathcal{I}(y)$ are neighborhoods of objects not necessarily defined by the equivalence relation in a more general case.

The idea of Boolean reasoning, introduced by George Boole [**?**, **?**], is based on construction for a given problem P of a corresponding Boolean function f_P with the following property: The solutions for the problem P can be decoded from prime implicants of the Boolean function f_P (see Figure 8.4. Let us mention that to solve real-life problems it is necessary to deal with Boolean functions having large number of variables.

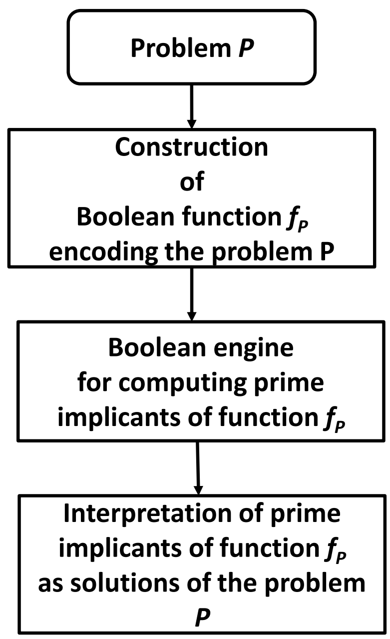


Fig. 8.4 Idea of Boolean reasoning

A successful methodology based on discernibility of objects and Boolean reasoning has been developed for computing of many entities important for applications, like reducts and their approximations, decision rules, association rules, discretization of real value attributes, symbolic value grouping, searching for new features defined by oblique hyperplanes or higher order surfaces, pattern extraction from data as well as conflict resolution or negotiation.

Most of the problems related to generation of the mentioned above entities are NP-complete or NP-hard. However, it was possible to develop efficient heuristics returning suboptimal solutions of the problems. The results of experiments on many data sets are very promising. They show very good quality of solutions generated by the heuristics in comparison with other methods reported in the literature (*e.g.*, with respect to the classification quality of unseen objects). Moreover, these heuristics are very efficient from the point of view of time necessary for computing of solutions. Many of these methods are based on discernibility matrices. Note that it is possible to compute the necessary information about these matrices using[1] information encoded in decision systems (*e.g.*, sorted in preprocessing [**?**, **?**, **?**]) directly, which significantly improves the efficiency of algorithms.

It is important to note that the methodology makes it possible to construct heuristics having a very important *approximation property* which can be formulated as follows: Expressions generated by heuristics, *i.e.*, implicants *close* to prime implicants define approximate solutions for the problem.

In this section, we discuss applications of methods based on rough sets and Boolean reasoning in machine learning, pattern recognition, and data mining.

In supervised machine learning paradigm [**?**, **?**, **?**, **?**], a learning algorithm is given a training data set, usually in the form of a decision system $\mathbb{A} = (U, A, d)^2$, prepared by an expert. Every such decision system classifies elements from U into decision classes. The purpose of the algorithm is to return a set of decision rules together with matching procedure and conflict resolution strategy, called a classifier, which makes it possible to classify unseen objects, *i.e.*, objects that are not described in the original decision table. In this section, we provide a number of rough set methods that can be used in construction of classifiers. For more information the reader is referred, *e.g.*, to [**?**, **?**]), and for papers on hierarchical learning and ontology approximation, *e.g.*, to [**?**, **?**, **?**, **?**, **?**, **?**, **?**, **?**, **?**, **?**, **?**].

Most of the techniques discussed below are based on computing prime implicants for computing different kinds of reducts. Unfortunately, they are computationally hard. However, many heuristics have been developed which turned out to be very promising. The results of experiments on many data sets, reported in the literature, show a very good quality of classification of unseen objects using these heuristics. A variety of methods for computing reducts and their applications can be found in [**?**, **?**, **?**, **?**, **?**, **?**, **?**, **?**, **?**, **?**, **?**, **?**, **?**, **?**]. The fact that the problem of finding a minimal reduct of a given information system is NP-hard was proved in [**?**].

As we mentioned, there exists a number of good heuristics that compute sufficiently many reducts in an acceptable time. Moreover, a successful methodology, based on different reducts, has been developed for solution of many problems like attribute selection, decision rule generation, association rule generation, discretization of real-valued attributes, and symbolic value grouping. For further readings the

[1] That is, without the necessity of generation and storing of the discernibility matrices

[2] For simplicity, we consider decision systems with one decision.

reader is referred to [**?**, **?**, **?**] (attribute selection); [**?**, **?**, **?**, **?**, **?**] (discretization); [**?**, **?**] (discretization of data stored in relational databases); and [**?**] (reduct approximation and association rules).

Many of these methods are based on discernibility matrices defined in this section. It is possible to compute the necessary information about these matrices using information or decision systems (*e.g.*, sorted in preprocessing [**?**, **?**]) directly what significantly improves the efficiency of algorithms.

The results presented in this section have been implemented, *e.g.*, in the RSES[3] software system (see also [**?**, **?**, **?**, **?**, **?**]). Sections 8.2.1-8.2.6 are based on a chapter of the book [**?**]. For links to other rough set software systems the reader is referred to the RSDS[4].

8.2.1 Reducts in Information and Decision Systems

A crucial concept in the rough set approach to machine learning is that of a reduct. In fact, the term "reduct" corresponds to a wide class of concepts. What typifies all of them is that they are used to reduce information (decision) systems by removing redundant attributes. In this section, we consider three kinds of reducts which will be used in the remainder of this chapter.

Given an information system $\mathbb{A} = (U, A)$, a *reduct* is a minimal set (wrt inclusion) of attributes $B \subseteq A$ such that $\mathcal{IND}_B = I(A)$, where $\mathcal{IND}_B, I(A)$ are the indiscernibility relations defined by B and A, respectively [**?**]. The intersection of all reducts is called a *core*.

Intuitively, a reduct is a minimal set of attributes from A that preserves the original classification defined by A. Reducts are extremely valuable in applications. Unfortunately, finding a minimal reduct is NP-hard in the general case. One can also show that, for any m, there is an information system with m attributes having an exponential (wrt m) number of reducts. Fortunately, there are reasonably good heuristics which allow one to compute sufficiently many reducts in an acceptable amount of time.

To provide a general method for computing reducts, we will use the following constructs.

Let $\mathbb{A} = (U, A)$ be an information system with n objects. The *discernibility matrix* of $\mathbb{A}$ is an $n \times n$ matrix with elements c_{ij} consisting of the set of attributes from A on which objects x_i and x_j differ, *i.e.*,

$$c_{ij} = \{a \in A \ : \ a(x_i) \neq a(x_j)\}, \text{ for } i, j = 1, ..., n. \tag{8.5}$$

A *discernibility function* $f_{\mathbb{A}}$ for $\mathbb{A}$ is a propositional formula of m Boolean variables, $a_1^*, ..., a_m^*$, corresponding to the attributes $a_1, ..., a_m$, defined by

[3] the Rough Set Exploration System: `https://www.mimuw.edu.pl/~szczuka/rses/start.html`

[4] the Rough Set Database System: `http://rsds.ur.edu.pl`

Table 8.1 The information table considered in Example 8.3

Object	*Speed*	*Color*	*Humidity*
car1	medium	green	high
car2	medium	yellow	low
car3	high	blue	high

Table 8.2 The discernibility matrix for the information table provided in Table 8.1

$\mathscr{M}(\mathbb{A})$	car1	car2	car3
car1		c,h	s,c
car2	c,h		s,c,h
car3	s,c	s,c,h	

$$f_{\mathbb{A}}(a_1^*,...,a_m^*) = \bigwedge_{1\leq j<i\leq m} \bigvee_{c\in c_{ij}^*, c_{ij}\neq\varnothing} c, \tag{8.6}$$

where $c_{ij}^* = \{a^* \ : \ a \in c_{ij}\}$. In the sequel, we write a_i instead of a_i^*, for simplicity. ▫

The discernibility function $f_{\mathbb{A}}$ describes constraints which must hold to preserve discernibility between all pairs of discernible objects from $\mathbb{A}$. It requires keeping at least one attribute from each non-empty element of the discernibility matrix corresponding to any pair of discernible objects.

It can be shown [**?**] that for any information system $\mathbb{A} = (U,A)$ the set of all prime implicants of $f_{\mathbb{A}}$ determines the set of all reducts of $\mathbb{A}$.

Example 8.3. Consider the information system $\mathbb{A}$ whose associated information table is provided in Table 8.1. The discernibility matrix for $\mathbb{A}$ is presented in Table 8.2. (The letters s, c and h stand for *Speed*, *Color* and *Humidity*, respectively.) The discernibility function for the information system $\mathbb{A}$ is then given by

$$f_{\mathbb{A}}(s,c,h) \equiv (c\vee h)\wedge(s\vee c)\wedge(s\vee c\vee h).$$

The prime implicants of $f_{\mathbb{A}}(s,c,h)$ can be computed in order to derive the reducts for $\mathbb{A}$:

$$\begin{aligned} f_{\mathbb{A}}(s,c,h) &\equiv (c\vee h)\wedge(s\vee c)\wedge(s\vee c\vee h) \\ &\equiv (c\vee h)\wedge(s\vee c) \\ &\equiv c\vee(h\wedge s). \end{aligned}$$

The prime implicants of $f_{\mathbb{A}}(s,c,h)$ are c and $h\wedge s$. Accordingly, there are two reducts of $\mathbb{A}$, namely $\{Color\}$ and $\{Humidity, Speed\}$. ▫

The second type of reduct used in this chapter are the *decision-relative reducts* for decision systems.

In terms of decision tables, $\partial_A(x)$, called the generalized decision function, is the mapping on U such that for any object x it specifies all rows in the table whose

Table 8.3 The decision table considered in Example 8.4

Object	*Speed*	*Color*	*Humidity*	*Danger*
car1	medium	green	high	no
car2	medium	yellow	small	no
car3	high	blue	high	yes

attribute values are the same as for x, and then collects the decision values from each row. A *decision-relative reduct* of $\mathbb{A} = (U,A,d)$ is a minimal (wrt inclusion) non-empty set of attributes $B \subseteq A$ such that $\partial_B = \partial_A$. Intuitively, the definition states that B allows us to classify exactly the same objects, as belonging to equivalence classes U/∂_A, as A. In terms of decision tables, the columns associated with the attributes $A - B$ may be removed without affecting the classification power of the original table.

To compute decision-relative reducts, we extend the definitions of discernibility matrix and discernibility function in the following straightforward manner. Let $\mathbb{A} = (U,A,d)$ be a consistent decision system (*i.e.*, $\partial_A(x)$ consists of exactly one decision for any $x \in U$) and let $\mathscr{M}(\mathbb{A}) = [c_{ij}]$ be the discernibility matrix of the information system (U,A). We construct a new matrix, $\mathscr{M}'(\mathbb{A}) = [c'_{ij}]$, where

$$c'_{ij} = \begin{cases} \varnothing, & \text{if and only if} \quad d(x_i) = d(x_j), \\ c_{ij}, & \text{otherwise.} \end{cases}$$

$\mathscr{M}'(\mathbb{A})$ is called the *decision-relative discernibility matrix* of $\mathbb{A}$. The *decision-relative discernibility function* $f^r_{\mathbb{A}}$ for $\mathbb{A}$ is constructed from the decision-relative discernibility matrix for $\mathbb{A}$ in the same way as a discernibility function is constructed from a discernibility matrix. Then it can be shown [**?**], that the set of all prime implicants of $f^r_{\mathbb{A}}$ determines the set of all decision-relative reducts of the consistent decision system $\mathbb{A}$.

Example 8.4. Consider the decision table associated with a decision system $\mathbb{A}$ as represented in Table 8.3.

The discernibility matrix for $\mathbb{A}$ is the same as the one given in Table 8.2, and the decision-relative discernibility matrix for $\mathbb{A}$ is provided in Table 8.4.

Using the decision-relative discernibility matrix, we can compute the decision-relative discernibility function for $\mathbb{A}$:

$$f^r_{\mathbb{A}}(s,c,h) \equiv (s \vee c) \wedge (s \vee c \vee h) \equiv (s \vee c).$$

The set of all prime implicants of $f^r_{\mathbb{A}}(s,c,h)$ is $\{s,c\}$. Therefore, there are two decision-relative reducts of $\mathbb{A}$, namely $\{Speed\}$ and $\{Color\}$.

To each decision-relative reduct B of a decision system $\mathbb{A}$, we assign a new decision system, called the *B-reduction* of $\mathbb{A}$. The details are as follows. Let

Table 8.4 The decision-relative discernibility matrix corresponding to the decision system shown in Table 8.3

$\mathscr{M}'(\mathbb{A})$	car1	car2	car3
car1			s,c
car2			s,c,h
car3	s,c	s,c,h	

Table 8.5 $\{Speed\}$-reduction of the decision system $\mathbb{A}$

Objects	*Speed*	*Danger*
car1, car2	medium	no
car3	high	yes

$\mathbb{A} = (U,A,d)$ be a consistent decision system and suppose that B is a decision-relative reduct of $\mathbb{A}$. A *B-reduction* of $\mathbb{A}$ is a decision system $\mathbb{A}^* = (V,B,d)$, where:[5]

- $V = \{[x]_B : x \in U\}$;
- $a([x]_B) = a(x)$, for each $a \in B$ and each $[x]_B \in V$;
- $d([x]_B) = d(x)$, for each $[x]_B \in V$.

Let $\mathbb{A}^*$ be the $\{Speed\}$-reduction of the decision system $\mathbb{A}$. The decision table associated with $\mathbb{A}^*$ is provided in Table 8.5. ▫

The above defined method for decision relative reducts computation can be easily extended to inconsistent decision systems.

Observe that another kind of reducts can be obtained by using the discernibility requirement relative to the positive regions, *i.e.*, $POS_A(d) = POS_B(d)$ instead of $\partial_B = \partial_A$. Certainly, for inconsistent decision systems the former requirement is less restrictive than the latter.

The last type of reduct, considered in this section, is used in applications where approximations to reducts are prefered to standard reducts. For example, approximate reducts for decision-relative reducts are making it possible to generate approximate decision rules. In the case of approximate reducts we relax the requirement for the discernibility preserving. Instead of preserving the discernibility for all entries of the discernibility matrix where it is necessary we preserve it to a degree, *i.e.*, in a number of entries characterized by a coefficient α. Such reducts are called α-reducts, where α is a real number from the interval $[0,1]$. More formal definition of approximate reducts is the following:

Let $\mathbb{A} = (U,A,d)$ be a decision system and let $\mathscr{M}(\mathbb{A})$ be the discernibility matrix of $\mathbb{A}$. Assume further that n is the number of non-empty sets in $\mathscr{M}(\mathbb{A})$. A set of attributes $B \subseteq A$ is called an α*-reduct* if and only if $\frac{m}{n} \geq \alpha$, where m is the number of sets that have a non-empty intersection with B.

[5] Recall that $[x]_B$, where $x \in U$, denotes the equivalence class of the relation $\mathcal{IND}_B$ which contains x.

The reader is referred to [**?**, **?**, **?**, **?**] for information on various types of approximate reducts. Additionally, [**?**, **?**, **?**, **?**] provide approximation criteria based on discernibility and, therefore, related to Boolean reasoning principles.

8.2.2 Attribute Selection

In the supervised machine learning approach, a learning algorithm is provided with training data. In the context of rough set machine learning techniques, training data is provided in the form of training decision systems, or their equivalent representations as decision tables.

Since the conditional attributes of a specific decision table are typically extracted from large sets of unstructured data, it is often the case that some of the attributes are irrelevant for the purpose of classification. Such attributes should be removed from the table if possible. The *attribute selection problem* is the problem of choosing a relevant subset of attributes, while removing the irrelevant ones.

A natural solution of the attribute selection problem is to assume that the intersection of the decision-relative reducts of a training decision table is the source of the relevant attributes. Unfortunately, there are two problems with this solution. Firstly, the intersection can be empty. Secondly, the number of attributes contained in all decision-relative reducts is usually small. Consequently, although these attributes perfectly characterize the training decision table, they are, in general, inadequate for providing a satisfactory classification of new objects not occurring in the training data.

To deal with the attribute selection problem, it is often reasonable to use various approximations of decision-relative reducts.

Let $\mathbb{A} = (U, A, d)$ be a consistent decision system. Any subset B of A is called an *approximate reduct* of $\mathbb{A}$. The number

$$\varepsilon_{A,\{d\}}(B) = \frac{\gamma(A,\{d\}) - \gamma(B,\{d\})}{\gamma(A,\{d\})} = 1 - \frac{\gamma(B,\{d\})}{\gamma(A,\{d\})}, \tag{8.7}$$

is called an *error of reduct approximation*.[6]

The error of reduct approximation expresses exactly how the set of attributes B approximates the set of condition attributes A with respect to determination of d. Note that $\varepsilon_{A,\{d\}}(B) \in [0,1]$, where 0 indicates no error, and the closer $\varepsilon_{A,\{d\}}(B)$ is to 1, the greater is the error. The reader is referred, *e.g.*, to [**?**, **?**] for more information on approximate reducts.

There are two general approaches to attribute selection: an open-loop approach and a closed-loop approach. Methods based on the open-loop approach are characterized by the fact that they do not use any feedback information about classifier

[6] Recall that the coefficient $\gamma(X,Y)$ expresses the degree of dependency between sets of attributes X and Y.

quality for attribute selection. In contrast, the methods based on the closed-loop approach use feedback information as criteria for attribute selection.

A number of attribute selection algorithms have been proposed in the machine learning literature, but they will not be considered here since our focus is on rough set based techniques. Rough set techniques which attempt to solve the attribute selection problem are typically based on the closed-loop approach and consist of the following basic steps:[7]

1. Decision-relative reducts are extracted from a training decision table. The attributes contained in these reducts (or in their intersection) are viewed as potentially relevant.
2. Using the specific machine learning algorithm, a classifier based on the chosen attributes is constructed.
3. The classifier is then tested on a new set of training data; if its performance is unsatisfactory (wrt some measure), a new set of attributes is constructed by extracting approximate additional reducts for the initial training table, and the process is repeated.

Reducts need not be the only source of information used in the selection of attributes. The rough set approach offers another interesting possibility. The main idea is to generalize the notion of attribute reduction by introducing the concept of *significance of attributes*. This measure enables attributes to be evaluated using a multi-valued scale which assigns a real number from the interval [0,1] to an attribute. This number, expressing the importance of an attribute in a decision system, is evaluated by measuring the effect of removing the attribute from the table.

The *significance of an attribute a* in a decision table $\mathbb{A} = (U, A, d)$ is defined by

$$\sigma_{A,\{d\}}(a) = \frac{\gamma(A,\{d\}) - \gamma(A-\{a\},\{d\})}{\gamma(A,\{d\})} = 1 - \frac{\gamma(A-\{a\},\{d\})}{\gamma(A,\{d\})}. \tag{8.8}$$

Assume that $B \subseteq A$. The significance coefficient can be extended to sets of attributes as follows,

$$\sigma_{(A,\{d\})}(B) = \frac{\gamma(A,\{d\}) - \gamma(A-B,\{d\})}{\gamma(A,\{d\})} = 1 - \frac{\gamma(A-B,\{d\})}{\gamma(A,\{d\})}. \tag{8.9}$$

The coefficient $\sigma_{A,\{d\}}(B)$, can be understood as a classification error which occurs when the attributes $a \in B$ are removed from the decision system. Note that $\sigma_{A,\{d\}}(B) \in [0,1]$, where 0 indicates that removal of attributes in B causes no error, and the closer $\sigma_{A,\{d\}}(B)$ is to 1, the greater the error is.

In this section, we have mainly concentrated on the case, where the attributes are selected from the set of attributes of the input decision system. In some cases it might be useful to replace some attributes by a new one.

---

[7] There are public domain software packages, for instance the RSES system (for references see, *e.g.*, [**?**] and `http://logic.mimuw.edu.pl/$\sim$rses/`), which offer software that may be used to solve the attribute selection problem.

For example, if one considers a concept of a safe distance between vehicles, then attributes, say VS standing for "vehicle speed" and SL standing for "speed limit", can be replaced by an attribute DIF representing the difference $SL - VS$. In fact, the new attribute better corresponds to the concept of safe distance than the pair (VS, SL).

For more readings on the rough set based methods for features selection the reader is referred to surveys [**?**, **?**, **?**] and articles [**?**, **?**, **?**, **?**, **?**, **?**, **?**, **?**, **?**, **?**, **?**, **?**, **?**, **?**, **?**].

8.2.3 Value Set Reduction

Consider a decision system with a large number of attribute values. There is a very low probability that a new object will be properly recognized by matching its attribute value vector with any of the rows in the decision table associated with the decision system. So, in order to construct a high quality classifier, it is often necessary to reduce the cardinality of the value sets of specific attributes in a training decision table. The task of reducing the cardinality of value sets is referred to as the *value set reduction problem*.

In this section, two methods of value set reduction are considered:

1. discretization, used for real value attributes, and
2. symbolic attribute value grouping, used for symbolic attributes.

8.2.3.1 Discretization

A discretization replaces value sets of conditional real-valued attributes with intervals. The replacement ensures that a consistent decision system is obtained (assuming a given consistent decision system) by substituting original values of objects in the decision table by the unique names of the intervals comprising these values. This substantially reduces the size of the value sets of real-valued attributes.

The use of discretization is not specific to the rough set approach to machine learning. In fact, a majority of rule or tree induction algorithms require it for a good performance.

Let $\mathbb{A} = (U, A, d)$ be a consistent decision system. Assume $V_a = [l_a, r_a) \subseteq \mathbb{R}$,[8] for any $a \in A$, and $l_a < r_a$. A pair (a, c), where $a \in A$ and $c \in V_a$, is called a *cut* on V_a.

Any attribute $a \in A$ defines a sequence of real numbers $v_1^a < v_2^a < \cdots < v_{k_a}^a$, where $\{v_1^a, v_2^a, \ldots, v_{k_a}^a\} = \{a(x) : x \in U\}$. The *set of basic cuts* on a, written B_a, is specified by

$$B_a = \{(a, (v_1^a + v_2^a)/2), (a, (v_2^a + v_3^a)/2), \ldots, (a, (v_{k_a-1}^a + v_{k_a}^a)/2)\}.$$

The set $\bigcup_{a \in A} B_a$ is called the *set of basic cuts* on $\mathbb{A}$. □

---

[8] $\mathbb{R}$ denotes the set of real numbers.

Table 8.6 The discretization process: (a) The original decision system $\mathbb{A}$ considered in Example 8.5 considered in Example 8.6

$\mathbb{A}$	a	b	d
u_1	0.8	2.0	1
u_2	1.0	0.5	0
u_3	1.3	3.0	0
u_4	1.4	1.0	1
u_5	1.4	2.0	0
u_6	1.6	3.0	1
u_7	1.3	1.0	1

Table 8.7 The discretization process: (b) The C-discretization of $\mathbb{A}$ considered in Example 8.6

$\mathbb{A}^C$	a^C	b^C	d
u_1	0	2	1
u_2	1	0	0
u_3	2	3	0
u_4	3	1	1
u_5	3	2	0
u_6	4	3	1
u_7	2	1	1

Example 8.5. Consider a consistent decision system $\mathbb{A}$ and the associated decision table presented in Table 8.6.

We assume that the initial value domains for the attributes a and b are

$$V_a = [0,2)\,; V_b = [0,4)\,.$$

The sets of values of a and b for objects from U are

$$a(U) = \{0.8,\, 1.0,\, 1.3,\, 1.4,\, 1.6\}\,;$$
$$b(U) = \{0.5,\, 1.0,\, 2.0,\, 3.0\}\,.$$

By definition, the sets of basic cuts for a and b are

$$B_a = \{(a,0.9),\ (a,1.15),\ (a,1.35),\ (a,1.5)\};$$
$$B_b = \{(b,0.75);\ (b,1.5);\ (b,2.5)\}.$$

□

Using the idea of cuts, decision systems with real-valued attributes can be discretized. For a decision system $\mathbb{A} = (U,A,d)$ and $a \in A$, let

$$C_a = \{(a,c_1^a),(a,c_2^a),\ldots,(a,c_k^a)\},$$

be any set of cuts of a. Assume that $c_1^a < c_2^a < \cdots < c_k^a$. The set of cuts $C = \bigcup_{a\in A} C_a$ defines a new decision system $\mathbb{A}^{\mathrm{C}} = (U,A^{\mathrm{C}},d)$, called the *$C$-discretization of* $\mathbb{A}$, where

- $A^{\mathrm{C}} = \{a^{\mathrm{C}} : a \in A\}$;
- $a^C(x) = \begin{cases} 0, & \text{if and only if } a(x) < c_1^a, \\ i, & \text{if and only if } a(x) \in [c_i^a, c_{i+1}^a), \text{ for } i \in \{1, \ldots, k-1\}, \\ k+1, & \text{if and only if } a(x) > c_k^a. \end{cases}$

Example 8.6 (Example 8.5 continued). Let $C = B_a \cup B_b$. It is easy to check that the C-discretization of $\mathbb{A}$ is the decision system whose decision table is provided in Table 8.7. ▫

Since a decision system can be discretized in many ways, a natural question arises how to evaluate various possible discretizations.

A set of cuts C is called $\mathbb{A}$-*consistent*, if $\partial_{\mathbb{A}} = \partial_{\mathbb{A}^{\mathrm{C}}}$, where $\partial_{\mathbb{A}}$ and $\partial_{\mathbb{A}^{\mathrm{C}}}$ are generalized decision functions for $\mathbb{A}$ and $\mathbb{A}^{\mathrm{C}}$, respectively. An $\mathbb{A}$-consistent set of cuts C is $\mathbb{A}$-*irreducible* if C' is not $\mathbb{A}$-consistent for any $C' \subset C$. The $\mathbb{A}$-consistent set of cuts C is $\mathbb{A}$-*optimal* if $card(C) \leq card(C')$, for any $\mathbb{A}$-consistent set of cuts C'.

As easily observed, the set of cuts considered in Example 8.6 is $\mathbb{A}$-consistent. However, as we shall see in Example 8.7, it is neither optimal nor irreducible.

Since the purpose of the discretization process is to reduce the size of individual value sets of attributes, we are primarily interested in optimal sets of cuts. These are extracted from the basic sets of cuts for a given decision system.

Let $\mathbb{A} = (U, A, d)$ be a consistent decision system where $U = \{u_1, \ldots, u_n\}$. Recall that any attribute $a \in A$ defines a sequence $v_1^a < v_2^a < \cdots < v_{k_a}^a$, where $\{v_1^a, v_2^a, \ldots, v_{k_a}^a\} = \{a(x) : x \in U\}$. Let $ID(\mathbb{A})$ be the set of pairs (i, j) such that $i < j$ and $d(u_i) \neq d(u_j)$. We now construct a propositional formula, called the *discernibility formula* of $\mathbb{A}$, as follows:

1. To each interval of the form $\left[v_k^a, v_{k+1}^a\right)$, $a \in A$ and $k \in \{1, \ldots, n_a - 1\}$, we assign a Boolean variable denoted by p_k^a. The set of all these variables is denoted by $V(\mathbb{A})$.
2. We first construct a family of formulas
$$\{B(a, i, j) : a \in A \text{ and } (i, j) \in ID(\mathbb{A})\},$$
where $B(a, i, j)$ is a disjunction of all elements from the set
$$\left\{ p_k^a : \left[v_k^a, v_{k+1}^a\right) \subseteq [\min\{a(u_i), a(u_j)\}, \max\{a(u_i), a(u_j)\}) \right\}.$$
3. Next, we construct a family of formulas
$$\{C(i, j) : i, j \in \{1, \ldots, n\}, i < j \text{ and } (i, j) \in ID(\mathbb{A})\},$$
where $C(i, j) = \bigvee_{a \in A} B(a, i, j)$.
4. Finally, the discernibility formula for $\mathbb{A}$, $D(\mathbb{A})$, is defined as
$$D(\mathbb{A}) = \bigwedge C(i, j),$$
where $i < j$ and $(i, j) \in ID(\mathbb{A})$ and $C(i, j) \not\equiv \text{FALSE}$.

Any non empty set $S = \{p_{k_1}^{a_1}, \ldots, p_{k_r}^{a_r}\}$ of Boolean variables from $V(\mathbb{A})$ uniquely defines a set of cuts, $C(S)$, given by

$$C(S) = \{(a_1, (v_{k_1}^{a_1} + v_{k_1+1}^{a_1})/2), \cdots, (a_r, (v_{k_r}^{a_r} + v_{k_r+1}^{a_r})/2)\}.$$

Then we have the following properties:

Let $\mathbb{A} = (U, A, d)$ be a consistent decision system. For any non-empty set $S \subseteq V(\mathbb{A})$ of Boolean variables, the following two conditions are equivalent:

1. The conjunction of variables from S is a prime implicant of the discernibility formula for $\mathbb{A}$.
2. $C(S)$ is an $\mathbb{A}$-irreducible set of cuts on $\mathbb{A}$. ▫

Let $\mathbb{A} = (U, A, d)$ be a consistent decision system. For any non-empty set $S \subseteq V(\mathbb{A})$ of Boolean variables, the following two conditions are equivalent:

1. The conjunction of variables from S is a minimal (wrt to length) prime implicant of the discernibility formula for $\mathbb{A}$.
2. $C(S)$ is an $\mathbb{A}$-optimal set of cuts on $\mathbb{A}$.

Example 8.7 (Example 8.6 continued).

$$\begin{aligned} ID(\mathbb{A}) = \{&(1,2),\ (1,3),\ (1,5),\ (2,4),\ (2,6),\ (2,7) \\ &(3,4),\ (3,6),\ (3,7),\ (4,5),\ (5,6),\ (5,7)\}. \end{aligned}$$

1. We introduce four Boolean variables, p_1^a, p_2^a, p_3^a, p_4^a, corresponding respectively to the intervals

   $$[0.8, 1.0), [1.0, 1.3), [1.3, 1.4), [1.4, 1.6)$$

   of the attribute a, and three Boolean variables, p_1^b, p_2^b, p_3^b, corresponding respectively to the intervals

   $$[0.5, 1.0), [1.0, 2.0), [2, 3.0)$$

   of the attribute b
2. The following are the formulas $B(a,i,j)$ and $B(b,i,j)$, where $i < j$ and $(i,j) \in ID(\mathbb{A})$:

$$\begin{array}{ll}
B(a,1,2) \equiv p_1^a & B(b,1,2) \equiv p_1^b \vee p_2^b \\
B(a,1,3) \equiv p_1^a \vee p_2^a & B(b,1,3) \equiv p_3^b \\
B(a,1,5) \equiv p_1^a \vee p_2^a \vee p_3^a & B(b,1,5) \equiv \textsc{False} \\
B(a,2,4) \equiv p_2^a \vee p_3^a & B(b,2,4) \equiv p_1^b \\
B(a,2,6) \equiv p_2^a \vee p_3^a \vee p_4^a & B(b,2,6) \equiv p_1^b \vee p_2^b \vee p_3^b \\
B(a,2,7) \equiv p_2^a & B(b,2,7) \equiv p_1^b \\
B(a,3,4) \equiv p_3^a & B(b,3,4) \equiv p_2^b \vee p_3^b \\
B(a,3,6) \equiv p_3^a \vee p_4^a & B(b,3,6) \equiv \textsc{False}
\end{array}$$

$$\begin{array}{ll} B(a,3,7) \equiv \text{FALSE} & B(b,3,7) \equiv p_2^b \vee p_3^b \\ B(a,4,5) \equiv \text{FALSE} & B(b,4,5) \equiv p_2^b \\ B(a,5,6) \equiv p_4^a & B(b,5,6) \equiv p_3^b \\ B(a,5,7) \equiv p_3^a & B(b,5,7) \equiv p_2^b. \end{array}$$

3. The following are the formulas $C(i,j)$, where $i < j$ and $(i,j) \in ID(\mathbb{A})$:

$$\begin{array}{ll} C(1,2) \equiv p_1^a \vee p_1^b \vee p_2^b & C(1,3) \equiv p_1^a \vee p_2^a \vee p_3^b \\ C(1,5) \equiv p_1^a \vee p_2^a \vee p_3^a & C(2,4) \equiv p_2^a \vee p_3^a \vee p_1^b \\ C(2,6) \equiv p_2^a \vee p_3^a \vee p_4^a \vee p_1^b \vee p_2^b \vee p_3^b & C(2,7) \equiv p_2^a \vee p_1^b \\ C(3,4) \equiv p_3^a \vee p_2^b \vee p_3^b & C(3,6) \equiv p_3^a \vee p_4^a \\ C(3,7) \equiv p_2^b \vee p_3^b & C(4,5) \equiv p_2^b \\ C(5,6) \equiv p_4^a \vee p_3^b & C(5,7) \equiv p_3^a \vee p_2^b. \end{array}$$

4. The discernibility formula for $\mathbb{A}$ is then given by

$$\begin{aligned} D(\mathbb{A}) \equiv\ & (p_1^a \vee p_1^b \vee p_2^b) \wedge (p_1^a \vee p_2^a \vee p_3^b) \wedge \\ & (p_1^a \vee p_2^a \vee p_3^a) \wedge (p_2^a \vee p_3^a \vee p_1^b) \wedge \\ & (p_2^a \vee p_3^a \vee p_4^a \vee p_1^b \vee p_2^b \vee p_3^b) \wedge (p_2^a \vee p_1^b) \wedge \\ & (p_3^a \vee p_2^b \vee p_3^b) \wedge (p_3^a \vee p_4^a) \wedge (p_2^b \vee p_3^b) \wedge \\ & p_2^b \wedge (p_4^a \vee p_3^b) \wedge (p_3^a \vee p_2^b). \end{aligned}$$

The prime implicants of the formula $D(\mathbb{A})$ are

$$\begin{array}{l} p_2^a \wedge p_4^a \wedge p_2^b \\ p_2^a \wedge p_3^a \wedge p_2^b \wedge p_3^b \\ p_3^a \wedge p_1^b \wedge p_2^b \wedge p_3^b \\ p_1^a \wedge p_4^a \wedge p_1^b \wedge p_2^b. \end{array}$$

Suppose we take the prime implicant $p_1^a \wedge p_4^a \wedge p_1^b \wedge p_2^b$. Its corresponding set of cuts is

$$C = \{(a,0.9),(a,1.5),(b,0.75),(b,1.5)\}.$$

The decision table for the C-discretization of $\mathbb{A}$ is provided in Table 8.8.

Observe that the set of cuts corresponding to the prime implicant $p_2^a \wedge p_4^a \wedge p_2^b$ is $\{(a,1.15),(a,1.5),(b,1.5)\}$. Thus C is not an optimal set of cuts. ▫

The problem of searching for an optimal set of cuts P in a given decision system $\mathbb{A}$ is NP-hard. However, it is possible to devise efficient heuristics which, in general, return reasonable sets of cuts. One of them, called MD-heuristics, is presented below.

We say that a cut (a,c) discerns objects x and y if and only if $a(x) < c \leq a(y)$ or $a(y) < c \leq a(x)$.

Let n be the number of objects and let k be the number of attributes of a decision system $\mathbb{A}$. It can be shown that the best cut can be found in $O(kn)$ steps using $O(kn)$ space only.

Table 8.8 The C-discretization considered in Example 8.7

$\mathbb{A}^C$	a^C	b^C	d
u_1	0	2	1
u_2	1	0	0
u_3	1	2	0
u_4	1	1	1
u_5	1	2	0
u_6	2	2	1
u_7	1	1	1

Algorithm 1: MD-heuristics

INPUT: a decision system $\mathbb{A} = (U, A, d)$
OUTPUT: a set of cuts $\mathscr{C}$

1. Set $\mathscr{C}$ to $\varnothing$.
2. Let $\bigcup_{a \in A} C_a$ be the set of basic cuts on $\mathbb{A}$.
3. Construct an information table $\mathbb{A}^* = (U^*, A^*)$ such that
   - U^* is the set of pairs (u_i, u_j) of objects discerned by d (in $\mathbb{A}$) such that $i < j$;
   - $A^* = \bigcup_{a \in A} C_a$, where for each $c \in A^*$,

$$c(x,y) = \begin{cases} 1, & \text{if and only if } c \text{ discerns } x \text{ and } y \text{ (in } \mathbb{A}\text{)}, \\ 0, & \text{otherwise.} \end{cases}$$

4. Choose a column from $\mathbb{A}^*$ with the maximal number of occurrences of 1's; add the cut corresponding to this column to $\mathscr{C}$; delete the column from $\mathbb{A}^*$, together with all rows marked with 1 in it.
5. If A^* is non-empty, then go to step 4 else stop. □

Example 8.8. Consider the decision table with the associated decision system $\mathbb{A}$, provided in Table 8.6 from Example 8.5. The associated information table for the information system $\mathbb{A}^*$ is presented in Table 8.9.

Under the assumption that columns with maximal number of 1's are chosen from left to right (if many such columns exist in a given step), the set of cuts returned by the algorithm is $\{(a, 1.35), (b, 1.5), (a, 1.15), (a, 1.5)\}$. However, as shown in Example 8.7, it is not an optimal set of cuts. □

8.2.3.2 Symbolic Attribute Value Grouping

Symbolic attribute value grouping is a technique for reducing the cardinality of value sets of symbolic attributes. Let $\mathbb{A} = (U, A, d)$ be a decision system. Any function $c_a : V_a \to \{1, \ldots, m\}$, where $m \leq card(V_a)$, is called a *clustering function* for V_a. The *rank of* c_a, denoted by *rank* (c_a), is the value $card(\{c_a(x) \mid x \in V_a\})$.

For $B \subseteq A$, a family of clustering functions $\{c_a\}_{a \in B}$ is *B-consistent* if and only if

Table 8.9 The information table for the information system $\mathbb{A}^*$

$\mathbb{A}^*$	$(a,0.9)$	$(a,1.15)$	$(a,1.35)$	$(a,1.5)$	$(b,0.75)$	$(b,1.5)$	$(b,2.5)$
(u_1,u_2)	1	0	0	0	1	1	0
(u_1,u_3)	1	1	0	0	0	0	1
(u_1,u_5)	1	1	1	0	0	0	0
(u_2,u_4)	0	1	1	0	1	0	0
(u_2,u_6)	0	1	1	1	1	1	1
(u_2,u_7)	0	1	0	0	1	0	0
(u_3,u_4)	0	0	1	0	0	1	1
(u_3,u_6)	0	0	1	1	0	0	0
(u_3,u_7)	0	0	0	0	0	1	1
(u_4,u_5)	0	0	0	0	0	1	0
(u_5,u_6)	0	0	0	1	0	0	1
(u_5,u_7)	0	0	1	0	0	1	0

$$\forall a \in B \ [c_a(a(u)) = c_a(a(u'))],$$

implies

$$(u,u') \in \mathcal{IND}_B \cup I(\{d\}), \text{ for any pair } (u,u') \in U.$$

The notion of B-consistency has the following intuitive interpretation: If two objects are indiscernible wrt clustering functions for value sets of attributes from B, then they are indiscernible either by the attributes from B or by the decision attribute.

We consider the following problem, called the *symbolic value partition grouping problem*:

> Given a decision system $\mathbb{A} = (U,A,d)$, where $U = \{u_1,\ldots,u_k\}$, and a set of attributes $B \subseteq A$, search for a B-consistent family $\{c_a\}_{a\in B}$ of clustering functions such that $\sum_{a\in B} rank(c_a)$ is minimal.

In order to solve this problem, we apply the following steps:

1. Introduce a set of new Boolean variables:[9]

$$\{a_v^{v'} : a \in B \text{ and } v,v' \in V_a \text{ and } v \neq v'\}.$$

   We extract a subset S of this set such that $a_v^{v'} \in S$ implies that $v' < v$ wrt some arbitrary linear order $<$ on the considered domain.
2. Construct matrix $\mathscr{M} = [c_{ij}]_{i,j=1,\ldots,k}$ as follows:

$$c_{ij} = \{a_v^{v'} \in S : v' = a(u_i) \text{ and } v = a(u_j) \text{ and } d(u_i) \neq d(u_j)\}.$$

   It is easily seen that in the case of a binary decision, the matrix can be reduced by placing objects corresponding to the first decision in rows and those corre-

[9] The introduced variables serve to discern between pairs of objects wrt an attribute a.

sponding to the second decision in columns. We call such a matrix a *reduced discernibility matrix.*

3. Using the reduced matrix, $\mathscr{M}'$, obtained in the previous step, construct the function

$$\bigwedge_{c_{ij}\in\mathscr{M}'}\left(\bigvee_{c\in c_{ij},c_{ij}\neq\varnothing} c\right).$$

4. Compute the shortest prime implicant I of the constructed function.
5. Using I, construct, for each attribute $a \in B$, an undirected graph $\Gamma_a = \langle V_a^\Gamma, E_a^\Gamma \rangle$, where
   - $V_a^\Gamma = \{a_v \mid v \in V_a\}$;
   - $E_a^\Gamma = \{(a_x, a_y) \mid x, y \in U \text{ and } a(x) \neq a(y)\}$.
     Note that using I one can construct E_a^Γ due to the equality

$$E_a^\Gamma = \{(a_v, a_{v'}) : a_v^{v'} \text{ occurs in } I\}.$$

6. Find a minimal coloring of vertices for Γ_a.[10] The coloring defines a partition of V_a^Γ by assuming that all vertices of the same color belong to the same partition set and no partition set contains vertices with different colors. Partition sets are named using successive natural numbers.
   The clustering function for V_a^Γ is $c_a(a_v) = i$, provided that a_v is a member of the i-th partition set.

Remark 8.1. In practical implementations, one does not usually construct the matrix $\mathscr{M}$ explicitly, as required in Steps (2)-(3) above. Instead, prime implicants are directly extracted from the original decision system.

It should be emphasized that in Step (4) above, there can be many different shortest prime implicants and in Step (6) there can be many different colorings of the obtained graphs. Accordingly, one can obtain many substantially different families of clustering functions resulting in different classifiers. In practice, one often generates a number of families of clustering functions, tests them against data and chooses the best one.

Using the construction above to generate a family of partitions, it is usually possible to obtain a substantially smaller decision table, according to the following definition.

Let $\mathbb{A} = (U, A, d)$ be a decision system and $B \subseteq A$. Any family of clustering functions $c = \{c_a\}_{a\in B}$ specifies a new decision system $\mathbb{A}^c = (U, A^c, d)$ called the *c-reduction of* $\mathbb{A}$ wrt B, where $A^c = \{a^{\mathbf{c}} : a \in B\}$ and $a^c(x) = c_a(a(x))$.

Example 8.9. Consider the decision table provided in Table 8.10. The goal is to solve the symbolic value partition problem for $B = A$.

One then has to perform the following steps:

[10] The colorability problem is solvable in polynomial time for $k = 2$, but remains NP-complete for all $k \geq 3$. But, similarly to discretization, one can apply some efficient search heuristics for generating (sub-) optimal partitions.

Table 8.10 The decision table considered in Example 8.9

$\mathscr{A}$	a	b	d
u_1	a_1	b_1	0
u_2	a_1	b_2	0
u_3	a_2	b_3	0
u_4	a_3	b_1	0
u_5	a_1	b_4	1
u_6	a_2	b_2	1
u_7	a_2	b_1	1
u_8	a_4	b_2	1
u_9	a_3	b_4	1
u_{10}	a_2	b_5	1

Table 8.11 The reduced matrix corresponding to the decision table provided in Table 8.10

$\mathscr{M}'$	u_1	u_2	u_3	u_4
u_5	$b_{b_4}^{b_1}$	$b_{b_4}^{b_2}$	$a_{a_2}^{a_1}, b_{b_4}^{b_3}$	$a_{a_3}^{a_1}, b_{b_4}^{b_1}$
u_6	$a_{a_2}^{a_1}, b_{b_2}^{b_1}$	$a_{a_2}^{a_1}$	$b_{b_3}^{b_2}$	$a_{a_3}^{a_2}, b_{b_2}^{b_1}$
u_7	$a_{a_2}^{a_1}$	$a_{a_2}^{a_1}, b_{b_2}^{b_1}$	$b_{b_3}^{b_1}$	$a_{a_3}^{a_2}$
u_8	$a_{a_4}^{a_1}, b_{b_2}^{b_1}$	$a_{a_4}^{a_1}$	$a_{a_4}^{a_2}, b_{b_3}^{b_2}$	$a_{a_4}^{a_3}, b_{b_2}^{b_1}$
u_9	$a_{a_3}^{a_1}, b_{b_4}^{b_1}$	$a_{a_3}^{a_1}, b_{b_4}^{b_2}$	$a_{a_3}^{a_2}, b_{b_4}^{b_3}$	$b_{b_4}^{b_1}$
u_{10}	$a_{a_2}^{a_1}, b_{b_5}^{b_1}$	$a_{a_2}^{a_1}, b_{b_5}^{b_2}$	$b_{b_5}^{b_3}$	$a_{a_3}^{a_2}, b_{b_5}^{b_1}$

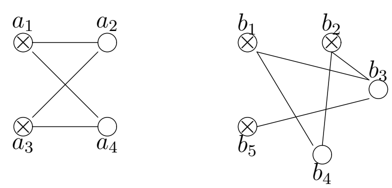


Fig. 8.5 Coloring of attribute value graphs constructed in Example 8.9

Table 8.12 The reduced table corresponding to graphs shown in Figure 8.5

a^c	b^c	d
1	1	0
2	2	0
1	2	1
2	1	1

1. Introduce new Boolean variables a_v^u, b_x^w, for all $u, v \in V_a, u < v$ and $w, x \in V_b, w < x$.
2. The reduced matrix $\mathscr{M}'$ is presented in Table 8.11.
3. The required Boolean function is given by

$$b^{b_1}_{b_4} \wedge b^{b_2}_{b_4} \wedge (a^{a_1}_{a_2} \vee b^{b_3}_{b_4}) \wedge (a^{a_1}_{a_3} \vee b^{b_1}_{b_4}) \wedge$$
$$(a^{a_1}_{a_2} \vee b^{b_1}_{b_2}) \wedge a^{a_1}_{a_2} \wedge b^{b_2}_{b_3} \wedge (a^{a_2}_{a_3} \vee b^{b_1}_{b_2}) \wedge$$
$$a^{a_1}_{a_2} \wedge (a^{a_1}_{a_2} \vee b^{b_1}_{b_2}) \wedge b^{b_1}_{b_3} \wedge a^{a_2}_{a_3} \wedge$$
$$(a^{a_1}_{a_4} \vee b^{b_1}_{b_2}) \wedge a^{a_1}_{a_4} \wedge (a^{a_2}_{a_4} \vee b^{b_2}_{b_3}) \wedge (a^{a_3}_{a_4} \vee b^{b_1}_{b_2}) \wedge$$
$$(a^{a_1}_{a_3} \vee b^{b_1}_{b_4}) \wedge (a^{a_1}_{a_3} \vee b^{b_2}_{b_4}) \wedge (a^{a_2}_{a_3} \vee b^{b_3}_{b_4}) \wedge b^{b_1}_{b_4} \wedge$$
$$(a^{a_1}_{a_2} \vee b^{b_1}_{b_5}) \wedge (a^{a_1}_{a_2} \vee b^{b_2}_{b_5}) \wedge b^{b_3}_{b_5} \wedge (a^{a_2}_{a_3} \vee b^{b_1}_{b_5}).$$

4. The shortest prime implicant for the function is

$$I \equiv a^{a_1}_{a_2} \wedge a^{a_2}_{a_3} \wedge a^{a_1}_{a_4} \wedge a^{a_3}_{a_4} \wedge b^{b_1}_{b_4} \wedge b^{b_2}_{b_4} \wedge b^{b_2}_{b_3} \wedge b^{b_1}_{b_3} \wedge b^{b_3}_{b_5}.$$

5. The graphs corresponding to a and b are shown in Figure 8.5.
6. The graphs are 2-colored, as shown in Figure 8.5, where nodes marked by $\otimes$ are colored black and the other nodes are colored white. These colorings generate the following clustering functions:

$$c_a(a_1) = c_a(a_3) = 1$$
$$c_a(a_2) = c_a(a_4) = 2$$
$$c_b(b_1) = c_b(b_2) = c_b(b_5) = 1$$
$$c_b(b_3) = c_b(b_4) = 2.$$

Given these clustering functions, one can construct a new decision system (see Table 8.12). ▫

Observe that discretization and symbolic attribute value grouping can be simultaneously used in decision systems including both real-value and symbolic attributes.

8.2.4 Minimal Decision Rules

In this section, techniques for constructing minimal rules for decision systems will be considered.

Given a decision table $\mathbb{A}$, a *minimal decision rule* (wrt $\mathbb{A}$) is a rule which is TRUE in $\mathbb{A}$ and which becomes FALSE in $\mathbb{A}$ if any elementary descriptor from the left-hand side of the rule is removed.[11]

The minimal number of elementary descriptors in the left-hand side of a minimal decision rule defines the largest subset of a decision class. Accordingly, information included in the conditional part of any minimal decision rule is sufficient for predicting the decision value of all objects satisfying this part of the rule. The conditional parts of minimal decision rules define the largest object sets relevant for

[11] A decision rule $\varphi \Rightarrow \psi$ is TRUE in $\mathbb{A}$ if and only if $\|\varphi\|_{\mathbb{A}} \subseteq \|\psi\|_{\mathbb{A}}$.

Table 8.13 Decision table considered in Example 8.10

Object	*L*	*W*	*C*	*S*
1	7.0	large	green	no
2	7.0	large	blue	no
3	4.0	medium	green	yes
4	4.0	medium	red	yes
5	5.0	medium	blue	no
6	4.5	medium	green	no
7	4.0	large	red	no

Table 8.14 $\{L,W\}$-reduction considered in Example 8.10

Objects	*L*	*W*	*S*
1, 2	7.0	large	no
3,4	4.0	medium	yes
5	5.0	medium	no
6	4.5	medium	no
7	4.0	large	no

approximating decision classes. The conditional parts of minimal decision rules can be computed using prime implicants.

To compute the set of all minimal rules wrt to a decision system $\mathbb{A} = (U,A,d)$, we proceed as follows, for any object $x \in U$:

1. Construct a decision-relative discernibility function f_x^r by considering the row corresponding to object x in the decision-relative discernibility matrix for $\mathbb{A}$.
2. Compute all prime implicants of f_x^r.
3. On the basis of the prime implicants, create minimal rules corresponding to x. To do this, consider the set $A(I)$ of attributes corresponding to propositional variables in I, for each prime implicant I, and construct the rule:

$$\left(\bigwedge_{a \in A(I)} (a = a(x)) \right) \Rightarrow d = d(x).$$

The following example illustrates the idea.

Example 8.10. Consider the decision system $\mathbb{A}$ whose decision table is provided in Table 8.13. Table 8.13 contains the values of conditional attributes of vehicles (L,W,C, standing for *Length, Width*, and *Color*, respectively), and a decision attribute S standing for *Small* which allows one to decide whether a given vehicle is small.

This system has exactly one decision-relative reduct consisting of attributes L and W. The $\{L,W\}$-reduction of $\mathbb{A}$ as shown in Table 8.14.

Table 8.14 results in the following set of non-minimal decision rules:

Table 8.15 Reduced decision - relative discernibility matrix from Example 8.10

	3	4
1	L,W	L,W,C
2	L,W,C	L,W,C
5	L,C	L,C
6	L	L,C
7	W,C	W

$$(L = 7.0) \wedge (W = \text{large}) \Rightarrow (S = \text{no})$$
$$(L = 4.0) \wedge (W = \text{medium}) \Rightarrow (S = \text{yes})$$
$$(L = 5.0) \wedge (W = \text{medium}) \Rightarrow (S = \text{no})$$
$$(L = 4.5) \wedge (W = \text{medium}) \Rightarrow (S = \text{no})$$
$$(L = 4.0) \wedge (W = \text{large}) \Rightarrow (S = \text{no}).$$

To obtain the minimal decision rules, we apply the construction provided above, for $x \in \{1, \ldots, 7\}$.

1. The decision-relative discernibility functions $f_1^r, \ldots, f_7^r$ are constructed on the basis of the reduced discernibility matrix shown in Table 8.15:

$$f_1^r \equiv (L \vee W) \wedge (L \vee W \vee C) \equiv (L \vee W)$$
$$f_2^r \equiv (L \vee W \vee C) \wedge (L \vee W \vee C) \equiv (L \vee W \vee C)$$
$$f_3^r \equiv (L \vee W) \wedge (L \vee W \vee C) \wedge (L \vee C) \wedge L \wedge (W \vee C)$$
$$\equiv (L \wedge W) \vee (L \wedge C)$$
$$f_4^r \equiv (L \vee W \vee C) \wedge (L \vee W \vee C) \wedge (L \vee C) \wedge (L \vee C) \wedge W$$
$$\equiv (L \wedge W) \vee (C \wedge W)$$
$$f_5^r \equiv (L \vee C) \wedge (L \vee C) \equiv (L \vee C)$$
$$f_6^r \equiv L \wedge (L \vee C) \equiv L$$
$$f_7^r \equiv (W \vee C) \wedge W \equiv W.$$

2. The following prime implicants are obtained from formulas $f_1^r, \ldots, f_7^r$:

   f_1^r: L, W
   f_2^r: L, W, C
   f_3^r: $L \wedge W$, $L \wedge C$
   f_4^r: $L \wedge W$, $C \wedge W$
   f_5^r: L, C
   f_6^r: L
   f_7^r: W.

3. Based on the prime implicants, minimal decision rules are created for objects $1, \ldots, 7$. For instance, from prime implicants L and W corresponding to f_1^r, the following minimal decision rules are generated based on object 1:

$$(L = 7.0) \Rightarrow (S = \mathsf{no})$$
$$(W = \mathsf{large}) \Rightarrow (S = \mathsf{no}).$$

On the basis of object 3 and prime implicants $L \wedge W$ and $L \wedge C$ for f_3^r we obtain the following rules:

$$(L = 4.0) \wedge (W = \mathsf{medium}) \Rightarrow (S = \mathsf{yes})$$
$$(L = 4.0) \wedge (C = \mathsf{green}) \Rightarrow (S = \mathsf{yes}).$$

Similarly, minimal decision rules can easily be obtained for all other formulas. □

In practice, the number of minimal decision rules can be large. One then tries to consider only subsets of these rules or to drop some conditions from minimal rules.

Remark 8.2. The main challenge in inducing rules from decision systems lies in determining which attributes should be included into the conditional parts of the rules. Using the strategy outlined above, the minimal rules are computed first. Their conditional parts describe the largest object sets with the same generalized decision value in a given decision system. Although such minimal decision rules can be computed, this approach can result in a set of rules of unsatisfactory classification quality. Such rules might appear too general or too specific for classifying new objects. This depends on the data analyzed. Techniques have been developed for the further tuning of minimal rules. □

8.2.5 Example: Learning of Concepts

Given that one has all the techniques described in the previous sections at one's disposal, an important task is to induce definitions of concepts from training data, where the representation of the definition is as efficient and of high quality as possible. These definitions may then be used as classifiers for the induced concepts.

Let us concentrate on the concept of *Distance* between cars on the road. The rough relation $Distance(x,y,z)$ denotes the approximate distance between vehicles x and y, where $z \in \{\mathsf{small}, \mathsf{medium}, \mathsf{large}, \mathsf{unknown}\}$. Below we simplify the definition somewhat, and consider $Distance(x,z)$ which denotes that the distance between x and the vehicle directly preceding x is z.[12] Assume that sample training data has been gathered in a decision table which is provided in Table 8.16, where[13]

- SL stands for the "speed limit" on a considered road segment;
- VS stands for the "vehicle speed";
- W stands for "weather conditions";

[12] In fact, here we consider a distance to be small if it causes a dangerous situation, and to be large if the situation is safe.

[13] Of course, real-life sample data would consist of hundreds or thousands of examples.

Table 8.16 Training data considered in Sect. 8.2.5

Object	*SL*	*VS*	*W*	*AD*	*Distance*
1	70	60	rain	3.0	small
2	70	70	sun	5.0	medium
3	50	60	rain	5.0	small
4	50	60	sun	9.0	medium
5	30	15	rain	9.0	large
6	30	10	sun	5.0	large
7	70	60	rain	15.0	large
8	50	40	rain	15.0	large

Table 8.17 Discernibility matrix of Table 8.16 for decision small

Object	1	3
2	VS, W, AD	SL, VS, W
4	SL, W, AD	W, AD
5	SL, VS, AD	SL, VS, AD
6	SL, VS, W, AD	SL, VS, W
7	AD	SL, AD
8	SL, VS, AD	VS, AD

- *AD* stands for "actual distance" between a given vehicle and its predecessor on the road.

For the sake of simplicity, we concentrate on generating rules to determine whether the distance between two objects is small.

On the basis of the training data, one can compute a discernibility matrix. Since we are interested in rules for the decision small only, it suffices to consider a simplified discernibility matrix with columns labelled by objects 1 and 3, as these are the only two objects, where the corresponding decision is small. The resulting discernibility matrix is shown in Table 8.17.

The discernibility matrix gives rise to the following discernibility functions:

$$\begin{aligned}
f_1 &\equiv (VS \lor W \lor AD) \land (SL \lor W \lor AD) \land (SL \lor VS \lor AD) \\
&\quad \land (SL \lor VS \lor W \lor AD) \land AD \land (SL \lor VS \lor AD) \\
&\equiv AD \\
f_3 &\equiv (SL \lor VS \lor W) \land (W \lor AD) \land (SL \lor VS \lor AD) \\
&\quad \land (SL \lor VS \lor W) \land (SL \lor AD) \land (VS \lor AD) \\
&\equiv (W \land AD) \lor (SL \land AD) \lor (VS \land AD) \lor (SL \land VS \land W).
\end{aligned}$$

Based on the discernibility functions, one can easily find prime implicants and obtain the following rules for the decision small:[14]

[14] In practical applications one would have to discretize *AD* before extracting rules.

Table 8.18 Information table considered in Example 8.11

Customer	*Bread*	*Milk*	*Jam*	*Beer*
1	yes	yes	no	no
2	yes	yes	yes	yes
3	yes	yes	yes	no
4	no	yes	yes	no

$$
\begin{aligned}
&(AD = 3.0) \Rightarrow (Distance = \mathsf{small}) \\
&(W = \mathsf{rain}) \wedge (AD = 5.0) \Rightarrow (Distance = \mathsf{small}) \\
&(SL = 50) \wedge (AD = 5.0) \Rightarrow (Distance = \mathsf{small}) \\
&(VS = 60) \wedge (AD = 5.0) \Rightarrow (Distance = \mathsf{small}) \\
&(SL = 50) \wedge (VS = 60) \wedge (W = \mathsf{rain}) \Rightarrow (Distance = \mathsf{small}).
\end{aligned}
\tag{8.10}
$$

There have been also developed methods for approximation of compound concepts based on rough sets, hierarchical learning, and ontology approximation (see, *e.g.*, [**?**, **?**, **?**, **?**, **?**, **?**, **?**, **?**, **?**, **?**, **?**, **?**]).

8.2.6 Association Rules

In this section [**?**, **?**], we show how rough set techniques can be used to extract association rules from information systems. Association rules playing an important role in the field of data mining, provide associations among attributes[15]. A real number from the interval [0,1] is assigned to each rule and provides a measure of the confidence of the rule. The following example will help to illustrate this.

Example 8.11. Consider the information table provided in Table 8.18.

Each row in the table represents items bought by a customer. For instance, customer 1 bought bread and milk, whereas customer 4 bought milk and jam. An association rule that can be extracted from the above table is: *a customer who bought bread also bought milk*. This is represented by

$$(Bread = \mathsf{yes}) \Rightarrow (Milk = \mathsf{yes}).$$

Since all customers who bought bread actually bought milk too, the confidence of this rule is 1. Now consider the rule

$$(Bread = \mathsf{yes}) \wedge (Milk = \mathsf{yes}) \Rightarrow (Jam = \mathsf{yes})$$

stating that a customer who bought bread and milk, bought jam as well. Since three customers bought both bread and milk and two of them bought jam, the confidence of this rule is 2/3. ▫

[15] Association between attributes are also studied using association reducts [**?**].

We now formalize this approach to confidence measures for association rules. Recall that by a template we mean a conjunction of elementary descriptors, *i.e.*, expressions of the form $a = v$, where a is an attribute and $v \in V_a$. For an information system $\mathbb{A}$ and a template T we denote by $support_{\mathbb{A}}(T)$ the number of objects satisfying T. Let $\mathbb{A}$ be an information system and $T = D_1 \wedge \ldots \wedge D_m$ be a template. By an *association rule generated from T*, we mean any expression of the form

$$\bigwedge_{D_i \in P} D_i \Rightarrow \bigwedge_{D_j \in Q} D_j,$$

where $\{P, Q\}$ is a partition of $\{D_1, \ldots, D_m\}$. By a *confidence of an association rule* $\phi \equiv \bigwedge_{D_i \in P} D_i \Rightarrow \bigwedge_{D_j \in Q} D_j$ we mean the coefficient

$$confidence_{\mathbb{A}}(\phi) = \frac{support_{\mathbb{A}}(D_1 \wedge \ldots \wedge D_m)}{support_{\mathbb{A}}(\bigwedge_{D_i \in P} D_i)}.$$

There are two basic steps used in methods aimed at generating association rules. (Below s and c stand for support and confidence thresholds wrt a given information system $\mathbb{A}$, respectively.)

1. Generate as many templates $T = D_1 \wedge \ldots \wedge D_k$ as possible, such that $support_{\mathbb{A}}(T) \geq s$ and $support_{\mathbb{A}}(T \wedge D_i) < s$, for any descriptor D_i different from all descriptors $D_1, \ldots, D_k$.
2. Search for a partition $\{P, Q\}$ of T, for each T generated in the previous step, satisfying

   a. $support_{\mathbb{A}}(P) < \dfrac{support_{\mathbb{A}}(T)}{c}$

   b. P has the shortest length among templates satisfying (a).

   Every such partition leads to an association rule of the form $P \Rightarrow Q$ whose confidence is greater than c.

The second step, crucial to the process of extracting association rules, can be solved using rough set methods.

Let $T = D_1 \wedge D_2 \wedge \ldots \wedge D_m$ be a template such that $support_{\mathbb{A}}(T) \geq s$. For a given confidence threshold $c \in [0,1]$, the association rule $\phi \equiv P \Rightarrow Q$ is called *c-irreducible* if $confidence_{\mathbb{A}}(P \Rightarrow Q) \geq c$ and for any association rule $\phi' \equiv P' \Rightarrow Q'$ such that P' is a sub-formula of P, we have

$$confidence_{\mathbb{A}}(P' \Rightarrow Q') < c.$$

The problem of searching for c-irreducible association rules from a given template is equivalent to the problem of searching for α-reducts in a decision table, for some $\alpha \in [0,1]$ (see Sect. 8.2.1).

Let $\mathbb{A}$ be an information system and $T = D_1 \wedge D_2 \wedge \ldots \wedge D_m$ be a template. By a *characteristic table* for T wrt $\mathbb{A}$, we understand a decision system $\mathbb{A}|_T = (U, A|_T, d)$, where

1. $A|_T = \{a_{D_1}, a_{D_2}, ..., a_{D_m}\}$ is a set of attributes corresponding to the descriptors of T such that

$$a_{D_i}(u) = \begin{cases} 1, \text{ if the object } u \text{ satisfies } D_i, \\ 0, \text{ otherwise;} \end{cases}$$

2. the decision attribute d determines if the object satisfies a template T, *i.e.*,

$$d(u) = \begin{cases} 1, \text{ if the object } u \text{ satisfies } T, \\ 0, \text{ otherwise.} \end{cases}$$

The following property provides the relationship between association rules and approximations of reducts.

For a given information system $\mathbb{A} = (U, A)$, a template $T = D_1 \wedge D_2 \wedge \ldots \wedge D_m$ and a set of descriptors $P \subseteq \{D_1, \ldots, D_m\}$, the association rule

$$\bigwedge_{D_i \in P} D_i \quad \Rightarrow \bigwedge_{D_j \in \{D_1,...,D_m\} - P} D_j,$$

is

1. a 1-irreducible association rule from T if and only if $\bigcup_{D_i \in P} \{a_{D_i}\}$ is a decision-relative reduct of $\mathbb{A}|_T$;
2. a c-irreducible association rule from T if and only if $\bigcup_{D_i \in P} \{a_{D_i}\}$ is an α-reduct of $\mathbb{A}|_T$, where

$$\alpha = 1 - \left[\left(\frac{1}{c} - 1 \right) \Big/ \left(\frac{|U|}{support_{\mathbb{A}}(T)} - 1 \right) \right].$$

The problem of searching for the shortest association rules is NP-hard.

The following example illustrates the main ideas used in the searching method for association rules.

Example 8.12. Consider the information table $\mathscr{A}$ with 18 objects and 9 attributes presented in Table 8.19.

Consider the template

$$T = (a_1 = 0) \wedge (a_3 = 2) \wedge (a_4 = 1) \wedge (a_6 = 0) \wedge (a_8 = 1). \tag{8.11}$$

It is easily seen that $support_{\mathscr{A}}(T) = 10$. The new constructed decision table $\mathscr{A}|_T$ is presented in Table 8.20.

The reduced discernibility matrix $\mathscr{A}|_T$ is provided in Table 8.21, where for simplicity, the second column represents, in fact, ten columns with identical contents,

Table 8.19 Information table $\mathbb{A}$ considered in Example 8.12

$\mathscr{A}$	a_1	a_2	a_3	a_4	a_5	a_6	a_7	a_8	a_9
u_1	0	1	1	1	80	2	2	2	3
u_2	0	1	2	1	81	0	aa	1	aa
u_3	0	2	2	1	82	0	aa	1	aa
u_4	0	1	2	1	80	0	aa	1	aa
u_5	1	1	2	2	81	1	aa	1	aa
u_6	0	2	1	2	81	1	aa	1	aa
u_7	1	2	1	2	83	1	aa	1	aa
u_8	0	2	2	1	81	0	aa	1	aa
u_9	0	1	2	1	82	0	aa	1	aa
u_{10}	0	3	2	1	84	0	aa	1	aa
u_{11}	0	1	3	1	80	0	aa	2	aa
u_{12}	0	2	2	2	82	0	aa	2	aa
u_{13}	0	2	2	1	81	0	aa	1	aa
u_{14}	0	3	2	2	81	2	aa	2	aa
u_{15}	0	4	2	1	82	0	aa	1	aa
u_{16}	0	3	2	1	83	0	aa	1	aa
u_{17}	0	1	2	1	84	0	aa	1	aa
u_{18}	1	2	2	1	82	0	aa	2	aa

Table 8.20 Decision table $\mathbb{A}|_T$ considered in Example 8.12

$\mathscr{A}\|_T$	a_{D_1} $(a_1 = 0)$	a_{D_2} $(a_3 = 2)$	a_{D_3} $(a_4 = 1)$	a_{D_4} $(a_6 = 0)$	a_{D_5} $(a_8 = 1)$	d
u_1	1	0	1	0	0	0
u_2	1	1	1	1	1	1
u_3	1	1	1	1	1	1
u_4	1	1	1	1	1	1
u_5	0	1	0	0	1	0
u_6	1	0	0	0	1	0
u_7	0	0	0	0	1	0
u_8	1	1	1	1	1	1
u_9	1	1	1	1	1	1
u_{10}	1	1	1	1	1	1
u_{11}	1	0	1	1	0	0
u_{12}	1	0	0	1	0	0
u_{13}	1	1	1	1	1	1
u_{14}	1	1	0	0	0	0
u_{15}	1	1	1	1	1	1
u_{16}	1	1	1	1	1	1
u_{17}	1	1	1	1	1	1
u_{18}	0	1	1	1	0	0

Table 8.21 Reduced discernibility matrix for $\mathscr{A}|_T$ from Example 8.12

$\mathscr{M}(\mathscr{A}|_T)$	u_2, u_3, u_4, u_8, u_9 $u_{10}, u_{13}, u_{15}, u_{16}, u_{17}$
u_1	$a_{D_2}, a_{D_4}, a_{D_5}$
u_5	$a_{D_1}, a_{D_3}, a_{D_4}$
u_6	$a_{D_2}, a_{D_3}, a_{D_4}$
u_7	$a_{D_1}, a_{D_2}, a_{D_3}, a_{D_4}$
u_{11}	$a_{D_1}, a_{D_3}, a_{D_5}$
u_{12}	$a_{D_2}, a_{D_3}, a_{D_5}$
u_{14}	$a_{D_3}, a_{D_4}, a_{D_5}$
u_{18}	a_{D_1}, a_{D_5}

labeled by $u_2, u_3, u_4, u_8, u_9, u_{10}, u_{13}, u_{15}, u_{16}, u_{17}$, respectively. Given the discernibility matrix, one can easily compute the discernibility function $\mathscr{A}|_T$ for $\mathscr{A}|_T$:

$$\begin{aligned} f_{\mathbb{A}_T}(a_{D_1}, a_{D_2}, a_{D_3}, a_{D_4}, a_{D_5}) \equiv\ & (a_{D_2} \vee a_{D_4} \vee a_{D_5}) \\ & \wedge(a_{D_1} \vee a_{D_3} \vee a_{D_4}) \\ & \wedge(a_{D_2} \vee a_{D_3} \vee a_{D_4}) \\ & \wedge(a_{D_1} \vee a_{D_2} \vee a_{D_3} \vee a_{D_4}) \\ & \wedge(a_{D_1} \vee a_{D_3} \vee a_{D_5}) \\ & \wedge(a_{D_2} \vee a_{D_3} \vee a_{D_5}) \\ & \wedge(a_{D_3} \vee a_{D_4} \vee a_{D_5}) \\ & \wedge(a_{D_1} \vee a_{D_5}), \end{aligned}$$

where D_i denotes the i-th conjunct of (8.11).

The discernibility function has the following prime implicants: $a_{D_3} \wedge a_{D_5}$, $a_{D_4} \wedge a_{D_5}$, $a_{D_1} \wedge a_{D_2} \wedge a_{D_3}$, $a_{D_1} \wedge a_{D_2} \wedge a_{D_4}$, $a_{D_1} \wedge a_{D_2} \wedge a_{D_5}$, $a_{D_1} \wedge a_{D_3} \wedge a_{D_4}$. This gives rise to the reducts: $\{a_{D_3}, a_{D_5}\}$, $\{a_{D_4}, a_{D_5}\}$, $\{a_{D_1}, a_{D_2}, a_{D_3}\}$, $\{a_{D_1}, a_{D_2}, a_{D_4}\}$, $\{a_{D_1}, a_{D_2}, a_{D_5}\}$, $\{a_{D_1}, a_{D_3}, a_{D_4}\}$. Thus, there are 6 association rules with confidence 1, *i.e.*, 1-irreducible:

$$\begin{aligned} D_3 \wedge D_5 &\Rightarrow D_1 \wedge D_2 \wedge D_4 \\ D_4 \wedge D_5 &\Rightarrow D_1 \wedge D_2 \wedge D_3 \\ D_1 \wedge D_2 \wedge D_3 &\Rightarrow D_4 \wedge D_5 \\ D_1 \wedge D_2 \wedge D_4 &\Rightarrow D_3 \wedge D_5 \\ D_1 \wedge D_2 \wedge D_5 &\Rightarrow D_3 \wedge D_4 \\ D_1 \wedge D_3 \wedge D_4 &\Rightarrow D_2 \wedge D_5. \end{aligned}$$

For confidence 0.9, we look for α-reducts for the decision table $\mathscr{A}|_T$, where

$$\alpha = 1 - \left(\frac{1}{0.9} - 1\right) / \left(\frac{18}{10} - 1\right) \approx 0.86.$$

Hence, we look for a set of descriptors that covers at least $\lceil (18-10)*\alpha \rceil = \lceil 8*0.86 \rceil = 7$ elements of the discernibility matrix $\mathscr{M}(\mathscr{A}|_{\mathbf{T}})$. One can see that the following sets of descriptors: $\{D_1,D_2\}$, $\{D_1,D_3\}$, $\{D_1,D_4\}$, $\{D_1,D_5\}$, $\{D_2,D_3\}$, $\{D_2,D_5\}$, $\{D_3,D_4\}$ have nonempty intersections with exactly 7 members of the discernibility matrix $\mathscr{M}(\mathscr{A}|_T)$. Consequently, the 0.9-irreducible association rules obtained from those sets are the following:

$$
\begin{aligned}
D_1 \wedge D_2 &\Rightarrow D_3 \wedge D_4 \wedge D_5 \\
D_1 \wedge D_3 &\Rightarrow D_2 \wedge D_4 \wedge D_5 \\
D_1 \wedge D_4 &\Rightarrow D_2 \wedge D_3 \wedge D_5 \\
D_1 \wedge D_5 &\Rightarrow D_2 \wedge D_3 \wedge D_4 \\
D_2 \wedge D_3 &\Rightarrow D_1 \wedge D_4 \wedge D_5 \\
D_2 \wedge D_5 &\Rightarrow D_1 \wedge D_3 \wedge D_4 \\
D_3 \wedge D_4 &\Rightarrow D_1 \wedge D_2 \wedge D_5.
\end{aligned}
$$

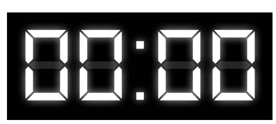
The technique illustrated by this example can be applied to find useful dependencies between attributes in complex application domains. In particular, one could use such dependencies in constructing robust classifiers conforming to the laws of the underlying reality. □

The following exercises are provided by Professor Hung Son Nguyen.

Problem 8.1. Digital Clock Font.

Each digit of the following 24 hours Digital Clock is made of a certain number of dashes, as shown in the image below.

Each dash is displayed by a LED (light-emitting diode). Therefore the clock consists of 28 dashes.



Assume that we want to switch off some LEDs to save the energy. Design a decision table to store the information about those digits and use the rough set methods to solve the following problems:

1. For the third position (ten digit of minute value), only digits 0, 1, 2,3,4,5 can be displayed. What is the minimal number of dashes necessary to discern those values.
2. For the forth position (unit digit of minute value), what is the minimal set of dashes you want to use if we want to recognize the parity of the displayed digit?

3. For the first two positions (the hour value), there are 14 dashes but only values $00, 01, \ldots, 23$ can be displayed. What is the minimal number of dashes necessary to discern those values.

Problem 8.2. Core attributes.
Propose an algorithm of searching for all core attributes that does not use the discernibility matrix and has time complexity of $O(k \cdot n \log n)$.

Problem 8.3. Decision table with maximal number of reducts.
We know that the number of reducts for any decision table $\mathbb{S}$ with m attributes can not exceed the upper bound

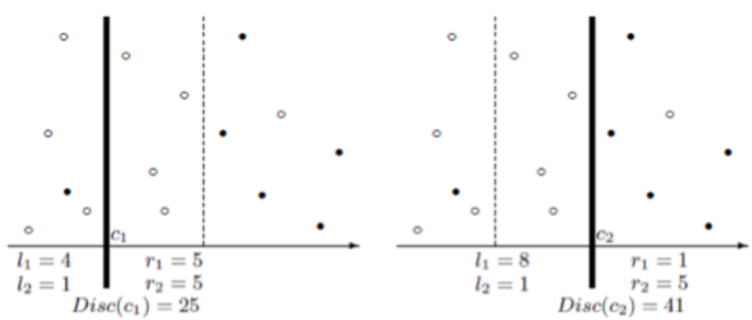
$$N(m) = \binom{m}{\lfloor m/2 \rfloor}.$$

For any integer m construct a decision table with m attributes such that the number of reducts for this table equals to $N(m)$.

Problem 8.4. Decision rules vs. decision tree.

1. True or false:
   ”Each path of a minimal decision tree is a minimal consistent decision rule" ?
   Justify your answer.
2. Find the maximal possible number $M(k)$ of minimal and consistent decision rules for a decision table with k attributes?

Problem 8.5. Boundary cuts.
Recall that boundary cut on an attribute is the cut that two nearest objects are from different decision classes. In the following figure, the cut c_1 is not a boundary cut and the cut c_2 is the example of boundary cut.



Prove that if c is the best cut with respect to discernibility measure for an attribute a then c must be one of the boundary cut.

Problem 8.6. Are the best cuts really good?

A real number $v_i \in a(U)$ is called *single value* of an attribute a if there is exactly one object $u \in U$ such that $a(u) = v_i$. The cut $(a;c)$ is called *the single cut* if c is lying between two single values v_i and v_{i+1}.

Prove the following properties related to single cuts:

Theorem. Let DT be a decision system with two decision classes and real valued conditional attribute a. Any single cut c_i of a realising a local maximum of the function *Disc*, resolves at least half of conflicts in the decision table DT, i.e.

$$Disc\,(c_i) \geq \frac{1}{2} \cdot \text{conflict}\,(\mathbb{S})\,.$$

where conflict$\,(\mathbb{S})$ is the number pairs of objects from different decision classes.

Problem 8.7. Decision tree build by MD-heuristics?

Prove that if all cuts of a decision table with two decision classes are single, then the decision tree build by MD-heuristics has height as most $2\log n - 1$.

References

1.
2. An, A., Huang, Y., Huang, X., Cercone, N.: Feature selection with rough sets for web page classification. In: Transactions on Rough Sets **II**. Rough sets and fuzzy sets: Journal Subline [**?**], pp. 1–13. doi:10.1007/11548669_20
3. Banerjee, M., Mitra, S., Anand, A.: Feature selection using rough sets. In: Y. Jin (ed.) Multi-Objective Machine Learning, *Studies in Computational Intelligence*, vol. 16, pp. 3–20. Springer, Heidelberg (2006). doi:10.1007/3-540-33019-4_1
4. Banerjee, M., Mitra, S., Banka, H.: Evolutionary-rough feature selection in gene expression data. IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews **37**, 622–632 (2007). doi:10.1109/TSMCC.2007.897498
5. Banerjee, M., Pal, S.K.: Roughness of a fuzzy set. Information Sciences **93**(3-4), 235–246 (1996). doi:10.1016/j.ins.2004.07.017
6. Banka, H., Mitra, S.: Feature selection, classification and rule generation using rough sets. In: Rough Sets: Selected Methods and Applications in Management and Engineering, pp. 51–76 (2012). doi:10.1007/978-1-4471-2760-4_4
7. Bargiela, A., Pedrycz, W.: Granular Computing: An Introduction. Kluwer Academic Publishers (2003)
8. Bazan, J.: Hierarchical classifiers for complex spatio-temporal concepts. In: J.F. Peters, A. Skowron, H. Rybiński (eds.) Transactions on Rough Sets **IX**: Journal Subline, *Lecture Notes in Computer Science*, vol. 5390, pp. 474–750. Springer, Heidelberg (2008). doi:10.1007/978-3-540-89876-4_26
9. Bazan, J., Latkowski, R., Szczuka, M.: DIXER - Distributed executor for rough set exploration system. In: Proceedings of the 10th International Conference on Rough Sets, Fuzzy Sets, Data Mining, and Granular Computing (RSFDGrC'2005), Regina, Canada,August 31-September 3, 2005, Part II, pp. 362–371 (2005). doi:10.1007/11548706_5
10. Bazan, J., Nguyen, H.S., Nguyen, S.H., Synak, P., Wróblewski, J.: Rough set algorithms in classification problems. In: Polkowski et al. [**?**], pp. 49–88. doi:10.1007/978-3-7908-1840-6_3
11. Bazan, J., Skowron, A.: On-line elimination of non-relevant parts of complex objects in behavioral pattern identification. In: Proceedings of the 1st International Conference on Pattern

Recognition and Machine Intelligence (PReMI 2005) December 18-22, 2005, Kolkata, India, pp. 720–725 (2005). doi:10.1007/11590316_116
12. Bazan, J., Skowron, A., Ślezak, D., Wróblewski, J.: Searching for the complex decision reducts: The case study of the survival analysis. In: N. Zhong, Z.W. Ras, S. Tsumoto, E. Suzuki (eds.) Foundations of Intelligent Systems, 14th International Symposium on Methodologies for Intelligent Systems (ISMIS 2003), Maebashi City, Japan, October 28-31, 2003, *Lecture Notes in Artificial Intelligence*, vol. 2871, pp. 160–168. Springer, Heidelberg (2003). doi:10.1007/978-3-540-39592-8_22
13. Bazan, J., Szczuka, M., Wojna, M., Wojnarski, M.: On the evolution of rough set exploration system. In: Proceedings of the 4th International Conference on Rough Sets and Current Trends in Computing (RSCTC'2004), Uppsala, Sweden, June 1-5, 2004, pp. 592–601. doi:10.1007/978-3-540-25929-9_73
14. Bazan, J.G.: A comparison of dynamic and non-dynamic rough set methods for extracting laws from decision tables. In: Polkowski and Skowron [**?**], pp. 321–365
15. Bazan, J.G., Nguyen, H.S., Peters, J.F., Skowron, A., Szczuka, M.: Rough set approach to pattern extraction from classifiers. In: Proceedings of the Workshop on Rough Sets in Knowledge Discovery and Soft Computing at ETAPS 2003, April 12-13, 2003, pp. 20–29. doi:10.1016/S1571-0661(04)80702-3
16. Bazan, J.G., Peters, J.F., Skowron, A.: Behavioral pattern identification through rough set modelling. In: Proceedings of the 10th International Conference on Rough Sets, Fuzzy Sets, Data Mining, and Granular Computing (RSFDGrC'2005), Regina, Canada,August 31-September 3, 2005, Part II, pp. 688–697 (2005). doi:10.1007/11548706_73
17. Bazan, J.G., Skowron, A.: Classifiers based on approximate reasoning schemes. In: B. Dunin-Keplicz, A. Jankowski, A. Skowron, M. Szczuka (eds.) Monitoring, Security, and Rescue Tasks in Multiagent Systems (MSRAS'2004), Advances in Soft Computing, pp. 191–202. Springer, Heidelberg (2005). doi:0.1007/978-3-7908-1773-7_35
18. Bazan, J.G., Szczuka, M.: RSES and RSESlib - a collection of tools for rough set computations. In: Ziarko and Yao [**?**], pp. 106–113. doi:10.1007/3-540-45554-X_12
19. Boole, G. (ed.): An Investigation of the Laws of Thought on which are founded the Mathematical Theories of Logic and Probabilities. Mac Millan and Co., London (1854)
20. Brown, F.: Boolean Reasoning. Kluwer Academic Publishers, Dordrecht (1990)
21. Chmielewski, M.R., Grzymała-Busse, J.W.: Global discretization of continuous attributes as preprocessing for machine learning. International Journal of Approximate Reasoning **15**(4), 319–331 (1996). doi:10.1016/S0888-613X(96)00074-6
22. Choubey, S.K., Deogun, J.S., Raghavan, V.V., Sever, H.: A comparison of feature selection algorithms in the context of rough classifiers. In: F. Petry (ed.) International Conference on Fuzzy Systems (FUZZ-IEEE '1996), September 8-11, 1996, New Orleans, LA, vol. 2, pp. 1122–1128. IEEE Service Center, Pistcataway, NJ (1996). doi:10.1109/FUZZY.1996.561296
23. Cios, K., Pedrycz, W., Swiniarski, R., Kurgan, L.A.: Data mining. A Knowledge Discovery Approach. Springer, New York (2007)
24. Deogun, J., Raghavan, V.V., Sarkar, A., Sever, H.: Data mining: Trends in research and development. In: Lin and Cercone [**?**], pp. 9–46
25. Doherty, P., Łukaszewicz, W., Skowron, A., Szałas, A.: Knowledge Engineering: A Rough Set Approach, *Studies in Fuzziness and Soft Computing*, vol. 202. Springer, Heidelberg (2006)
26. Dubois, V., Quafafou, M.: Concept learning with approximation: Rough version spaces. In: Third International Conference on Rough Sets and Current Trends in Computing (RSCTC'2002), Malvern, PA, October 14-16, 2002, pp. 239–246 (2002). doi:10.1007/3-540-45813-1_31
27. Duda, R., Hart, P., Stork, R.: Pattern Classification. John Wiley & Sons, New York, NY (2002)
28. Düntsch, I., Gediga, G.: Rough set data analysis: A road to non-invasive knowledge discovery. Methodos Publishers, Bangor, UK (2000)
29. Gediga, G., Düntsch, I.: Rough approximation quality revisited. Artificial Intelligence **132**, 219–234 (2001). doi:10.1016/s0004-3702(01)00147-3

30. Gediga, G., Düntsch, I.: On model evaluation, indices of importance, and interaction values in rough set analysis. In: Pal et al. [**?**], pp. 251–276. doi:0.1007/978-3-642-18859-6_10
31. Góra, G., Wojna, A.G.: RIONA: A new classification system combining rule induction and instance-based learning. Fundamenta Informaticae **51**(4), 369–390 (2002)
32. Greco, S., Inuiguchi, M., Słowiński, R.: Fuzzy rough sets and multiple-premise gradual decision rules. International Journal of Approximate Reasoning **41**(2), 179–211 (2006). doi:10.1016/j.ijar.2005.06.014
33. Grzymała-Busse, J.W.: Selected algorithms of machine learning from examples. Fundamenta Informaticae **18**, 193–207 (1993)
34. Grzymała-Busse, J.W.: Classification of unseen examples under uncertainty. Fundamenta Informaticae **30**(3-4), 255–267 (1997). doi:10.3233/FI-1997-303403
35. Grzymała-Busse, J.W.: A new version of the rule induction system LERS. Fundamenta Informaticae **31**(1), 27–39 (1997). doi:10.3233/FI-1997-3113
36. Grzymała-Busse, J.W.: Three strategies to rule induction from data with numerical attributes. In: Transactions on Rough Sets II. Rough sets and fuzzy sets: Journal Subline [**?**], pp. 54–62. doi:10.1007/978-3-540-27778-1_4
37. Grzymała-Busse, J.W.: LERS - A data mining system. In: Maimon and Rokach [**?**], pp. 1347–1351. doi:10.1007/0-387-25465-X_65
38. Grzymała-Busse, J.W.: Rule induction. In: Maimon and Rokach [**?**], pp. 277–294. doi:10.1007/0-387-25465-X_13
39. Grzymala-Busse, J.W.: Generalized parameterized approximations. In: Proceedings of the 6th International Conference on Rough Sets and Knowledge Technology (RSKT'2011), Banff, Canada, October 9-12, 2011, pp. 136–145 (2011)
40. Grzymała-Busse, J.W., Grzymała-Busse, W.J.: Handling missing attribute values. In: Maimon and Rokach [**?**], pp. 37–57. doi:10.1007/0-387-25465-X_3
41. Grzymała-Busse, J.W., Ziarko, W.: Data mining and rough set theory. Communications of the ACM **43**, 108–109 (2000). doi:10.1145/332051.332082
42. Hastie, T., Tibshirani, R., Friedman, J.H.: The Elements of Statistical Learning: Data Mining, Inference, and Prediction. Springer-Verlag, Heidelberg (2009). doi:10.1007/b94608
43. Herbert, J., Yao, J.T.: Time-series data analysis with rough sets. In: Proceedings of the 4th International Conference on Computational Intelligence in Economics and Finance (CIEF'2005), Salt Lake City, UT, July 21-26, 2005, pp. 908–911 (2005)
44. Hu, Q., Pedrycz, W., Yu, D., Lang, J.: Selecting discrete and continuous features based on neighborhood decision error minimization. IEEE Transactions on Systems, Man, and Cybernetics-Part B: Cybernetics **40**, 137–150 (2010). doi:10.1109/TSMCB.2009.2024166
45. Hu, Q.H., Yu, D.R., Liu, J.F., X.Wu, C.: Neighborhood rough set based heterogeneous feature subset selection. Information Sciences **178**, 3577–3594 (2008). doi:10.1016/j.ins.2008.05.024
46. Hu, X., Cercone, N.: Learning in relational databases: A rough set approach. Computational Intelligence: An International Journal **11**(2), 323–338 (1995). doi:10.1007/BF03325090
47. Hu, X., Cercone, N.: Data mining via discretization, generalization and rough set feature selection. Knowledge and Information Systems: An International Journal **1**(1), 33–60 (1999). doi:10.1007/BF03325090
48. Hu, X., Cercone, N.: Discovering maximal generalized decision rules through horizontal and vertical data reduction. Computational Intelligence: An International Journal **17**(4), 685–702 (2001). doi:10.1111/0824-7935.00169
49. I. Chikalov, V. Lozin, I. Lozina, M. Moshkov, H. S. Nguyen, A. Skowron, B. Zielosko: Three Approaches to Data Analysis. Test Theory, Rough Sets and Logical Analysis of Data, *Series Intelligent Systems Reference Library*, vol. 41. Springer (2012). doi:10.1007/978-3-642-28667-4
50. Jankowski, A., Skowron, A., Wasilewski, P.: Rough sets and sorites paradox. Fundamenta Informaticae **157**(4), 371–384 (2018). doi:10.3233/FI-2018-1633
51. Jensen, R.: Rough set-based feature selection: A review. In: A.E. Hassanien, Z. Suraj, D. Slezak, P. Lingras (eds.) Rough Computing: Theories, Technologies and Applications, pp. 70–107. IGI Global, Hershey, PA (2008). doi:10.4018/978-1-59904-552-8.ch003

52. Kim, D.: Data classification based on tolerant rough set. Pattern Recognition **34**(8), 1613–1624 (2001). doi:10.1016/S0031-3203(00)00057-1
53. Kim, D., Bang, S.Y.: A handwritten numeral character classification using tolerant rough set. IEEE Transactions on Pattern Analysis and Machine Intelligence **22**(9), 923–937 (2000). doi:10.1109/34.877516
54. Kloesgen, W., Żytkow, J. (eds.): Handbook of Knowledge Discovery and Data Mining. Oxford University Press, Oxford (2002)
55. Komorowski, J., Øhrn, A., Skowron, A.: Rosetta and other software systems for rough sets. In: W. Klösgen, J. Żytkow (eds.) Handbook of Data Mining and Knowledge Discovery, pp. 554–559. Oxford University Press (2000)
56. Kryszkiewicz, M., Rybiński, H.: Computation of reducts of composed information systems. Fundamenta Informaticae **27**(2-3), 183–195 (1996). doi:10.3233/FI-1996-272306
57. Latkowski, R.: On decomposition for incomplete data. Fundamenta Informaticae **54**(1), 1–16 (2003)
58. Latkowski, R.: Flexible indiscernibility relations for missing attribute values. Fundamenta Informaticae **67**(1-3), 131–147 (2005)
59. Li, J., Cercone, N.: A rough set based model to rank the importance of association rules. In: Proceedings of the 10th International Conference on Rough Sets, Fuzzy Sets, Data Mining, and Granular Computing (RSFDGrC'2005), Regina, Canada,August 31-September 3, 2005, Part II, pp. 109–118 (2005). doi:10.1007/11548706_12
60. Li, Y., Shiu, S.C.K., Pal, S.K., Liu, J.N.K.: A rough set-based case-based reasoner for text categorization. International Journal of Approximate Reasoning **41**(2), 229–255 (2006). doi:10.1016/j.ijar.2005.06.019
61. Lin, T.Y., Cercone, N. (eds.): Rough Sets and Data Mining - Analysis of Imperfect Data. Kluwer Academic Publishers, Boston, MA (1997)
62. Lingras, P.: Fuzzy - rough and rough - fuzzy serial combinations in neurocomputing. Neurocomputing **36**(1-4), 29–44 (2001). doi:10.1016/S0925-2312(00)00334-9
63. Lingras, P.: Unsupervised rough set classification using GAs. Journal of Intelligent Information Systems **16**(3), 215–228 (2001). doi:10.1023/A:1011219918340
64. Lingras, P., West, C.: Interval set clustering of Web users with rough *K*-means. Journal of Intelligent Information Systems **23**(1), 5–16 (2004). doi:10.1023/B:JIIS.0000029668.88665.1a
65. Maimon, O., Rokach, L. (eds.): The Data Mining and Knowledge Discovery Handbook. Springer, Heidelberg (2005). doi:10.1007/b107408
66. Mazumdar, D., Mitra, S., Mitra, S.: Evolutionary-rough feature selection for face recognition. In: J.F. Peters, A. Skowron, R. Słowiński, P. Lingras, D. Miao, S. Tsumoto (eds.) Transactions on Rough Sets IV: Journal Subline, *Lecture Notes in Computer Science*, vol. 6190, pp. 117–142. Springer, Heidelberg (2010). doi:10.1007/978-3-642-14467-7_7
67. Min, F., Hu, Q., Zhu, W.: Feature selection with test cost constraint. International Journal of Approximate Reasoning pp. 167–179 (2014). doi:10.1016/j.ijar.2013.04.003
68. Mitchel, T.M.: Machine Learning. McGraw-Hill Series in Computer Science, Boston, MA (1999)
69. Mitra, P., Mitra, S., Pal, S.K.: Modular rough fuzzy MLP: Evolutionary design. In: Proceedings of the 7-th International Workshop on Rough Sets, Fuzzy Sets, Data Mining, and Granular-Soft Computing (RSFDGrC'99), Yamaguchi, November 9-11, 1999, pp. 128–136 (1999). doi:10.1007/978-3-540-48061-7_17
70. Mitra, P., Pal, S.K., Siddiqi, M.A.: Non-convex clustering using expectation maximization algorithm with rough set initialization. Pattern Recognition Letters **24**(6), 863–873 (2003). doi:10.1016/S0167-8655(02)00198-8
71. Nakata, M., Sakai, H.: Rough sets handling missing values probabilistically interpreted. In: Proceedings of the 10th International Conference on Rough Sets, Fuzzy Sets, Data Mining, and Granular Computing (RSFDGrC'2005), Regina, Canada, August 31-September 3, 2005, Part I, pp. 325–334 (2005). doi:10.1007/11548669_34
72. Nguyen, H.S.: Discretization of real value attributes, boolean reasoning approach. Ph.D. thesis, Warsaw University, Warsaw, Poland (1997)

73. Nguyen, H.S.: From optimal hyperplanes to optimal decision trees. Fundamenta Informaticae **34**(1-2), 145–174 (1998). doi:10.3233/FI-1998-341206
74. Nguyen, H.S.: Efficient SQL-learning method for data mining in large data bases. In: T. Dean (ed.) Sixteenth International Joint Conference on Artificial Intelligence IJCAI, pp. 806–811. Morgan-Kaufmann Publishers, Stockholm, Sweden (1999)
75. Nguyen, H.S.: On efficient handling of continuous attributes in large data bases. Fundamenta Informaticae **48**(1), 61–81 (2001)
76. Nguyen, H.S.: Approximate boolean reasoning approach to rough sets and data mining. In: Proceedings of the 10th International Conference on Rough Sets, Fuzzy Sets, Data Mining, and Granular Computing (RSFDGrC'2005), Regina, Canada,August 31-September 3, 2005, Part II, pp. 12–22 (2005). doi:10.1007/11548706_2. (plenary talk)
77. Nguyen, H.S.: Approximate boolean reasoning: Foundations and applications in data mining. In: J.F. Peters, A. Skowron (eds.) Transactions on Rough Sets V: Journal Subline, *Lecture Notes in Computer Science*, vol. 4100, pp. 344–523. Springer, Heidelberg (2006). doi::10.1007/11847465_16
78. Nguyen, H.S., Jankowski, A., Skowron, A., Stepaniuk, J., Szczuka, M.: Discovery of process models from data and domain knowledge: A rough-granular approach. In: J.T. Yao (ed.) Novel Developments in Granular Computing: Applications for Advanced Human Reasoning and Soft Computation, pp. 16–47. IGI Global, Hershey, PA (2010)
79. Nguyen, H.S., Nguyen, S.H.: Pattern extraction from data. Fundamenta Informaticae **34**, 129–144 (1998). doi:10.3233/FI-1998-341205
80. Nguyen, H.S., Nguyen, S.H.: Rough sets and association rule generation. Fundamenta Informaticae **40**(4), 383–405 (1999). doi:10.3233/FI-1999-40403
81. Nguyen, H.S., Skowron, A.: Quantization of real value attributes. In: Proceedings of the Second Joint Annual Conference on Information Sciences, Wrightsville Beach, North Carolina, 1995, USA, pp. 34–37 (1995)
82. Nguyen, H.S., Skowron, A.: Rough sets: From rudiments to challenges. In: Skowron and Suraj [**?**], pp. 75–173. doi:10.1007/978-3-642-30344-9_3
83. Nguyen, H.S., Ślezak, D.: Approximate reducts and association rules - correspondence and complexity results. In: Proceedings of the 7-th International Workshop on Rough Sets, Fuzzy Sets, Data Mining, and Granular-Soft Computing (RSFDGrC'1999), Yamaguchi, November 9-11, 1999, pp. 137–145 (1999). doi:10.1007/978-3-540-48061-7_18
84. Nguyen, S.H.: Regularity analysis and its applications in data mining. In: Polkowski et al. [**?**], pp. 289–378. doi:10.1007/978-3-7908-1840-6_7
85. Nguyen, S.H., Bazan, J., Skowron, A., Nguyen, H.S.: Layered learning for concept synthesis. In: Peters and Skowron [**?**], pp. 187–208. doi:10.1007/978-3-540-27794-1_9
86. Nguyen, S.H., Nguyen, H.S.: Some efficient algorithms for rough set methods. In: Sixth International Conference on Information Processing and Management of Uncertainty on Knowledge Based Systems IPMU'1996, vol. III, pp. 1451–1456. Granada, Spain (1996)
87. Nguyen, T.T.: Eliciting domain knowledge in handwritten digit recognition. In: Proceedings of the 1st International Conference on Pattern Recognition and Machine Intelligence (PReMI 2005) December 18-22, 2005, Kolkata, India, pp. 762–767. doi:10.1007/11590316_123
88. Nguyen, T.T., Skowron, A.: Rough set approach to domain knowledge approximation. In: Proceedings of the 9-th International Conference on Rough Sets, Fuzzy Sets, Data Mining, and Granular Computing (RSFDGrC'2003), Chongqing, China, May 26-29, 2003, pp. 221–228 (2003). doi:10.1007/3-540-39205-X_28
89. Pal, S.K., Dasgupta, B., Mitra, P.: Rough self organizing map. Applied Intelligence **21**, 289–299 (2004). doi:10.1023/B:APIN.0000043561.99513.69
90. Pal, S.K., Mitra, P.: Case generation using rough sets with fuzzy representation. IEEE Trans. Knowledge and Data Engineering **16**(3), 292–300 (2004). doi:10.1109/TKDE.2003.1262181
91. Pal, S.K., Mitra, P.: Pattern Recognition Algorithms for Data Mining. CRC Press, Boca Raton, Florida (2004)
92. Pal, S.K., Pedrycz, W., Skowron, A., Swiniarski, R. (eds.): Preface, Special volume: Rough-neuro computing, *Neurocomputing*, vol. 36 (2001). doi:10.1016/S0925-2312(00)00332-5

93. Pal, S.K., Polkowski, L., Skowron, A. (eds.): Rough-Neural Computing: Techniques for Computing with Words. Cognitive Technologies. Springer-Verlag, Heidelberg (2004). doi:10.1007/978-3-642-18859-6
94. Pal, S.K., Skowron, A. (eds.): Rough Fuzzy Hybridization: A New Trend in Decision-Making. Springer-Verlag, Singapore (1999)
95. Pawlak, Z.: Rough Sets: Theoretical Aspects of Reasoning about Data, *System Theory, Knowledge Engineering and Problem Solving*, vol. 9. Kluwer Academic Publishers, Dordrecht, The Netherlands (1991). doi:10.1007/978-94-011-3534-4
96. Pawlak, Z., Polkowski, L., Skowron, A.: Rough sets and rough logic: A KDD perspective. In: Polkowski et al. [**?**], pp. 583–646. doi:10.1007/978-3-7908-1840-6_13
97. Pawlak, Z., Skowron, A.: Rough sets and boolean reasoning. Information Sciences **177**(1), 41–73 (2007). doi:10.1016/j.ins.2006.06.007
98. Pawlak, Z., Skowron, A.: Rough sets: Some extensions. Information Sciences **177**(1), 28–40 (2007). doi:10.1016/j.ins.2006.06.006
99. Pawlak, Z., Skowron, A.: Rudiments of rough sets. Information Sciences **177**(1), 3–27 (2007). doi:10.1016/j.ins.2006.06.003
100. Pedrycz, W., Skowron, S., Kreinovich, V. (eds.): Handbook of Granular Computing. John Wiley & Sons, Hoboken, NJ (2008). doi:10.1002/9780470724163
101. Peters, J.F., Skowron, A. (eds.): Transactions on Rough Sets I: Journal Subline, *Lecture Notes in Computer Science*, vol. 3100. Springer, Heidelberg (2004)
102. Peters, J.F., Suraj, Z., Shan, S., Ramanna, S., Pedrycz, W., Pizzi, N.J.: Classification of meteorological volumetric radar data using rough set methods. Pattern Recognition Letters **24**(6), 911–920 (2003). doi:10.1016/S0167-8655(02)00203-9
103. Polkowski, L., Lin, T.Y., Tsumoto, S. (eds.): Rough Set Methods and Applications: New Developments in Knowledge Discovery in Information Systems, *Studies in Fuzziness and Soft Computing*, vol. 56. Springer-Verlag/Physica-Verlag, Heidelberg (2000). doi:10.1007/978-3-7908-1840-6
104. Polkowski, L., Skowron, A. (eds.): Rough Sets in Knowledge Discovery 1: Methodology and Applications, *Studies in Fuzziness and Soft Computing*, vol. 18. Physica-Verlag, Heidelberg (1998)
105. Polkowski, L., Skowron, A. (eds.): Rough Sets in Knowledge Discovery 2: Applications, Case Studies and Software Systems, *Studies in Fuzziness and Soft Computing*, vol. 19. Physica-Verlag, Heidelberg (1998). doi:10.1007/978-3-7908-1883-3
106. Qian, Y., Wang, Q., Cheng, H., Liang, J., Dang, C.: Fuzzy-rough feature selection accelerator. Fuzzy Sets and Systems **258**, 61–78 (2015). doi:10.1016/j.fss.2014.04.029
107. Qian, Y.H., Liang, J., Pedrycz, W., Dang, C.Y.: An efficient accelerator for attribute reduction from incomplete data in rough set framework. Pattern Recognition **44**, 1658–1670 (2011). doi:10.1016/j.patcog.2011.02.020
108. Quafafou, M., Boussouf, M.: Generalized rough sets based feature selection. Intelligent Data Analysis **4**(1), 3–17 (2000)
109. R. S. Michalski: A theory and methodology of inductive learning. Artificial Intelligence **20**, 111–161 (1983). doi:10.1016/0004-3702(83)90016-4
110. Rissanen, J.: Modeling by shortes data description. Automatica **14**, 465–471 (1978). doi:10.1016/0005-1098(78)90005-5
111. Rissanen, J.: Minimum-description-length principle. In: S. Kotz, N. Johnson (eds.) Encyclopedia of Statistical Sciences, pp. 523–527. John Wiley & Sons, New York, NY (1985). doi:.1007/978-0-387-30164-8_540
112. Roy, A., K.Pal, S.: Fuzzy discretization of feature space for a rough set classifier. Pattern Recognition Letters **24**(6), 895–902 (2003). doi:10.1016/S0167-8655(02)00201-5
113. Saxena, A., Gavel, L.K., Shrivas, M.M.: Rough sets for feature selection and classification: An overview with applications. International Journal of Recent Technology and Engineering **3**, 62–69 (2014)
114. Sever, H., Raghavan, V.V., Johnsten, T.D.: The status of research on rough sets for knowledge discovery in databases. In: S. Sivasundaram (ed.) Proceedings of the Second Internationall Conference On Nonlinear Problems in Aviation and Aerospace (ICNPAA'1998),

April 29-May 1, 1998, Daytona Beach, FL, vol. 2, pp. 673–680. Embry-Riddle Aeronautical University, Daytona Beach, FL (1998)
115. Shan, N., Ziarko, W.: An incremental learning algorithm for constructing decision rules. In: W. Ziarko (ed.) Rough Sets, Fuzzy Sets and Knowledge Discovery, pp. 326–334. Springer Verlag, Berlin (1994). doi:10.1007/978-1-4471-3238-7_38
116. Skowron, A.: Approximate reasoning in distributed environments. In: Intelligent Technologies for Information Analysis, pp. 433–474. doi:10.1007/978-3-662-07952-2_18
117. Skowron, A.: Boolean reasoning for decision rules generation. In: J. Komorowski, Z.W. Raś (eds.) Proceedings of ISMIS'1993, Trondheim, Norway, June 15-18, 1993, *Lecture Notes in Artificial Intelligence*, vol. 689, pp. 295–305. Springer-Verlag (1993)
118. Skowron, A.: Synthesis of adaptive decision systems from experimental data. In: A. Aamodt, J. Komorowski (eds.) Fifth Scandinavian Conference on Artificial Intelligence SCAI'1995, *Frontiers in Artificial Intelligence and Applications*, vol. 28, pp. 220–238. IOS Press, Trondheim, Norway (1995)
119. Skowron, A.: Rough sets in KDD - plenary talk. In: Z. Shi, B. Faltings, M. Musen (eds.) 16-th World Computer Congress (IFIP'2000): Proceedings of Conference on Intelligent Information Processing (IIP'2000), pp. 1–14. Publishing House of Electronic Industry, Beijing (2000)
120. Skowron, A.: Rough sets and boolean reasoning. In: W. Pedrycz (ed.) Granular Computing: an Emerging Paradigm, *Studies in Fuzziness and Soft Computing*, vol. 70, pp. 95–124. Springer-Verlag/Physica-Verlag, Heidelberg (2001)
121. Skowron, A., Nguyen, H.S.: Boolean reasoning scheme with some applications in data mining. In: Proceedings of the 3-rd European Conference on Principles and Practice of Knowledge Discovery in Databases, *Lecture Notes in Computer Science*, vol. 1704, pp. 107–115. Springer Verlag, Berlin (1999). doi:10.1007/978-3-540-48247-5_12
122. Skowron, A., Pal, S.K.: Preface: Rough sets, pattern recognition and data mining (special volume). Pattern Recognition Letters **24(6)**, 829–831 (2003). doi:10.1016/S0167-8655(02)00195-2
123. Skowron, A., Pawlak, Z., Komorowski, J., Polkowski, L.: A rough set perspective on data and knowledge. In: W. Kloesgen, J. Żytkow (eds.) Handbook of KDD, pp. 134–149. Oxford University Press, Oxford (2002)
124. Skowron, A., Peters, J.: Rough sets: Trends and challenges. In: Proceedings of the 9-th International Conference on Rough Sets, Fuzzy Sets, Data Mining, and Granular Computing (RSFDGrC'2003), Chongqing, China, May 26-29, 2003, pp. 25–34 (2003). doi:10.1007/3-540-39205-X_4. (plenary talk)
125. Skowron, A., Rauszer, C.: The discernibility matrices and functions in information systems. In: R. Słowiński (ed.) Intelligent Decision Support - Handbook of Applications and Advances of the Rough Sets Theory, *System Theory, Knowledge Engineering and Problem Solving*, vol. 11, pp. 331–362. Kluwer Academic Publishers, Dordrecht, The Netherlands (1992). doi:10.1007/978-94-015-7975-9_21
126. Skowron, A., Stepaniuk, J.: Information granules and rough-neural computing. In: Pal et al. [**?**], pp. 43–84. doi:10.1007/978-3-642-18859-6
127. Skowron, A., Stepaniuk, J.: Ontological framework for approximation. In: Proceedings of the 10th International Conference on Rough Sets, Fuzzy Sets, Data Mining, and Granular Computing (RSFDGrC'2005), Regina, Canada,August 31-September 3, 2005, Part I, pp. 718–727 (2005). doi:10.1007/11548669_74
128. Skowron, A., Stepaniuk, J.: Approximation spaces in rough-granular computing. Fundamenta Informaticae **100**, 141–157 (2010). doi:10.3233/FI-2010-267
129. Skowron, A., Stepaniuk, J., Swiniarski, R.: Modeling rough granular computing based on approximation spaces. Information Sciences **184**, 20–43 (2012). doi:10.1016/j.ins.2011.08.001
130. Skowron, A., Suraj, Z. (eds.): Rough Sets and Intelligent Systems. Professor Zdzislaw Pawlak in Memoriam. Series Intelligent Systems Reference Library. Springer (2013). doi:10.1007/978-3-642-30344-9; 10.1007/978-3-642-30341-8
131. Ślezak, D.: Various approaches to reasoning with frequency-based decision reducts: A survey. In: Polkowski et al. [**?**], pp. 235–285. doi:10.1007/978-3-7908-1840-6

132. Ślezak, D.: Approximate entropy reducts. Fundamenta Informaticae **53**, 365–387 (2002)
133. Ślezak, D.: Association reducts: A framework for mining multi-attribute dependencies. In: M.S. Hacid, N.V. Murray, Z.W. Ras, S. Tsumoto (eds.) Foundations of Intelligent Systems, 15th International Symposium on Methodologies for Intelligent Systems (ISMIS 2005), Saratoga Springs, NY, May 25-28, 2005, *Lecture Notes in Artificial Intelligence*, vol. 3488, pp. 354–363. Springer, Heidelberg (2005). doi:10.1007/11425274_37
134. Stepaniuk, J.: Knowledge discovery by application of rough set models. In: Polkowski et al. [**?**], pp. 137–233. doi:10.1007/978-3-7908-1840-6_5
135. Sun, L., Xu, J., Tian, Y.: Feature selection using rough entropy-based uncertainty measures in incomplete decision systems. Journal of Computational Information Systems **36**, 206–216 (2012). doi:10.1016/j.knosys.2012.06.010
136. Swiniarski, R.: Rough sets and principal component analysis and their applications.Data model building and classification. In: Pal and Skowron [**?**], pp. 275–300
137. Swiniarski, R.: An application of rough sets and Haar wavelets to face recognition. In: Ziarko and Yao [**?**], pp. 561–568. doi:10.1007/3-540-45554-X_70
138. Swiniarski, R., Hargis, L.: A new halftoning method based on error diffusion with rough set filterin. In: Polkowski and Skowron [**?**], pp. 336–342. doi:10.1007/978-3-7908-1883-3
139. Swiniarski, R., Skowron, A.: Rough set methods in feature selection and extraction. Pattern Recognition Letters **24**(6), 833–849 (2003). doi:10.1016/S0167-8655(02)00196-4
140. Swiniarski, R.W., Skowron, A.: Independent component analysis, principal component analysis and rough sets in face recognition. In: Peters and Skowron [**?**], pp. 392–404. doi:10.1007/978-3-540-27794-1_19
141. Tsumoto, S., Tanaka, H.: PRIMEROSE: Probabilistic rule induction method based on rough sets and resampling methods. Computational Intelligence: An International Journal **11**, 389–405 (1995). doi:10.1111/j.1467-8640.1995.tb00040.x
142. Vapnik, V.: Statistical Learning Theory. John Wiley & Sons, New York, NY (1998)
143. Wang, C., He, Q., Chen, D., Hu, Q.: A novel method for attribute reduction of covering decision systems. Information Sciences **254**, 181–196 (2014). doi:10.1016/j.ins.2013.08.057
144. Wang, G.Y., Zhao, J., An, J.J.: A comparative study of algebra viewpoint and information viewpoint in attribute reduction. Fundamenta Informaticae **68**, 289–301 (2005)
145. Wang, J., Jia, C., Zhao, K.: Investigation on AQ11, ID3 and the principle of discernibility matrix. Journal of Computer Science and Technology **16**(1), 1–12 (2001). doi:10.1007/BF02948848
146. Wojna, A.: Analogy based reasoning in classifier construction. In: Transactions on Rough Sets IV: Journal Subline, pp. 277–374 (2005). doi:10.1007/11574798_11
147. Wróblewski, J.: Theoretical foundations of order-based genetic algorithms. Fundamenta Informaticae **28**, 423–430 (1996). doi:10.3233/FI-1996-283414
148. Wróblewski, J.: Analyzing relational databases using rough set based methods. In: Eighth International Conference on Processing and Management of Uncertainty in Knowledge-Based Systems IPMU, vol. I, pp. 256–262. Madrid, Spain (2000)
149. Wróblewski, J.: Adaptive aspects of combining approximation spaces. In: Pal et al. [**?**], pp. 139–156. doi:10.1007/978-3-642-18859-6
150. Yu, H., Wang, G., Lan, F.: Solving the attribute reduction problem with ant colony optimization. Transactions on Rough Sets XIII: Journal Subline **13**, 240–259 (2011). doi:10.1007/978-3-642-18302-7_13
151. Zhu, W.: Topological approaches to covering rough sets. Information Sciences **177**, 1499–1508 (2007). doi:10.1016/j.ins.2006.06.009
152. Ziarko, W., Yao, Y. (eds.): Proceedings of the 2nd International Conference on Rough Sets and Current Trends in Computing (RSCTC'2000), Banff, Canada, October 16-19, 2000, *Lecture Notes in Artificial Intelligence*, vol. 2005. Springer-Verlag, Heidelberg (2001). doi:10.1007/3-540-45554-X

Chapter 9
Deductive Logics from Rough Sets

A “logic of rough sets" would, in the natural sense, represent a formal system, statements in the language of which would be interpreted as rough sets in some approximation space. Thus “models" in the semantics of such a system would be approximation spaces, equipped with a meaning function that assigns rough sets to well-formed formulae (wffs) of the language.

Rough sets have been defined in more than one way for a Pawlak approximation space (X,R) – [**?**] lists five definitions, all of which are equivalent to each other. One of these is most commonly used:

(*) a rough set in (X,R), is the pair $(\underline{A},\overline{A})$, for each $A \subseteq X$,
where $\underline{A},\overline{A}$ denote the lower and upper approximations of A respectively. Another is a definition given by Pawlak in [**?**], and of interest to us in this paper:

(**) $A \subseteq X$ is a rough set in (X,R), provided the boundary of A, $BnA \neq \emptyset$.
For generality's sake, we could remove the restriction in (**) and consider definable sets (i.e. subsets with empty boundary) as special cases of rough sets.

Thus, in the semantics based on approximation spaces, the meaning function defining models, assigns to wffs either subsets of the domain, or pairs of subsets in accordance with (*) [**?**, **?**, **?**, **?**, **?**, **?**, **?**]. This is true even for semantics based on generalized approximation spaces, where different relations (may be more than one in number, with operations on them) are considered [**?**, **?**]. The logics invariably involve modalities to express the concepts of lower and upper approximations – some are simply known normal modal logics, or have non-Boolean connectives (and no modalities) in the language, but there are translations into modal logics. We make a study of this group of systems in Section 9.1. It may be remarked that the “rough logic" proposed by Pawlak [**?**] (the first system to be called so) makes an appearance here (cf. Section 9.1.6).

The “practical" source of Pawlak approximation spaces are *complete / deterministic* information systems. These have the form $\mathscr{S} \equiv (U,A,Val,f)$, where U is a set of objects, A a set of *attributes*, Val a set of *values* for the attributes, and f a function from $U \times A$ to Val. An equivalence relation $R_{\mathscr{S}}$ is induced on U (thus giving the approximation space $(U,R_{\mathscr{S}})$), as

$$x\, R_{\mathscr{S}}\, y \text{ in } U, \text{ if and only if } f(x,a) = f(y,a), \text{ for all } a \in A.$$

Th converse also holds: given any approximation space (U,R), one can define an information system $\mathscr{S} \equiv (U,A,Val,f)$ such that the induced equivalence $R_{\mathscr{S}}$ is just the relation R. So, in effect, a semantics based on approximation spaces induced by complete information systems, is identical to the one discussed above.

Generalized information systems, termed *incomplete/nondeterministic*, are those where f is a function from $U \times A$ to $\mathscr{P}(Val)$, and yields different kinds of binary relations (e.g. similarity, inclusion – cf. Section 9.2.1) apart from equivalences, on U. Thus any information system (complete or incomplete) on a domain U, induces a relational system or a (generalized) approximation space on U, i.e. the (non-empty) set U together with a set of binary relations. This is called a *standard structure* on U [**?**, **?**, **?**]. For example, for the complete information system (U,A,Val,f) above, $(U,R_{\mathscr{S}})$ is a standard structure on U. In Section 9.2.1, $(U,sim_{\mathscr{S}},in_{\mathscr{S}})$ is a standard structure for the incomplete information system $\mathscr{S} \equiv (U,A,Val,f)$, with similarity and inclusion relations $sim_{\mathscr{S}},in_{\mathscr{S}}$. (Different sets of relations can give different standard structures on the same set U.)

The induced relations in the standard structure may be characterized by a set of properties. As we know, equivalences are characterized by the properties of reflexivity, symmetry and transitivity. The similarity and inclusion relations considered in Section 9.2.1 are characterized by the properties $(S1),(S2),(S4)-(S6)$ given there. By a *general structure* on U [**?**, **?**, **?**], one means *any* relational system comprising a non-empty set, along with binary relations that satisfy the set of properties characterizing the induced relations in the standard structure. Again, for the complete information system (U,A,Val,f) above, any Pawlak approximation space (U,R) is a general structure. A general structure for $\mathscr{S}$ of Section 9.2.1, would be of the form (U,sim,in), where sim, in are binary relations on U satisfying $(S1),(S2),(S4)-(S6)$.

One finds logics with semantics defined on incomplete information systems, for instance, in [**?**], or with semantics defined on general structures [**?**]. However, Vakarelov [**?**, **?**, **?**, **?**] has established a series of characterization results, enabling an identification of semantics based on general and standard structures (as in case of the Pawlak approximation space and complete information system above). In case of [**?**] too, we demonstrate here that the logic in question is equivalent to a normal modal logic with certain generalized approximation spaces defining models. These systems are discussed in Section 9.2.

In another line, there are "logics of information systems", which accommodate in their language, expressions corresponding to objects and attributes [**?**, **?**, **?**, **?**]. Amongst these is a system that addresses the temporal aspect of information (cf. [**?**]), while [**?**] presents a logic for multiagent systems. There are also treatises on "rough relations" – a logic has been proposed [**?**] on the one hand, and on the other, we have the proposal of a logic programming language in "rough datalog" [**?**]. In Section 9.3, we briefly sketch these and other approaches, such as rough mereology [**?**]. It will be seen that, some of the logics [**?**, **?**, **?**] have atomic propositions as (or built from) *descriptors*, the key feature of *decision logic* [**?**]. Decision logic is well-known, and not presented in this article.

One should mention that a few of the logics described here, have also been used as a base to express various concepts involving rough sets. For instance, Yao and Lin [**?**] have defined graded and probabilistic rough sets, using graded and probabilistic modal operators in the language of normal modal systems. Common and distributed knowledge operators have been interpreted in generalized approximation spaces by Wong [**?**]. In [**?**], another modal system (inspired by [**?**]) has been used to propose postulates for rough belief change.

A comparative study of the presented logics is made in Section 9.4. The paper concludes by indicating possible future directions of investigation in Section 9.5.

9.1 Logics with semantics based on approximation spaces

In this section, we take a look at logics with approximation spaces defining models. We find six kinds of systems.

For a logic $\mathscr{L}$, "α is a theorem of $\mathscr{L}$" shall be indicated by the notation $\vdash_{\mathscr{L}} \alpha$.

9.1.1 Normal modal systems

The modal nature of the lower and upper approximations of rough sets was evident from the start. Hence, it is no surprise that normal modal systems were focussed upon, during investigations on logics for rough sets. In particular, in case of Pawlak rough sets, the two approximations considered as operators clearly obey all the $S5$ laws. The formal connection between the syntax of $S5$ and its semantics in terms of rough sets is given as follows.

According to the Kripke semantics for $S5$, a wff α is interpreted by a function π as a subset in a non-empty domain U, the subset representing the extension of the formula – i.e. the collection of situations/objects/worlds where the wff holds. Moreover, in an $S5$-model $\mathscr{M} \equiv (U,R,v)$ (say), the accessibility relation R is an equivalence on U. Further, if $\Box$, $\Diamond$ denote the necessity and possibility operators respectively then for any wff α, $v(\Box\alpha) = \underline{v(\alpha)}$ and $v(\Diamond\alpha) = \overline{v(\alpha)}$.

A wff α is *true* in $\mathscr{M}$, if $v(\alpha) = U$. Now it can easily be seen that all the $S5$ theorems involving $\Box$ and $\Diamond$ translate into valid properties of lower and upper approximations.

Taking a cue from this connection, similar links have been pointed out (e.g. in [**?**, **?**]) between "rough sets" on generalized approximation spaces, and different normal modal systems. The basic idea is to define generalized approximation operators corresponding to any binary relation R on the domain U – this has been done by many (e.g. for tolerance relations in [**?**] and others – cf. [**?**]). More explicitly, a map $r : U \to \mathscr{P}(U)$ is defined as $r(x) \equiv \{y \in U : xRy\}$. Then the operators $\underline{apr}, \overline{apr} : \mathscr{P}(U) \to \mathscr{P}(U)$ are given by

$\underline{apr}(A) \equiv \{x : r(x) \subseteq A\}$, and $\overline{apr}(A) \equiv \{x : r(x) \cap A \neq \emptyset\}$.

The rough set operators then satisfy various properties, depending upon the nature of R. Now let $\mathscr{L}$ denote a normal modal language, and $\mathscr{M} \equiv (U, R, v)$ be a model for $\mathscr{L}$. v, as before, interprets a wff as a subset in U. Then it is straightforward to observe that for any wff α of $\mathscr{L}$,

$v(\Box\alpha) = \underline{apr}(v(\alpha))$, and dually, $v(\Diamond\alpha) = \overline{apr}(v(\alpha))$.

By the above interpretation, the modal logics like $KB, KT, K4, S5$ etc. could be said to capture the properties of rough sets in generalized approximation spaces based on different R (symmetric, reflexive, transitive, equivalence etc.).

As remarked in the Introduction, this link has been made use of further. Considering graded and probabilistic modal operators on the above systems, *graded* and *probabilistic* rough sets have been defined in [**?**]. Wong [**?**] has interpreted common and distributed knowledge operators (as defined in logic of knowledge) in generalized approximation spaces with an indexed set of indiscernibility relations (corresponding to the knowledge operator of each agent).

9.1.2 DAL

[**?**] considers generalized approximation spaces containing a family of equivalence relations instead of just one. The logic *DAL* that is defined in [**?**], has models based on these spaces. Further, the set of equivalence relations is assumed to be closed with respect to the operations of intersection and transitive closure of union of relations.

The language of *DAL*, expectedly, includes a family of modal operators intended to correspond to the indiscernibility relations on the domains of the models. Formally, this is done by having a set $\mathscr{R}$ (say) of *relational variables* apart from the set $\mathscr{P}$ of propositional ones. There are binary operations $\cap, \uplus$, and a collection REL of *relational expressions* is built inductively out of the members of $\mathscr{R}$ with these operations. Apart from the classical Boolean connectives, a modal connective $[R]$ is then introduced in the language for each $R \in REL$.

A *DAL*-model is a structure $\mathscr{U} \equiv (U, \{\rho_R\}_{R \in REL}, m)$, where, (i) for any $R \in REL, \rho_R$ is an equivalence relation in the set U; (ii) $\rho_{R\cap S}$ is the greatest equivalence relation in U included in both ρ_R and ρ_S; (iii) $\rho_{R\uplus S}$ is the least equivalence relation including both ρ_R and ρ_S; and (iv) m is the meaning function from $\mathscr{P} \cup \mathscr{R}$ to $\mathscr{P}(U) \cup \{\rho_R\}_{R\in REL}$ such that $m(p) \subseteq U$, for $p \in \mathscr{P}$, and $m(R) \equiv \rho_R$, for $R \in REL$.

For evaluating truth of wffs in *DAL*-models, one defines a function v that is determined by the meaning function m:

$v(p) \equiv m(p)$, for $p \in \mathscr{P}$,

$v([R]\alpha) \equiv \{x \in U : y \in v(\alpha),$ for all y such that $x\ m(R)\ y\}$,

the Boolean cases being defined in the standard way.

Definitions of truth and validity then are as usual: α is true in $\mathscr{U}$, provided $v(\alpha) = U$, and valid if it is true in all *DAL*-models.

DAL has been axiomatized as follows. The connective $\langle\rangle$ is the dual of $[]$.

$A1$. All classical tautologies,
$A2$. $[R](\alpha \to \beta) \to ([R]\alpha \to [R]\beta)$,
$A3$. $[R]\alpha \to \alpha$,
$A5$. $\langle R\rangle\alpha \to [R]\langle R\rangle\alpha$,
$A5$. $[R \uplus S]\alpha \to [R]\alpha \wedge [S]\alpha$,
$A6$. $(([P]\alpha \to [R]\alpha) \wedge ([P]\alpha \to [S]\alpha)) \to ([P]\alpha \to [R \uplus S]\alpha)$,
$A7$. $[R]\alpha \vee [S]\alpha \to [R \wedge S]\alpha$,
$A8$. $(([R]\alpha \to [P]\alpha) \wedge ([S]\alpha \to [P]\alpha)) \to ([R \wedge S]\alpha \to [P]\alpha)$.

The only rules of inference are Modus Ponens and Necessitation (corresponding to the connective $[R]$ for each $R \in REL$).

The axiomatization yields a completeness result with respect to the afore-mentioned semantics.

Theorem 9.1. *For any DAL-wff α, $\vdash_{DAL} \alpha$, if and only if α is valid.*

9.1.3 Pre-rough logic

Following in the footsteps of Rasiowa, the algebra of rough sets was investigated in [**?**] in order to arrive at a logic for the theory. An algebraic structure called *pre-rough algebra* was proposed – this is a *quasi Boolean algebra* [**?**] along with a topological operator satisfying all the properties of an *interior*, and more. A corresponding logic *PRL* was framed, and observed to be sound and complete with respect to a semantics based on rough sets.

The language of *PRL* has the primitive logical symbols $\neg, \sqcap, \Box$. $\sqcup, \Diamond$ are duals of $\sqcap, \Box$, while $\Rightarrow$ is defined as:

$$\alpha \Rightarrow \beta \equiv (\neg\Box\alpha \sqcup \Box\beta) \sqcap (\neg\Diamond\alpha \sqcup \Diamond\beta),$$

for any wffs α, β of *PRL*.

As in the case of $S5$, a model for *PRL* is of the form $\mathscr{M} \equiv (U, R, v)$, where the departure from the $S5$-semantics lies in the definition of the meaning function v with respect to the connectives of conjunction $\sqcap$ and implication $\Rightarrow$. For any α, β in *PRL*, $S, T \subseteq U$,

$v(\alpha \sqcap \beta) \;\equiv\; v(\alpha) \sqcap v(\beta)$, and
$v(\alpha \Rightarrow \beta) \;\equiv\; v((\neg\Box\alpha \sqcup \Box\beta) \sqcap (\neg\Diamond\alpha \sqcup \Diamond\beta))$, where

$S \sqcap T \equiv (S \cap T) \cup (S \cap \overline{T} \cap (\overline{S \cap T})^c)$ (^{c} denoting complementation).

Definition of truth of a wff α in $\mathscr{M}$ remains the same: this is if and only if $v(\alpha) = U$. It may then be noticed that $\Rightarrow$ reflects *rough inclusion*: a wff $\alpha \Rightarrow \beta$ is true in (U, R, v) provided $v(\alpha)$ is roughly included in $v(\beta)$. Further, $\sqcap$ / $\sqcup$ are operations that reduce to ordinary set intersection / union only when working on definable sets.

α is *valid* (written $\models_{RS} \alpha$), if and only if α is true in every *PRL*-model.

Following are the axiom schemes for *PRL*:

1. $\alpha \Rightarrow \alpha$	
$2a$. $\neg\neg\alpha \Rightarrow \alpha$	$2b$. $\alpha \Rightarrow \neg\neg\alpha$
3. $\alpha \sqcap \beta \Rightarrow \alpha$	4. $\alpha \sqcap \beta \Rightarrow \beta \sqcap \alpha$
$5a$. $\alpha \sqcap (\beta \sqcup \gamma) \Rightarrow (\alpha \sqcap \beta) \sqcup (\alpha \sqcap \gamma)$	$5b$. $(\alpha \sqcap \beta) \sqcup (\alpha \sqcap \gamma) \Rightarrow \alpha \sqcap (\beta \sqcup \gamma)$
6. $\Box\alpha \Rightarrow \alpha$	
$7a$. $\Box(\alpha \sqcap \beta) \Rightarrow \Box(\alpha) \sqcap \Box(\beta)$	$7b$. $\Box(\alpha) \sqcap \Box(\beta) \Rightarrow \Box(\alpha \sqcap \beta)$
8. $\Box\alpha \Rightarrow \Box\Box\alpha$	9. $\Diamond\Box\alpha \Rightarrow \Box\alpha$
$10a$. $\Box(\alpha \sqcup \beta) \;\Rightarrow\; \Box\alpha \sqcup \Box\beta$	$10b$. $\Box\alpha \sqcup \Box\beta \;\Rightarrow\; \Box(\alpha \sqcup \beta)$

Rules of inference :

1. $$\frac{\alpha \quad \alpha \Rightarrow \beta}{\beta}$$ *modus ponens*

2. $$\frac{\alpha \Rightarrow \beta \quad \beta \Rightarrow \gamma}{\alpha \Rightarrow \gamma}$$ *hypothetical syllogism*

3. $$\frac{\alpha}{\beta \Rightarrow \alpha}$$

4. $$\frac{\alpha \Rightarrow \beta}{\neg\beta \Rightarrow \neg\alpha}$$

5. $$\frac{\alpha \Rightarrow \beta \quad \alpha \Rightarrow \gamma}{\alpha \Rightarrow \beta \sqcap \gamma}$$

6. $$\frac{\alpha \Rightarrow \beta, \beta \Rightarrow \alpha \quad \gamma \Rightarrow \delta, \delta \Rightarrow \gamma}{(\alpha \Rightarrow \gamma) \Rightarrow (\beta \Rightarrow \delta)}$$

7. $$\frac{\alpha \Rightarrow \beta}{\Box\alpha \Rightarrow \Box\beta}$$

8. $$\frac{\alpha}{\Box\alpha}$$

9. $$\frac{\Box\alpha \Rightarrow \Box\beta \quad \Diamond\alpha \Rightarrow \Diamond\beta}{\alpha \Rightarrow \beta}$$

One can then prove, for any *PRL*-wff α,

Theorem 9.2. *$\vdash_{PRL} \alpha$, if and only if $\models_{RS} \alpha$.*

We shall meet this logic and its semantics again in the coming sections.

9.1.4 3-valued Łukasiewicz logic $\mathscr{L}3$

The connection of rough sets with 3-valuedness, also came up in the context of algebraic investigations. For example, in [**?**, **?**, **?**], an equivalence of 3-valued Łukasiewicz (Moisil) algebras with rough set structures was observed. In terms of logic, the way we can set up a formal link between the intensely studied $\mathscr{L}3$ and a rough set semantics – in fact, the semantics just outlined in Section 9.1.3, is as follows.

Let us recall Wajsberg's axiomatization of $\mathscr{L}3$ (cf. [**?**]). The logical symbols $\neg, \rightarrow$ are taken to be primitive.

Axiom schemes:

1. $\alpha \rightarrow (\beta \rightarrow \alpha)$.
2. $(\alpha \rightarrow \beta) \rightarrow ((\beta \rightarrow \gamma) \rightarrow (\alpha \rightarrow \gamma))$.
3. $((\alpha \rightarrow \neg\alpha) \rightarrow \alpha) \rightarrow \alpha$.
4. $(\neg\alpha \rightarrow \neg\beta) \rightarrow (\beta \rightarrow \alpha)$.

The only rule of inference is Modus Ponens.

$\mathscr{L}3$ is known to be sound and complete with respect to the class of 3-valued Łukasiewicz (Moisil) algebras, as well as with respect to the semantics on $\mathbf{3} \equiv \{0, 1/2, 1\}$, with Łukasiewicz negation and implication [**?**].

Now a logic L_1 is said to be *embeddable* into a logic L_2, provided there is a translation * of wffs of L_1 into L_2, such that $\vdash_{L_1} \alpha$ if and only if $\vdash_{L_2} \alpha^\star$ for any wff α of L_1. We use the denotation $L_1 \rightharpoonup L_2$. $L_1 \rightleftharpoons L_2$ denotes existence of embeddings both ways.

[**?**] establishes the following. There are translations $^\circ$ from $\mathscr{L}3$ into *PRL* and * from *PRL* into $\mathscr{L}3$ given by

$(\neg\alpha)^\circ \equiv \neg\alpha^\circ$,
$(\alpha \rightarrow \beta)^\circ \equiv (\Diamond\neg\alpha^\circ \sqcup \beta^\circ) \sqcap (\Diamond\beta^\circ \sqcup \neg\alpha^\circ)$;
$(\neg\alpha)^* \equiv \neg\alpha^*$,
$(\alpha \sqcup \beta)^* \equiv (\alpha^* \rightarrow \beta^*) \rightarrow \beta^*$,
$(\alpha \sqcap \beta)^* \equiv \neg(\neg\alpha^* \sqcup \neg\beta^*)$,
$(\Diamond\alpha)^* \equiv \neg\alpha^* \rightarrow \alpha^*$.

(One may notice that for any α, $(\alpha^\circ)^*$ and $(\alpha^*)^\circ$ are logically equivalent to α in the respective systems.)

It is then shown that $\mathscr{L}3 \rightleftharpoons PRL$. Thus

Theorem 9.3.
(a) $\vdash_{\mathscr{L}3} \alpha$*, if and only if* $\models_{RS} \alpha^\circ$*, for an* $\mathscr{L}3$*-wff* α *and*
(b) $\vdash_{\mathscr{L}3} \alpha^*$*, if and only if* $\models_{RS} \alpha$*, for a PRL-wff* α.

9.1.5 Logic for regular double Stone algebras

Another line of algebraic investigation has resulted in linking rough set structures with the class of regular double Stone algebras [**?**]. A *double Stone algebra* (DSA) is a structure $(L, \sqcup, \sqcap, ^*, ^+, 0, 1)$ such that

$(L, \sqcup, \sqcap, 0, 1)$ is a bounded distributive lattice,
$y \leq x^*$ if and only if $y \sqcap x = 0$,
$y \geq x^+$ if and only if $y \sqcup x = 1$ and
$x^* \sqcup x^{**} = 1$, $x^+ \sqcap x^{++} = 0$.

The operations $^*, ^+$, as evident, are two kinds of complementation on the domain.

The DSA is *regular* if, in addition to the above, for all $x \in L$,

$$x \sqcap x^+ \leq x \sqcup x^*$$

holds. This is equivalent to requiring that $x^* = y^*,\ x^+ = y^+$ imply $x = y$, for all $x, y \in L$.

Considering the definition (*) of rough sets (cf. Introduction), one finds that the collection $\mathscr{RS}$ of rough sets $(\underline{X}, \overline{X})$ over an approximation space (U,R) can be made into a regular DSA. The zero of the structure is the element $(\emptyset, \emptyset)$, while the unit is (U,U). The operations $\sqcup, \sqcap, ^*, ^+$ are defined as

$$(\underline{X}, \overline{X}) \sqcup (\underline{Y}, \overline{Y}) \equiv (\underline{X} \cup \underline{Y}, \overline{X} \cup \overline{Y}),$$
$$(\underline{X}, \overline{X}) \sqcap (\underline{Y}, \overline{Y}) \equiv (\underline{X} \cap \underline{Y}, \overline{X} \cap \overline{Y}),$$
$$(\underline{X}, \overline{X})^* \equiv (\overline{X}^c, \overline{X}^c),$$
$$(\underline{X}, \overline{X})^+ \equiv (\underline{X}^c, \underline{X}^c).$$

For the converse, Comer shows that any regular DSA is isomorphic to a subalgebra of $\mathscr{RS}$ for some approximation space (U,R).

Using these facts, a logic $\mathscr{L}_\mathscr{D}$ for rough sets is defined by Düntsch [**?**] as follows.

The language of $\mathscr{L}_\mathscr{D}$ has two unary connectives $^*, ^+$ (for two kinds of negation), apart from the binary connectives $\vee, \wedge$ and constant symbol $\top$. We write α^*, α^+ instead of $^*\alpha, ^+\alpha$, just to keep parity with the algebraic notation used above.

A model of $\mathscr{L}_\mathscr{D}$ is a pair (W, v), where W is a (non-empty) set and v is the meaning function assigning to propositional variables, pairs in $\mathscr{P}(W) \times \mathscr{P}(W)$ such that if $v(p) = (A,B)$ then $A \subseteq B$. $v(p) = (A,B)$ is to express that "p holds at all states of A and does not hold at any state outside B". For $\top$, we have $v(\top) \equiv (W,W)$.

v is extended to the set of all wffs recursively:

if $v(\alpha) = (A,B)$ and $v(\beta) = (C,D)$ then

$$v(\alpha \vee \beta) \equiv (A \cup C, B \cup D),$$
$$v(\alpha \wedge \beta) \equiv (A \cap C, B \cap D),$$
$$v(\alpha^*) \equiv (B^c, B^c),$$
$$v(\alpha^+) \equiv (A^c, A^c).$$

A wff α is true in a model (W, v), provided $v(\alpha) = (W,W)$.

We would now like to make explicit, how v interprets the wffs of $\mathscr{L}_\mathscr{D}$ as rough sets over some approximation space. One refers to [**?**], and [**?**].

Consider the range $ran(v)$ of the map v in $\mathscr{P}(W) \times \mathscr{P}(W)$. It can be shown that it forms a regular DSA through the operations $\sqcup, \sqcap, ^*, ^+$:

$$v(\alpha) \sqcup v(\beta) \equiv v(\alpha \vee \beta),$$
$$v(\alpha) \sqcap v(\beta) \equiv v(\alpha \wedge \beta),$$
$$v(\alpha)^* \equiv v(\alpha^*),$$
$$v(\alpha)^+ \equiv v(\alpha^+).$$

$v(\top^*)$ (or $v(\top^+)$) is the zero ($(\emptyset, \emptyset)$) of the algebra, while $v(\top) = (W,W)$ is the unit.

In fact, the variety of regular DSA's is just the one generated by regular DSA's of the kind $ran(v)$, where v ranges over all meaning functions for all models.

Using the correspondence between classes of algebras and logic [**?**], [**?**] concludes, amongst other properties of $\mathscr{L}_\mathscr{D}$, that

Theorem 9.4. *$\mathscr{L}_\mathscr{D}$ has a finitely complete and strongly sound Hilbert style axiom system.*

Through Comer's representation result, $ran(v)$ corresponding to any model (W,v) of $\mathscr{L}_{\mathscr{D}}$, is isomorphic to a subcollection of $\mathscr{RS}$ for some approximation space (U,R). We can now say that $v(\alpha)$ for a wff α, can be identified with a rough set over some (U,R) in precisely the following manner.

Let U consist of all the *join irreducible* elements of $ran(v)$, i.e. $v(\alpha) \in U$, if and only if $v(\alpha) \neq (\emptyset,\emptyset)$, and for all wffs β,γ, if $v(\alpha) = v(\beta) \sqcup v(\gamma)$ then either $v(\alpha) = v(\beta)$ or $v(\alpha) = v(\gamma)$. An equivalence relation R on U can then be obtained, where R is given by:

$$v(\alpha)\ R\ v(\beta) \text{ if and only if } v(\alpha^{**}) = v(\beta^{**}),$$

i.e. if and only if $B = D$, where $v(\alpha) = (A,B)$ and $v(\beta) = (C,D)$.

Now define $f : ran(v) \rightarrow \mathscr{P}(U)$ such that for $v(\alpha) = (A,B)$,

$$f(A,B) \equiv \{v(\beta) = (C,D) \in U : C \subseteq A, D \subseteq B\}.$$

Finally, define the map $g : ran(v) \rightarrow \mathscr{P}(U) \times \mathscr{P}(U)$ as:

$$g(A,B) \equiv (f(A,A), f(B,B)), \text{ where } v(\alpha) = (A,B).$$

(Note that $(A,A),(B,B) \in U$, as $v(\alpha^{++}) = (A,A)$, and $v(\alpha^{**}) = (B,B)$.)

It can then be shown that (a) g is injective, and (b) g preserves $\sqcup, \sqcap, ^*, ^+$.

Moreover, if $v(\alpha) = (A,B)$,

$$g(v(\alpha)) = (\ \underline{f(A,B)}, \overline{f(A,B)}\ ),$$

a rough set in the approximation space (U,R).

[**?**] does not present an explicit proof method for the logic $\mathscr{L}_{\mathscr{D}}$ – the only comment on the matter is vide Theorem 9.4. Recently, Dai [**?**] has presented a sequent calculus for a logic (denoted *RDSL*) with a semantics based on the regular DSAs formed by collections of rough sets of the kind $\mathscr{RS}$ over some approximation space (U,R) (defined earlier in the section). The language of *RDSL* is the same as that of $\mathscr{L}_{\mathscr{D}}$, except that the constant symbol $\bot$ (dual for $\top$) is included amongst the primitive symbols. Models are of the form $(\mathscr{RS}, v)$, where v, the meaning function, is a map from the set of propositional variables to $\mathscr{RS}$. Thus $v(p)$, for a propositional variable p, is a pair $(\underline{X}, \overline{X})$ in the approximation space (U,R). v is extended to the set of all wffs in the same way as for models of $\mathscr{L}_{\mathscr{D}}$.

We note that an *RDSL*-model $(\mathscr{RS}, v)$ may be identified with the $\mathscr{L}_{\mathscr{D}}$-model (U,v). On the other hand, due to Comer's representation result, given any $\mathscr{L}_{\mathscr{D}}$-model (W,v), there is an isomorphism f from $ran(v)$ to a subalgebra ($\mathscr{S}$, say) of $\mathscr{RS}$ on some approximation space. One can thus find an *RDSL*-model $(\mathscr{RS}, v')$ such that $ran(v')$ is $\mathscr{S}$, i.e. $v'(p) \equiv f(v(p))$, for every propositional variable p. So, in this sense, the classes of models of the two logics are identifiable.

As in classical sequent calculus, for finite sequences of wffs $\Gamma \equiv (p_1, p_2, \ldots p_m)$ and $\Delta \equiv (q_1, q_2, \ldots q_n)$ in *RDSL*, the sequent $\Gamma \Rightarrow \Delta$ is said to be valid in a model $(\mathscr{RS}, v)$ if and only if

$$v(p_1) \sqcap \ldots \sqcap v(p_m) \leq v(q_1) \sqcup \ldots \sqcup v(q_n).$$

$\sqcup, \sqcap$ are the operations in the regular DSA $(\mathscr{RS}, \sqcup, \sqcap, ^*, ^+, <\emptyset,\emptyset>, <U,U>)$.

$\Gamma \Rightarrow \Delta$ is said to be valid (in notation, $\models_{RDSA} \Gamma \Rightarrow \Delta$) if and only if $\Gamma \Rightarrow \Delta$ is valid in every *RDSL*-model.

The standard classical axiom $p \Rightarrow p$ and rules for the connectives $\wedge, \vee$ and constant symbols $\top, \bot$ are considered to define derivability ($\vdash_{RDSL}$). In addition, the axioms and rules for the two negations $^*, ^+$ are as follows.

1. $p \Rightarrow p^{**}$.
2. $p^* \Rightarrow p^{***}$.
3. $p \Rightarrow p^{++}$.
4. $p^+ \Rightarrow p^{+++}$.

$$(R^*)\ \frac{\Gamma \Rightarrow \Delta}{\Delta^* \Rightarrow \Gamma^*} \quad (R^+)\ \frac{\Gamma \Rightarrow \Delta}{\Delta^+ \Rightarrow \Gamma^+}$$

Soundness and completeness are then proved, with respect to the semantics sketched.

Theorem 9.5. $\vdash_{RDSL} \Gamma \Rightarrow \Delta$*, if and only if* $\models_{RDSA} \Gamma \Rightarrow \Delta$.

9.1.6 Logic for rough truth or of rough consequence

In [**?**], a logic R_l (the first in literature to be called "rough logic") was proposed, along with a very appealing notion of *rough truth*. The language of R_l consists of the standard Boolean connectives, and models $\mathscr{M} \equiv (U, R, v)$ are based on approximation spaces. v assigns subsets of the domain U to wffs in the usual manner. Five logical values of "truth", "falsity", "rough truth", "rough falsity" and "rough inconsistency" are considered in this work, with truth and falsity representing the limit of our partial knowledge.

As we know, a wff α is *true* in $\mathscr{M}$, if $v(\alpha) = U$. α is said to be *surely/possibly* true on $x \in U$, if $x \in \underline{v(\alpha)}$ $(\overline{v(\alpha)})$ respectively. α is *roughly true* in $\mathscr{M}$, if it is possibly true on every x in U, i.e. $\overline{v(\alpha)} = U$, or in other words, $v(\alpha)$ is *externally indiscernible* [**?**] in (U, R). On the other hand, α is *roughly false*, when $\underline{v(\alpha)} = \emptyset$ ($v(\alpha)$ is *internally* indiscernible), and α is *roughly inconsistent*, if it is both roughly true and false ($v(\alpha)$ is *totally* indiscernible).

Let us consider the modal system $S5$. Note that models of $S5$ and R_l are identical. We can then effect a translation of the above concepts into $S5$. In (U, R, v), a wff α can be termed roughly true if $\overline{v(\alpha)} = v(\Diamond\alpha) = U$, roughly false if $\underline{v(\alpha)} = v(\Box\alpha) = \emptyset$, and roughly inconsistent if both hold.

In [**?**], a logic L_r having the same models as above was proposed, with the speciality that the syntax-semantics relationships are explored with *rough truth* replacing truth and *rough validity* replacing validity. The notion of consistency is replaced by one of *rough consistency* too. The consequence relation defining the logic is also non-standard. These ideas were first mooted in [**?**], and L_r is a modified version of the formal system discussed there.

L_r has a normal modal language. A model $\mathscr{M} \equiv (U, R, v)$ is a *rough model* of Γ, if and only if for every $\gamma \in \Gamma$, $v(\Diamond\gamma) = U$, i.e. γ is roughly true in $\mathscr{M}$. α is a *rough* *semantic consequence* of Γ (denoted $\Gamma|\!\approx\!\alpha$) if and only if every rough model of Γ is a rough model of α. If Γ is empty, α is said to be *roughly valid*, written $|\!\approx\!\alpha$.

There are two rules of inference:

$$R_1.\quad \frac{\alpha}{\beta} \quad \text{if } \vdash_{S5} \Diamond\alpha \to \Diamond\beta \qquad R_2.\quad \frac{\Diamond\alpha \quad \Diamond\beta}{\Diamond\alpha \wedge \Diamond\beta}$$

The consequence relation is defined as follows. Let Γ be any set of wffs and α any wff in L_r.

α is a *rough consequence* of Γ (denoted $\Gamma|\!\sim\!\alpha$) if and only if there is a sequence $\alpha_1,...,\alpha_n$ $(\equiv \alpha)$ such that each α_i $(i = 1,...,n)$ is either (i) a theorem of $S5$, or (ii) a member of Γ, or (iii) derived from some of $\alpha_1,...,\alpha_{i-1}$ by R_1 or R_2.
If Γ is empty, α is said to be a *rough theorem*, written $|\!\sim\!\alpha$.

A kind of "rough Modus Ponens" is then derivable, in the form: if $\Gamma|\!\sim\!\alpha$, $\vdash_{S5} \alpha' \to \beta$ with $\vdash_{S5} \alpha \approx \alpha'$ then β. Here $\approx$ reflects the notion of "rough equality", $\alpha \approx \beta \equiv (\Box\alpha \leftrightarrow \Box\beta) \wedge (\Diamond\alpha \leftrightarrow \Diamond\beta)$. One also obtains soundness of L_r with respect to the above semantics: if $\Gamma|\!\sim\!\alpha$ then $\Gamma|\!\approx\!\alpha$.

It is clear that in the face of an incomplete description of a concept p, p and "not" p (in the classical sense) may not always represent conflicting situations. To accommodate this possibility, a set Γ of wffs is termed *roughly consistent* if and only if the set $\Diamond\Gamma \equiv \{\Diamond\gamma : \gamma \in \Gamma\}$ is $S5$-consistent.

With the help of this notion, one obtains

Theorem 9.6. *(Completeness)*
(a) Γ is roughly consistent if and only if it has a rough model.
(b) For any L_r-wff α, if $\Gamma|\!\approx\!\alpha$ then $\Gamma|\!\sim\!\alpha$.

Thus, L_r appears as another system that is able to address rough sets and related notions. We shall remark on its relationship with other well-known systems in Section 9.4. It may be mentioned that L_r has been used as the base logic for a proposal of *rough belief change* in [**?**].

9.2 Logics with semantics based on information systems

We now present logics, the models of which are defined on approximation spaces induced by information systems. We find one pioneering system *NIL* that has inspired the proposal of many others in the same line. The section also includes a logic by Nakamura, the models of which are directly defined on information systems.

9.2.1 NIL

Recall that an incomplete information system is of the form $\mathscr{S} \equiv (U, A, Val, f)$, where U is a set of objects, A a set of attributes, Val a set of values for the attributes, and f a function from $U \times A$ to $\mathscr{P}(Val)$.

The logic *NIL* proposed by Orłowska and Pawlak [**?**] works on incomplete information systems, in which the function f satisfies an additional condition:

($\diamond$) $f(x,a) \neq \emptyset$, for all $x \in U,\ a \in A$.

One observes that, given $\mathscr{S} \equiv (U, A, Val, f)$, two particular kinds of binary relations on the domain U are induced – these dictate the formulation of *NIL*. Let $x, y \in U$.

Similarity ($sim_{\mathscr{S}}$): $x\ sim_{\mathscr{S}}\ y$ if and only if $f(x,a) \cap f(y,a) \neq \emptyset$, for all $a \in A$.
Inclusion ($in_{\mathscr{S}}$): $x\ in_{\mathscr{S}}\ y$ if and only if $f(x,a) \subseteq f(y,a)$, for all $a \in A$.

It can be shown that for every incomplete information system $\mathscr{S} \equiv (U, A, Val, f)$ and $x, y, z \in U$, the following hold.

$(S1)$ $x\ in_{\mathscr{S}}\ x$.
$(S2)$ if $x\ in_{\mathscr{S}}\ y$ and $y\ in_{\mathscr{S}}\ z$ then $x\ in_{\mathscr{S}}\ z$.
$(S3)$ if $x\ sim_{\mathscr{S}}\ y$ for some y, then $x\ sim_{\mathscr{S}}\ x$.
$(S4)$ if $x\ sim_{\mathscr{S}}\ y$ then $y\ sim_{\mathscr{S}}\ x$.
$(S5)$ if $x\ sim_{\mathscr{S}}\ y$, $x\ in_{\mathscr{S}}\ u$, $y\ in_{\mathscr{S}}\ v$ then $u\ sim_{\mathscr{S}}\ v$.

Further, if the condition ($\diamond$) is satisfied by f then *sim* satisfies

$(S6)$ $x\ sim_{\mathscr{S}}\ x$.

Thus a *standard structure* (cf. Introduction) corresponding to an incomplete information system $\mathscr{S} \equiv (U, A, Val, f)$ with condition ($\diamond$), would be $(U, sim_{\mathscr{S}}, in_{\mathscr{S}})$. On the other hand, a *general structure* for $\mathscr{S}$ would be of the form (U, sim, in), where *sim*, *in* are binary relations on U satisfying $(S1),(S2),(S4)-(S6)$. For brevity, we refer to these as standard and general *NIL*-structures respectively.

NIL could be termed as a modal version of decision logic introduced by Pawlak [**?**], an association similar to that of rough logic [**?**] and $S5$ (cf. Section 9.1.6). The atomic propositions of *NIL* are the descriptors of decision logic – of the form (a, v), where a is an "attribute constant", and v a constant representing "value of attribute".

Apart from the standard Boolean connectives $\neg, \vee$, the language contains modal connectives $\Box, \Box_1, \Box_2$ corresponding to *sim*, *in* and the *inverse* in^{-1} of *in* respectively. Wffs are built, as usual, out of the atomic propositions (descriptors) and the connectives. Note that there are no operations on the attribute or value constants.

A *NIL*-model $\mathscr{M} \equiv (U, sim, in, m)$ consists of a general structure (U, sim, in) as above, along with a meaning function m from the set of all descriptors to the set $\mathscr{P}(U)$.

m is extended recursively to the set of all *NIL*-wffs in the usual manner. In particular,

$$m(\Box\alpha) \equiv \{x \in U : y \in m(\alpha) \text{ for all } y \text{ such that } x\ sim\ y\}.$$

Similarly one defines $m(\Box_1\alpha)$, and $m(\Box_2\alpha)$.

α is true in the model $\mathscr{M}$, if $m(\alpha) = U$.

The following deductive system for *NIL* was proposed in [**?**].
Axiom schemes:

$A1$. All classical tautologies,
$A2$. $\Box_2(\alpha \to \beta) \to (\Box_2\alpha \to \Box_2\beta)$,
$A3$. $\Box_1(\alpha \to \beta) \to (\Box_1\alpha \to \Box_1\beta)$,
$A4$. $\Box(\alpha \to \beta) \to (\Box\alpha \to \Box\beta)$,
$A5$. $\alpha \to \Box_1\neg\Box_2\neg\alpha$,
$A6$. $\alpha \to \Box_2\neg\Box_1\neg\alpha$,
$A7$. $\Box_2\alpha \to \alpha$,
$A8$. $\Box_1\alpha \to \alpha$,
$A9$. $\Box\alpha \to \alpha$,
$A10$. $\Box_2\alpha \to \Box_2\Box_2\alpha$,
$A11$. $\Box_1\alpha \to \Box_1\Box_1\alpha$,
$A12$. $\alpha \to \Box\neg\Box\neg\alpha$,
$A13$. $\Box\alpha \to \Box_2\Box\Box_1\alpha$.

Rules of inference:

$$(R1)\ \frac{\alpha,\ \alpha \to \beta}{\beta} \qquad (R2)\ \frac{\alpha}{\Box_2\alpha}$$

$$(R3)\ \frac{\alpha}{\Box_1\alpha} \qquad (R4)\ \frac{\alpha}{\Box\alpha}$$

It has been proved that

Theorem 9.7. *For any NIL-wff α, $\vdash_{NIL} \alpha$ if and only if α is true in all NIL-models.*

9.2.2 Logics by Vakarelov

Vakarelov addresses the issue of completeness of various logics, the models of which are based on standard structures corresponding to some information system. For instance, in the case of *NIL*, the question would be about a completeness theorem with respect to the class of *NIL*-models defined on standard *NIL*-structures (cf. Section 9.2.1). In [**?**], such a theorem is proved, via a key characterization result. In fact, this result set the ground for a series of similar observations when the binary relations involved are changed.

Proposition 9.1. *(Characterization) Let (U, sim, in) be a general NIL-structure. Then there exists an information system $\mathscr{S} \equiv (U, A, Val, f)$ with f satisfying $(\diamond)$, such that $sim_{\mathscr{S}} = sim$ and $in_{\mathscr{S}} = in$.*

In other words, the classes of *NIL*-models based on standard and general *NIL*-structures are identical. Hence one obtains the required completeness theorem.

The condition ($\diamond$), viz. $f(x,a) \neq \emptyset$ for all $x \in U,\ a \in A$, is a restrictive one. However, it is observed by Vakarelov that even if this condition is dropped, a characterization result similar to Proposition 9.1 can be obtained. Instead of reflexivity of *sim* (cf. property ($S6$), Section 9.2.1), we now have just the condition of *quasireflexivity* – cf. property ($S3$): if x *sim* y for some y, then x *sim* x. The corresponding logic can be obtained from *NIL* by replacing the axiom $A9$ by

$$\neg\Box(p \wedge \neg p) \rightarrow (\Box\alpha \rightarrow \alpha).$$

Following this approach, one handles the cases of incomplete information systems inducing different binary relations. For example, [**?**, **?**, **?**] consider these relations amongst others, for $\mathscr{S} \equiv (U, A, Val, f)$:

Indiscernibility ($ind_{\mathscr{S}}$): $x\ ind_{\mathscr{S}}\ y$ if and only if $f(x,a) = f(y,a)$, for all $a \in A$,
Weak indiscernibility ($ind^{w}_{\mathscr{S}}$): $x\ ind^{w}_{\mathscr{S}}\ y$ if and only if $f(x,a) = f(y,a)$, for some $a \in A$,
Weak similarity ($sim^{w}_{\mathscr{S}}$): $x\ sim^{w}_{\mathscr{S}}\ y$ if and only if $f(x,a) \cap f(y,a) \neq \emptyset$, for some $a \in A$.
Complementarity (com): $x\ com\ y$ if and only if $f(x,a) = (VAL_a \setminus f(y,a))$, for all $a \in A$, where Val_a is the value set for the particular attribute a, and $Val \equiv \cup\{Val_a :\ a \in A\}$.

The characterization result for each has been obtained, the corresponding logical system is defined and the completeness theorem with respect to models on the intended standard structures is proved.

9.2.3 Logic by Nakamura

[**?**] discusses a logic with models on incomplete information systems. We recall (cf. Introduction) that given a complete information system $\mathscr{S} \equiv (U, A, Val, f)$, one can define the equivalence relation $R_{\mathscr{S}}$. The lower approximation of $X (\subseteq U)$ under this relation is denoted as $\underline{X}_{\mathscr{S}}$, and its upper approximation as $\overline{X}_{\mathscr{S}}$.
Nakamura defines a *completation* $\mathscr{S}_0$ of an incomplete information system $\mathscr{S}$ as a complete information system that can be constructed from $\mathscr{S}$ by selecting any one value from $f(x,a) (\subseteq Val)$, for each $x \in U, a \in A$. If $f(x,a) = \emptyset$, one selects a special symbol ε. The relationship of $\mathscr{S}_0$ and $\mathscr{S}$ is expressed as $\mathscr{S}_0 \geq \mathscr{S}$.

Now the "lower" and "upper approximations" $\underline{X}, \overline{X}$ of $X \subseteq U$ in an incomplete information system $\mathscr{S} \equiv (U, A, Val, f)$ are defined as follows:

$$(*) \quad \underline{X} \equiv \cap_{\mathscr{S}_0 \geq \mathscr{S}} \underline{X}_{\mathscr{S}_0}, \overline{X} \equiv \cup_{\mathscr{S}_0 \geq \mathscr{S}} \overline{X}_{\mathscr{S}_0}.$$

With this background, a logic *INCRL* is proposed, having the standard Boolean connectives, and two modal operators $[], \langle\rangle$ (corresponding to "surely" and "possibly" respectively).
An *INCRL*-model is an incomplete information system $\mathscr{S} \equiv (U, A, Val, f)$ along with a meaning function $v_{\mathscr{S}}$ from the set of propositional variables of the language to $\mathscr{P}(U)$. $v_{\mathscr{S}}$ is extended as usual for the wffs involving Boolean connectives. For

wffs with modal operators, one makes use of completations $\mathscr{S}_0$ of $\mathscr{S}$ and the preceding definitions of lower and upper approximations given in $(*)$.

$$v_{\mathscr{S}}([\,]\alpha) \equiv \cap_{\mathscr{S}_0 \geq \mathscr{S}} \underline{v_{\mathscr{S}}(\alpha)}_{\mathscr{S}_0} \;=\; \underline{v_{\mathscr{S}}(\alpha)},$$
$$v_{\mathscr{S}}(\langle\rangle\alpha) \equiv \cup_{\mathscr{S}_0 \geq \mathscr{S}} \overline{v_{\mathscr{S}}(\alpha)}_{\mathscr{S}_0} \;=\; \overline{v_{\mathscr{S}}(\alpha)}.$$

Truth and validity of wffs are defined again as for most of the previous systems. Nakamura points out relationships of *INCRL* with the modal system *KTB*, in particular that all theorems of *KTB* are valid wffs of *INCRL*. We shall take a further look at the two logics in Section 9.4.

9.3 Other approaches

This section outlines a few proposals of logics related to rough sets, the models of which are based on structures that are even more generalized than the ones already presented. As we shall see, these logics have dimensions not accounted for in the systems presented so far.

9.3.1 Temporal approach

Orłowska (cf. [**?**]), defines a logic $\mathscr{L}_{\mathscr{T}}$ with models on *dynamic information systems*, in order to deal with the temporal aspect of information. A set T of moments of time, and a suitable relation R on the set T are considered along with the set U of objects and A of attributes. Formally, a dynamic information system is a tuple $\mathscr{S} \equiv (U, A, Val, T, R, f)$, where $Val \equiv \cup\{Val_a : \; a \in A\}$, ($Val_a$, as in Section 9.2.2, being the value set for the particular attribute a) and the information function $f : U \times T \times A \rightarrow Val$ satisfies the condition that $f(x,t,a) \in VAL_a$, for any $x \in U$, $t \in T$, $a \in A$.

In the language of $\mathscr{L}_{\mathscr{T}}$, atomic statements are descriptors of decision logic, together with an object constant x – so these are triples (x,a,v), and are intended to express: "object x assumes value v for attribute a". There are modal operators to reflect the relations R and R^{-1}. The truth of all statements of the language is evaluated in a model based on a dynamic information system, with respect to moments of time, i.e. members of the set T.

An $\mathscr{L}_{\mathscr{T}}$-model is a tuple $\mathscr{M} \equiv (\mathscr{S}, m)$ where $\mathscr{S}$ is a dynamic information system, and m a meaning function which assigns objects, attributes and values from U, A, Val to the respective constants.

The satisfiability of a formula α in a model $\mathscr{M}$ at a moment $t (\in T)$ of time is defined inductively as follows:

$$\mathscr{M},\, t \models (x,a,v) \text{ if and only if } f(m(x),t,m(a)) = m(v).$$

For the Boolean cases, we have the usual definitions. For the modal case,

$$\mathscr{M},\, t \models [R]\alpha \text{ if and only if for all } t' \in T, \text{ if } (t,t') \in R \text{ then } \mathscr{M},\, t \models \alpha.$$

A wff is true in $\mathscr{M}$, provided it is satisfied in $\mathscr{M}$ at every $t \in T$. $\mathscr{L}_{\mathscr{T}}$ is complete with respect to this class of models, for the axioms of linear time temporal logic, and an axiom which says that the values of attributes are uniquely assigned to objects.

9.3.2 Multiagent systems

[**?**] describes a logic, that takes into account a (finite) collection of *agents* and their *knowledge bases*. We denote the logic as $\mathscr{L}_{\mathscr{M}\mathscr{A}}$. The language of $\mathscr{L}_{\mathscr{M}\mathscr{A}}$ has "agent constants" along with two special constants 0,1. Binary operations $+, \cdot$ are provided to build the set $\mathscr{T}$ of *terms* from these constants. Wffs of one kind are obtained from terms, and are of the form $s \Rightarrow t,\ s,t \in \mathscr{T}$, where $\Rightarrow$ is a binary relational symbol. $s \Rightarrow t$ is to reflect that "the classification ability of agent t is at least as good as that of agent s".

Furthermore, there are attribute as well as attribute-value constants. Descriptors formed by these constants constitute atomic propositions, and using connectives $\wedge, \neg$ and modal operators I_t, $t \in \mathscr{T}$ (representing "partial knowledge" of each agent), give wffs of another kind.

$\mathscr{L}_{\mathscr{M}\mathscr{A}}$-models are not approximation spaces, but what could be called "partition spaces" on information systems. Informally put, a model consists of an information system $\mathscr{S} \equiv (U, A, Val, f)$, and a family of *partitions* $\{E_t\}_{t \in \mathscr{T}}$ on the domain U – each corresponding to the knowledge base of an agent. The family is shown to have a lattice structure, and the ordering involved gives the interpretation of the relational symbol $\Rightarrow$. Wffs built out of descriptors are interpreted in the standard way, in the information system $\mathscr{S}$. The partial knowledge operator I_t for a term t reflects the lower approximation operator with respect to the partition E_t on U. An axiomatization of $\mathscr{L}_{\mathscr{M}\mathscr{A}}$ is presented, to give soundness and completeness results.

In the context of multiagent systems, it is worth mentioning the approach followed in [**?**], even though a formal logic based on it has not been defined yet. *Property systems* (P-systems) are defined as triples of the form $(U, A, \models)$, where U is a set of objects, A a set of *properties*, and $\models$ a "fulfilment" relation between U and A. For each P-system $\mathscr{P}$, a collection $\mathscr{P}^{op}$ of *interior* and *closure* operators satisfying specific properties are considered. These operators could be regarded as generalizations of lower and upper approximations. Now given a family $\{\mathscr{P}_k\}_{k \in K}$ of P-systems (each for an agent, say) over some index set K and over the same set U of objects, one obtains a *multiagent pre-topological approximation space* as a structure $(U, \{\mathscr{P}_k^{op}\}_{k \in K})$. It is to be seen if such a generalized structure could form the basis of a semantics of some formal logical framework.

9.3.3 Rough relations

Discussion about relations on approximation spaces, started from [**?**]. We find two directions of work on this topic.

9.3.3.1 Logic of rough relations:

[**?**] considers another generalization of the notion of an approximation space – taking systems of the form $\mathscr{A}S \equiv (U,I,v)$, where U is a non-empty set of objects, $I : U \to \mathscr{P}(U)$ an *uncertainty function*, and $v : \mathscr{P}(U) \times \mathscr{P}(U) \to [0,1]$ is a *rough inclusion* function satisfying the following conditions:

$v(X,X) = 1$ for any $X \subseteq U$,
$v(X,Y) = 1$ implies $v(Z,Y) \geq v(Z,X)$ for any $X,Y,Z \subseteq U$,
$v(\emptyset,X) = 1$ for any $X \subseteq U$.

For any subset X of U, we then have the *lower* and *upper approximations*:

$L(AS,X) \equiv \{x \in U : v(I(x),X) = 1\}$,
$U(AS,X) \equiv \{x \in U : v(I(x),X) > 0\}$.

A 'rough set' in AS is the pair $(L(AS,X),U(AS,X))$.

The above is motivated from the fact that any Pawlak approximation space (U,R) is an instance of a generalized space as just defined. Indeed, we consider the function I that assigns to every object its equivalence class under R, and the inclusion function v as:

$$v(S,R) \equiv \begin{cases} \frac{card(S\cap R)}{card(S)} & \text{if } S \neq \emptyset \\ 1 & \text{if } S = \emptyset \end{cases}$$

For an approximation space $AS \equiv (U,I,v)$ with $U = U_1 \times U_2$ and v as in the special case above, [**?**] discusses relations $R \subseteq U_1 \times U_2$. The lower and upper approximation of R in AS are taken, and a *rough relation* is just a rough set in AS.

A decidable multimodal logic is proposed – for reasoning about properties of rough relations. The modal operators correspond to a set of relations on the domain of the above generalized approximation spaces, as well as the lower and upper approximations of these relations. An axiomatization for the logic is given, and completeness is proved with respect to a Kripke-style semantics.

9.3.3.2 Rough datalog:

Just as *decision tables* [**?**] are (complete) information systems with special attributes, viz. the *decision attributes*, [**?**] considers a *decision system* $(U,A\cup\{d\})$ – but with a difference. Each attribute a in A is a *partial map* from U to a value set V_a, and d, the decision attribute, is a partial map from U to $\{0,1\}$. It is possible that for some $x \in U$, all attribute values (including the value of d) are undefined. A 'rough set' X is taken to be a pair (X^+, X^-), where X^+ is the set of elements of U that may belong to X, while X^- contains those elements of U that may not belong to X. d indicates the information about membership of an object of U in X.

Formally, let $A \equiv \{a_1,...,a_n\}$, $A(x) \equiv (a_1(x),...,a_n(x))$ for each $x \in U$, and $A^{-1}(t) \equiv \{x \in U : A(x) = t\}$, for $t \in V_{a_1} \times ... \times V_{a_n}$. (Note that for some $x \in U$, $A(x)$ could be undefined). Then

$X^+ \equiv \{x \in U : A$ is defined for x, and $d(x') = 1$, for some $x' \in A^{-1}(A(x))\}$, and
$X^- \equiv \{x \in U : A$ is defined for x, and $d(x') = 0$, for some $x' \in A^{-1}(A(x))\}$.

This definition implies that X^+ and X^- may not be disjoint, allowing for the presence of conflicting (contradictory) decisions in the decision table. On the other hand, X^+ and X^- may not cover U either, allowing for the possibility that there is no available information about membership in X.

With these definitions, 'rough relations' are considered in [**?**]. Standard relational data base techniques, such as relational algebraic operations (e.g. union, complement, Cartesian product, projection) on crisp relations, are extended to the case of rough relations. A declarative language for defining and querying these relations is introduced - pointing to a link of rough sets (as just defined) with logic programming.

9.3.4 Logics with attribute expressions

As we have seen, $\mathscr{L}_{\mathscr{T}}$ and $\mathscr{L}_{\mathscr{M}\mathscr{A}}$ (cf. Sections 9.3.1 and 9.3.2 respectively) have attribute expressions in the language that are interpreted in information systems. *NIL* (cf. Section 9.2.1), also has attribute constants in the language. But unlike the models of $\mathscr{L}_{\mathscr{T}}$ and $\mathscr{L}_{\mathscr{M}\mathscr{A}}$, the standard or general *NIL*-structures defining *NIL*-models do not accommodate attributes, and the wffs (which are built using the attribute constants) point to collections of objects of the domain.

A class of logics with attribute expressions are also defined in [**?**, **?**]. Models are based on structures of the form $(U, A, \{ind(P)\}_{P \subseteq A})$, where the "indiscernibility" relation $ind(P)$ for each subset P of the attribute set A, has to satisfy certain conditions. For the models of one of the logics, for example, the following conditions are stipulated for $ind(P)$:

$(U1)$ $ind(P)$ is an equivalence relation on U,
$(U2)$ $ind(P \cup Q) = ind(P) \cap ind(Q)$,
$(U3)$ if $P \subseteq Q$ then $ind(Q) \subseteq ind(P)$, and
$(U4)$ $ind(\emptyset) = U \times U$.

Other logics may be obtained by changing some of $(U1) - (U4)$. The language of the logics has a set of variables each representing a set of attributes, as well as constants to represent all one element sets of attributes. Further, the language can express the result of (set-theoretic) operations on sets of attributes. The logics are multimodal – there is a modal operator to reflect the indiscernibility relation for each set of attributes as above. A usual Kripke-style semantics is given, and a number of valid wffs presented. However, as remarked in [**?**], we do not know of a complete axiomatization for such logics.

9.3.5 Rough mereology

This is an approach inspired by the theory of *mereology* due to Leśniewski (1916). Leśniewski propounds a theory of sets that has *containment* as the primitive relation, rather than membership. Drawing from this classical theory, *rough mereology* has been proposed [**?**], providing a useful notion of *rough containment*, of "being a part, *in a degree*".

Formally, this can be defined as a real binary function μ on the domain with values in [0,1], satisfying certain conditions (abstracted from the properties of classical containment). A given information system (U,A,Val,f), a partition of A into, say $A_1,...,A_n$ and a set of weights $\{w_1,...,w_n\}$, can generate $\mu(x,y)$, $x,y \in U$. It is assumed that $w_i \in [0,1]$, $i = 1,...,n$, and $\sum_{i=1}^{n} w_i = 1$.
A *pre-rough* inclusion μ_o is first defined:

$$\mu_o(x,y) \equiv \sum_{i=1}^{n} w_i.(|ind_i(x,y)|/|A_i|),$$

where $ind_i(x,y) \equiv \{a \in A_i : f(x,a) = f(y,a)\}$. μ_o can then be extended to *rough inclusion* μ over $\mathscr{P}(U)$ by using t-norms and t-conorms. Rough inclusion can be used, for instance, in specifying approximate decision rules.

It may be remarked that predicate logics corresponding to rough inclusions have been proposed recently in [**?**].

9.4 Comparative Study

We now discuss some relationships between the logics presented in Sections 9.1 and 9.2.

9.4.1 Embeddings

Let us recall the notion of an embedding of logics – cf. Section 9.1.4. We consider the logics $\mathscr{L}3$, *PRL*, $\mathscr{L}_{\mathscr{D}}$, *RDSL* presented in Sections 9.1.3, 9.1.4 and 9.1.5 respectively, and point out interrelationships, as well as relations with other known logics.

9.4.1.1 (1) $\mathscr{L}3 \rightleftharpoons$ *PRL*:

This has already been seen in Section 9.1.4.

9.4.1.2 (2) $\mathscr{L}3 \rightleftharpoons \mathscr{L}_{\mathscr{D}}$:

As summarized in [**?**] and observed by Düntsch and Pagliani, regular double Stone algebras and 3-valued Łukasiewicz algebras are equivalent to each other via suitable transformations. Passing on to the respective logics, we would thus find embeddings both ways, between $\mathscr{L}_{\mathscr{D}}$ and $\mathscr{L}3$.

9.4.1.3 (3) $\mathscr{L}_{\mathscr{D}} \rightleftharpoons$ *RDSL*:

We can define, in *RDSL*, that *a wff* α *is a theorem* (*valid*), if and only if the sequent $\top \Rightarrow \alpha$ is derivable (valid). Using the formal argument made in Section 9.1.5 to show that the classes of models of the logics $\mathscr{L}_{\mathscr{D}}$ and *RDSL* are identifiable and Theorems 9.4, 9.5, one gets the result with the identity embedding.

9.4.1.4 (4) $\mathscr{L}3 \rightleftharpoons \mathscr{L}_{SN}$:

$\mathscr{L}_{SN}$ denotes constructive logic with strong negation [**?**]. We note that semi-simple Nelson algebras are the algebraic counterparts for $\mathscr{L}_{SN}$. The equivalence of semi-simple Nelson algebras and 3-valued Łukasiewicz algebras through suitable translations has also been observed e.g. by Pagliani. Hence the stated embedding.

9.4.1.5 (5) *PRL* $\rightharpoonup$ *S*5:

One observes [**?**] a translation * of wffs of *PRL* into *S*5 that assigns the operations of negation $\neg$ and necessity $\Box$ in *PRL* those same operations of *S*5. Further, $\sqcap$ is translated in terms of the conjunction $\wedge$ and disjunction $\vee$ of *S*5 as:

$$(\alpha \sqcap \beta)^\star \equiv (\alpha^\star \wedge \beta^\star) \vee (\alpha^\star \wedge M\beta^\star \wedge \neg M(\alpha^\star \wedge \beta^\star)).$$

Then it can be shown that $\vdash_{PRL} \alpha$ if and only if $\vdash \alpha^\star$, for any wff α of *PRL*.

9.4.1.6 (6) *S*5 $\rightleftharpoons L_r$:

The logic L_r for rough truth is able to capture, as the class of its theorems, exactly the "$\Diamond$-image" of the class of *S*5-theorems, i.e. $\vdash_{S5} \Diamond\alpha$ if and only if $\mid\!\sim \alpha$ [**?**, **?**]. Note that the languages of L_r and *S*5 are the same. We translate α in *S*5 to $\alpha^* \equiv L\alpha$. Then $\vdash \alpha$ if and only if $\mid\!\sim \alpha^*$. For the other direction, we consider the translation $\alpha^\circ \equiv M\alpha$.

9.4.1.7 (7) $J \rightleftharpoons L_r$**:**

In 1948, Jaśkowski proposed a "discussive" logic – he wanted a formalism to represent reasoning during a discourse. Each thesis, a *discussive assertion* of the system, is supposed either to reflect the opinion of a participant in the discourse, or to hold for a certain "admissible" meaning of the terms used in it. Formally, any thesis α is actually interpreted as "it is possible that α", and the modal operator $\Diamond$ is used for the expression. The logic J (cf. [**?**]) is such a system. The J-consequence, defined over $S5$, is such that:

$\vdash_J \alpha$ if and only if $\vdash_{S5} \Diamond\alpha$.

Because of the relationship between L_r and $S5$ noted in (6) above, we have $J \rightleftharpoons L_r$ with the identity embedding. In the whole process, one has obtained an alternative formulation of the paraconsistent logic J (proposed in a different context altogether), and established a link between Pawlak's and Jaśkowski's ideas.

9.4.2 KTB and Nakamura's logic $INCRL$

We refer to Section 9.2.3, and present a connection between $INCRL$, and the normal modal system KTB. KTB, as we know, is sound and complete with respect to the class of reflexive and symmetric Kripke frames.

Let $\mathscr{S} \equiv (U, A, Val, f)$ be an incomplete information system, and let us consider the relation $\Re$ on U defined as follows:

$x \Re y$ if and only if there exists a completation $\mathscr{S}_0$ of $\mathscr{S}$ such that $x R_{\mathscr{S}_0} y$.

Clearly $\Re$ is reflexive and symmetric, but not transitive. From the definitions of $v_{\mathscr{S}}([\,]\alpha)$ and $v_{\mathscr{S}}(\langle\rangle\alpha)$, we see that

$x \in v_{\mathscr{S}}([\,]\alpha)$ if and only if, for all $y \in U$ such that $x \Re y, y \in v_{\mathscr{S}}(\alpha)$, and

$x \in v_{\mathscr{S}}(\langle\rangle\alpha)$ if and only if, there exists $y \in U$ such that $x \Re y$ and $y \in v_{\mathscr{S}}(\alpha)$.

So all provable wffs of the modal logic KTB are valid in $INCRL$. What about the converse – are all valid wffs of $INCRL$ provable in KTB? [**?**] makes a cryptic comment about this, we establish the converse here.

9.4.2.1 KTB provides an axiomatization for $INCRL$:

We show that if α is not provable in KTB then it is not valid in $INCRL$. It suffices then, to construct an incomplete information system $\mathscr{S} \equiv (U, A, \{Val_a\}_{a \in A}, f)$ for any given KTB-frame (W, R), such that $\Re$ is identical with R.

Let g be a function from R $(\subseteq W \times W)$ to some set C of constants, satisfying the following conditions:

(i) $g(x,y) = g(y,x)$, (ii) $g(x,y) = g(t,z)$ implies that either $x = t$ and $y = z$, or $x = z$ and $y = t$.

(g essentially assigns, upto symmetry, a unique constant from C to every pair in R.)

Now consider $U \equiv W$, $A \equiv \{a\}$, where a is a *new* symbol. Further, define $f(x,a) \equiv \{g(x,y) : y \in U$ and $(x,y) \in R\}$, so that $Val_a \subseteq C$.

We claim that xRy if and only if $x \,\mathfrak{R}\, y$. Suppose xRy. Then $g(x,y) \in f(x,a) \cap f(y,a)$ and hence $x \,\mathfrak{R}\, y$. Conversely, if $x \,\mathfrak{R}\, y$, there exists $d \in f(x,a) \cap f(y,a)$. Now $d \in f(x,a)$ implies that $d = g(x,z)$, for some $z \in U$ such that $(x,z) \in R$, and $d \in f(y,a)$ implies that $d = g(y,t)$, for some $t \in U$ such that $(y,t) \in R$. From the property of g, it follows that either $x = y$ or $x = t$, whence by reflexivity and symmetry of R, we get xRy.

The proof above, in fact, yields a characterization theorem, viz. given any reflexive, symmetric frame (W,R), there exists an incomplete information system $\mathcal{S} \equiv (U,A,\{Val_a\}_{a\in A}, f)$ satisfying the condition $(\diamond)$ (cf. Section 9.2.1) such that $R = \mathfrak{R} = sim_{\mathcal{S}}$.

9.4.3 *Normal modal systems and Vakarelov's logics*

Vakarelov has proved the characterization theorem for incomplete information systems with respect to different sets of relations [**?**, **?**, **?**, **?**]. As we have remarked in the Introduction, a special case would be obtained with respect to the indiscernibility relation on the Pawlak approximation space. One finds that if we restrict the logics presented in [**?**, **?**, **?**] to take a modal operator corresponding only to the indiscernibility relation, the resulting system would be just the modal logic $S5$.

As noted at the end of Section 9.4.2, if an incomplete information system satisfies the condition $(\diamond)$, then the similarity relation $sim_{\mathcal{S}}$ is the same as the relation $\mathfrak{R}$. So it follows that if we restrict the logic NIL to take only the modality $\Box$ in the language then the corresponding logic will be just $INCRL$, or, in other words, KTB.

9.4.4 *DAL* *again*

Observing Vakarelov's strain of work, it may be tempting to look for a kind of characterization result in the case of *DAL* (cf. Section 9.1.2) as well. Consider a general *DAL*-structure $\mathcal{U} \equiv (U,\{R_i\}_{i\in I})$, where the family $\{R_i\}_{i\in I}$ of equivalence relations is closed under intersection and transitive closure of union. Can one find an incomplete information system $\mathcal{S} \equiv (U,A,Val,f)$ such that the standard structure for $\mathcal{S}$ is just $\mathcal{U}$? Let us assume that the standard structure is obtained "naturally" from $\mathcal{S}$, viz. that the equivalence relations in it are the ones induced by the *subsets* of A. As it turns out, this is a hard question.

However, we can find an information system, such that the standard structure obtained from it in the above manner cannot be a general *DAL*-structure.

Suppose for some incomplete information system $\mathcal{S} \equiv (U,A,Val,f)$, R and P are the equivalence relations induced by subsets R', P' of A respectively – we denote this as $ind(R') = R$ and $ind(P') = P$. For the equivalence relation $R \cap P$, $R' \cup P' \subseteq A$ is such that $ind(R' \cup P') = R \cap P$. But in the case of $R \uplus P$, there may not be any $Q \subseteq A$ such that $ind(Q) = R \uplus P$. Consider the following example [**?**].

Example 9.1. $U \equiv \{o1, o2, o3, o4, o5, o6, o7\}$, where each o_i consists of circles and squares. Let $A \equiv \{$number of circles ($\bigcirc$), number of squares ($\Box$)$\}$. The information function is given by the following table:

	$\bigcirc$	$\Box$
$o1$	1	1
$o2$	1	2
$o3$	2	1
$o4$	2	2
$o5$	3	3
$o6$	3	4
$o7$	3	4

Equivalence classes of indiscernibility relations $ind(\bigcirc)$ and $ind(\Box)$ are:

$ind(\bigcirc)$: $\{o1, o2\}, \{o3, o4\}, \{o5, o6, o7\}$,

$ind(\Box)$: $\{o1, o3\}, \{o2, o4\}, \{o5\}, \{o6, o7\}$.

The transitive closure of these relations gives the following equivalence classes:

$ind(\bigcirc) \uplus ind(\Box)$: $\{o1, o2, o3, o4\}, \{o5, o6, o7\}$.

Clearly there is no $Q \subseteq A$ such that $ind(Q) = ind(\bigcirc) \uplus ind(\Box)$.

9.5 Summary and questions

We have tried to present the various proposals of logics with semantics based on rough sets, including some generalizations. Two main approaches emerge, discussed in Sections 9.1 and 9.2. One of these considers logics, the models of which are approximation spaces, while the other considers approximation spaces, but those induced by information systems. However, it is found through characterization results, that both lines of study converge, in that the two semantics for a particular system are identical. This actually reflects on the apt description of the properties of the relations defining the approximation spaces.

The only exception is the logic *DAL* of the first category. As remarked in Section 9.4.4, given a general *DAL*-structure $\mathscr{U} \equiv (U, \{R_i\}_{i \in I})$, it does not seem easy to construct an information system "naturally" to obtain $\mathscr{U}$ back as its standard structure. In case of the logics with attributes as expressions (cf. Section 9.3.4), one encounters a problem even earlier. The models here are based on structures of the form $(U, A, \{ind(P)\}_{P \subseteq A})$, and there does not appear easily a corresponding "general" structure of the kind $\mathscr{U} \equiv (U, \{R_i\}_{i \in I})$, with appropriate closure conditions on

$\{R_i\}_{i\in I}$. These logics have not been axiomatized, though the language can express a lot about attributes – that few of the other systems are able to do.

An interesting picture is obtained from the logics of Section 9.1, leaving out *DAL* and other systems with models based on generalized spaces. Most of the logics are embeddable into each other (cf. Section 9.4). We have

$$\mathscr{L}_{\mathscr{D}} \rightleftharpoons \mathscr{L}3 \rightleftharpoons PRL \rightharpoonup S5 \rightleftharpoons L_r \rightleftharpoons J. \quad (1)$$

In one sense then, the embeddings in (1) establish that no 'new' logic surfaces with the kind of rough set semantics defined. But in another sense, well-known systems have been imparted a rough set interpretation. It should be noted that though the embeddings are defined with respect to *theoremhood*, the relationships would hold in some cases (e.g. $\mathscr{L}3 - PRL$ and $L_r - J$) if derivability of wffs from non-empty premise sets is considered [**?**, **?**]. One could attempt to settle the question for the rest. (1) indicates another interesting future line of work, viz. an investigation for logics and interrelations, that may result on replacing $S5$ by other non-modal systems (as in [**?**]).

All the systems presented other than $\mathscr{L}_{\mathscr{T}}$ (cf. Section 9.3.1), deal with static information. The semantics of $\mathscr{L}_{\mathscr{T}}$ essentially gives rise to a family of approximation spaces on the same domain, the indiscernibility relations changing with moments of time. One could further enquire about the behaviour of rough sets in such a dynamic information system.

As remarked in Section 9.3.2, another open direction relates to a study of logics that may be obtained from the generalized approach in [**?**].

Overall, one may say that it has been a remarkable journey in the exploration of logics, beginning with a deceptively simple proposal of “rough sets". We have seen the introduction of novel concepts – e.g. of “rough truth", “rough modus ponens", “rough consistency", “rough mereology". The journey has, by no means, ended. Pawlak's theory has just opened up the horizon before us, to reveal a number of yet unexplored directions in the study of “rough logics".

References

1. Banerjee, M.: Rough sets and 3-valued Łukasiewicz logic. Fundamenta Informaticae **32**, 213–220 (1997)
2. Banerjee, M., Chakraborty, M. K.: Rough algebra. Bull. Polish Acad. Sc.(Math.) **41**(4), 293–297 (1993)
3. Banerjee, M., Chakraborty, M. K.: Rough sets through algebraic logic. Fundamenta Informaticae **28**(3-4), 211–221 (1996)
4. Banerjee, M., Chakraborty, M. K.: Algebras from rough sets. In: Pal, S. K., Polkowski, L., Skowron, A. (eds.) Rough-neuro Computing: Techniques for Computing with Words, pp. 157–184. Springer Verlag, Berlin (2004)
5. Banerjee, M. and Pal, K.: The variety of topological quasi-Boolean algebras. In: Trends in Logic III – Int. Conf. in Memoriam A. Mostowski, H. Rasiowa, C. Rauszer, Warsaw, Poland (2005). Extended Abstract.
   `http://www.studialogica.org/`.

6. Banka, H. and Mitra, S.: Feature Selection, Classification and Rule Generation Using Rough Sets. In: Peters, G., Lingras, P., Ślezak, D. and Yao, Y. (eds.) Rough Sets: Selected Methods and Applications in Management and Engineering, pp. 51–76. Springer, London (2012)
7. Bazan, J. G.: A Comparison of Dynamic and non-Dynamic Rough Set Methods for Extracting Laws from Decision Tables. In: Polkowski, L. and Skowron, A. (eds.) Rough Sets in Knowledge Discovery 1: Methodology and Applications, Series Studies in Fuzziness and Soft Computing, vol 18, pp. 321–365. Physica-Verlag, Heidelberg (1998)
*@bookpolkowski98rough-methodology, editor = Lech Polkowski and Andrzej Skowron, title = Rough Sets in Knowledge Discovery 1: Methodology and Applications, publisher = Physica-Verlag, year = 1998, address = Heidelberg, series = Studies in Fuzziness and Soft Computing, volume = 18
8. Bazan, J. G., Hung Son Nguyen, Sinh Hoa Nguyen, Synak, P. and Wróblewski, J.: Rough set algorithms in classification problems. In: Polkowski, L., Lin, T. Y. and Tsumoto, S. (eds.) Rough Set Methods and Applications: New Developments in Knowledge Discovery in Information Systems, Series Studies in Fuzziness and Soft Computing, vol 56, pp. 49–88. Springer-Verlag/Physica-Verlag, Heidelberg (2000)
9. Blackburn, P., de Rijke, M., Venema, Y.: Modal Logic. Cambridge Univ. Press (2001)
10. Boicescu, V., Filipoiu, A., Georgescu, G., Rudeano, S.: Łukasiewicz-Moisil Algebras. North Holland, Amsterdam (1991)
11. Brown, F.M.: Boolean Reasoning. Kluwer Academic Publishers, Dordrecht, The Netherlands (1990)
12. Cattaneo, G.: Abstract approximation spaces for rough theories. In: Polkowski, L. and Skowron, A. (eds.) Rough Sets in Knowledge Discovery 1: Methodology and Applications, pp. 59–98. Physica-Verlag, Heidelberg (1998)
13. Chuchro, M.: On rough sets in topological Boolean algebras. In: Ziarko, W. P. (ed.) Rough Sets, Fuzzy Sets and Knowledge Discovery (Proceedings RSKD'93), pp. 157–160. Springer Verlag, London (1994)
14. Cignoli, R.: Representation of Łukasiewicz and Post algebras by continuous functions. Colloq. Math. **24**(2), 127–138 (1972)
15. Comer, S.: An algebraic approach to the approximation of information. Fundamenta Informaticae **14**, 492–502 (1991)
16. Comer, S.: Perfect extensions of regular double Stone algebras. Algebra Universalis **34**, 96–109 (1995)
17. Düntsch, I.: Rough sets and algebras of relations. In: Orłowska, E. (ed.) Incomplete Information: Rough Set Analysis, pp. 95–108. Physica-Verlag (1998)
18. Epstein, G., Horn, A.: Chain based lattices. J. of Mathematics **55**(1), 65–84 (1974) Reprinted In: Rine, D. C. (ed.) Computer Science and Multiple-valued Logic. Theory and Applications, pp. 58–76. North Holland (1991)
19. Garcia-Molina, H., Ullman, J.D. and Widom, J.D.: Database Systems: The Complete Book. Prentice Hall, Upper Saddle River, New Jersey (2002)
20. Grzymała-Busse, J.W.: LERS – A System for Learning from Examples based on Rough Sets. In: Słowiński, R. (ed.) Intelligent Decision Support – Handbook of Applications and Advances of the Rough Sets Theory, Series System Theory, Knowledge Engineering and Problem Solving, vol 11, pp. 3–18. Kluwer Academic Publishers, Dordrecht, The Netherlands (1992)
21. Grzymała-Busse, J.W.: LERS – A Data Mining System. In: Oded Maimon and Lior Rokach (eds.) The Data Mining and Knowledge Discovery Handbook, pp. 1347–1351. Springer, Heidelberg (2005)
22. Grzymała-Busse, J. W. and Rzasa, W.: Local and global approximations for incomplete data. In: Peters, J. F. and Skowron, A. (eds.) Transactions on Rough Sets VIII: Journal Subline, Series Lecture Notes in Computer Science, vol 4100, pp. 21–34. Springer, Heidelberg (2008)
23. Hastie, T., Tibshirani, R. and Friedman, J. H.: The Elements of Statistical Learning: Data Mining, Inference, and Prediction. Springer-Verlag, Heidelberg (2001)

24. He, X. and Xu, L. and Shen, W.: Dynamic information system and its rough set model based on time sequence. In: Yan-Qing Zhang and Lin, T.Y. (eds.) Proceedings of the IEEE International Conference on Granular Computing, Atlanta, May 10-12, 2006, pp. 542-545. IEEE Press (2006)
25. Higgs, D.: A category approach to Boolean-valued set theory. Preprint, University of Waterloo (1973)
26. Iturrioz, L.: Rough sets and three-valued structures. In: Orłowska, E. (ed.) Logic at Work: Essays Dedicated to the Memory of Helena Rasiowa, pp. 596–603. Physica-Verlag (1999)
27. Iwiński, T. B.: Algebraic approach to rough sets. Bull. Polish Acad. Sc. (Math.) **35**(9-10), 673–683 (1987)
28. Jinfu Liu, Qinghua Hu and Daren Yu: A comparative study on rough set based class imbalance learning. Knowledge-Based Systems **21**, 753–763 (2008)
29. Kryszkiewicz, M.: Rough set approach to incomplete information systems. Information Sciences **112**, 39–49 (1998)
30. Kryszkiewicz, M.: Rules in incomplete information systems. Information Sciences **113**, 271–292 (1999)
31. Kuroki, N.: Rough ideals in semigroups. Information Sciences **100**, 139–163 (1997)
32. Łukasiewicz, J.: Die logischen Grundlagen der Wahrscheinlichkeitsrechnung, 1913. In: Borkowski, L. (ed.) Jan Łukasiewicz: Selected Works. pp. 16–63. North Holland Publishing Company, Amsterdam (1970)
33. Moisil, Gr. C.: Sur les idéaux des algébres łukasiewicziennes trivalentes. An. Univ. C.I. Parhon, Acta Logica **3**, 83–95 (1960)
34. Monteiro, A.: Construction des algebres de Łukasiewicz trivalentes dans les algebres de Boole monadiques. I. Math. Japonicae **12**, 1–23 (1967)
35. Mundici, D.: The C^*-algebras of three-valued logic. In: Ferro, R. et al. (eds.) Logic Colloquium '88, pp. 61–77. North Holland, Amsterdam (1989)
36. Oded Maimon and Lior Rokach (eds.): The Data Mining and Knowledge Discovery Handbook. Springer, Heidelberg (2005)
37. Nguyen, H.S.: Approximate Boolean reasoning: Foundations and Applications in Data Mining. In: Peters, J. F. and Skowron, A. (eds.) Transactions on Rough Sets V: Journal Subline, Series Lecture Notes in Computer Science, vol 4100, pp. 344–523. Springer, Heidelberg (2006)
38. Nguyen, H.S. and Nguyen, S.H.: Rough sets and association rule generation. Fundamenta Informaticae **40**(4), 383–405 (1999)
39. Nguyen, H.S. and Skowron, A.: Rough Sets: From Rudiments to Challenges. In: Skowron, A. and Suraj, Z. (eds.) Rough Sets and Intelligent Systems. Professor Zdzisław Pawlak in Memoriam, Series Intelligent Systems Reference Library, vol 42-43, pp. 75–173. Springer, Heidelberg (2013)
40. Nguyen, H.S. and Ślezak, D.: Approximate Reducts and Association Rules - Correspondence and Complexity Results. In: Skowron, A., Setsuo Ohsuga and Zhong, N. (eds.) Proceedings of the 7-th International Workshop on Rough Sets, Fuzzy Sets, Data Mining, and Granular-Soft Computing (RSFDGrC'1999), Yamaguchi, November 9-11, 1999, Series Lecture Notes in Artificial Intelligence, vol 1711, pp. 137–145. Springer, Heidelberg (1999)
41. Orłowska, E.: Dynamic information system. Fundamenta Informaticae **5**, 101–118 (1982)
42. Orłowska, E.: Introduction: what you always wanted to know about rough sets. In: Orłowska, E. (ed.) Incomplete Information: Rough Set Analysis, pp. 1–20. Physica-Verlag (1998)
43. Orłowska, E.: Many-valuedness and uncertainty. International Journal of Multiple Valued Logic **4**, 207–277 (1999)
44. Orłowska, E. and Pawlak, Z.: Representation of nondeterministic information. Theoretical Computer Science **29**, 27–39 (1984)
45. Pawlak, Z.: Information systems - theoretical foundations. Information Systems **6**, 205–218 (1981)
46. Peters, J. F. and Skowron, A. (eds.): Transactions on Rough Sets V: Journal Subline, Series Lecture Notes in Computer Science, vol 4100. Springer, Heidelberg (2006)

47. Peters, G., Lingras, P., Ślezak, D. and Yao, Y. (eds.): Rough Sets: Selected Methods and Applications in Management and Engineering. Springer, London (2012)
48. Polkowski, L., Lin, T. Y. and Tsumoto, S. (eds.): Rough Set Methods and Applications: New Developments in Knowledge Discovery in Information Systems, Series Studies in Fuzziness and Soft Computing, vol 56. Springer-Verlag/Physica-Verlag, Heidelberg (2000)
49. Pomykała, J. A.: Approximation, similarity and rough construction. Preprint, ILLC Prepublication Series CT–93–07, University of Amsterdam (1993)
50. Pomykała, J. and Pomykała, J. A.: The Stone algebra of rough sets. Bull. Polish Acad. Sc. (Math.) **36**, 495–508 (1988)
51. Rasiowa, H.: An Algebraic Approach to Non-classical Logics. North Holland, Amsterdam (1974)
52. Rissanen, J.: Modeling by shortest data description. Automatica **14**, 465–471 (1978)
53. Rissanen, J.: Minimum-description-length principle. In: Kotz, S. and Johnson, N. (eds.) Encyclopedia of Statistical Sciences, pp. 523–527. John Wiley & Sons, New York, NY (1985)
54. Roman, S. (ed.): Introduction to Information and Coding Theory. Springer, Heidelberg (1997)
55. Sikorski, R.: Boolean Algebras. Springer-Verlag, New York (1969)
56. Skowron, A.: Boolean Reasoning for Decision Rules Generation. In: Jan Komorowski, J. and Raś, Z.W. (eds.) Proceedings of ISMIS'1993, Trondheim, Norway, June 15-18, 1993, Series Lecture Notes in Artificial Intelligence, vol 689, pp. 295–305. Springer-Verlag, Heidelberg (1993)
57. Skowron, A. and Grzymała-Busse, J. W.: From Rough Set Theory to Evidence Theory. In: Yager R., Fedrizzi, M. and Kacprzyk J. (eds.) Advances in the Dempster-Shafer Theory of Evidence, pp. 193–236. John Wiley & Sons, New York, NY (1994)
58. Skowron, A.: Extracting laws from decision tables. Computational Intelligence: An International Journal **11**(2), 371–388 (1995)
59. Skowron, A. and Suraj, Z. (eds.): Rough Sets and Intelligent Systems. Professor Zdzisław Pawlak in Memoriam, Series Intelligent Systems Reference Library, vol 42-43. Springer, Heidelberg (2013)
60. Ślezak, D.: Approximate reducts in decision tables. In: Proceedings Sixth International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems IPMU'1996, Granada, Spain, July 1-5, 1996, vol III, pp. 1159–1164 (1996)
61. Ślezak, D.: Various approaches to reasoning with frequency-based decision reducts: A survey. In: Polkowski, L., Lin, T. Y. and Tsumoto, S. (eds.) Rough Set Methods and Applications: New Developments in Knowledge Discovery in Information Systems, Series Studies in Fuzziness and Soft Computing, vol 56., pp. 235–285. Springer-Verlag/Physica-Verlag, Heidelberg (2000)
62. Ślezak, D.: Normalized decision functions and measures for inconsistent decision tables analysis. Fundamenta Informaticae **44**(3), 291–319 (2000)
63. Ślezak, D.: Approximate Entropy Reducts. Fundamenta Informaticae **53**(3-4), 365–387 (2002)
64. Słowiński, R. (ed.): Intelligent Decision Support - Handbook of Applications and Advances of the Rough Sets Theory. Series System Theory, Knowledge Engineering and Problem Solving, vol 11. Kluwer Academic Publishers, Dordrecht, The Netherlands (1992)
65. Stefanowski, J. and Tsoukiàs, A.: On the Extension of Rough Sets under Incomplete Information. In: Zhong, N., Skowron, A. and Ohsuga, S. (eds.) RSFDGrC '99: Proceedings of the 7th International Workshop on New Directions in Rough Sets, Data Mining, and Granular-Soft Computing, Yamaguchi, Japan, November 9-11, 1999, Series Lecture Notes in Artificial Intelligence, vol 1711, pp. 73–81. Springer-Verlag, Berlin Heidelberg (1999)
66. Vakarelov, D.: Notes on $\mathscr{N}$-lattices and constructive logic with strong negation. Studia Logica **XXXVI**, 109–125 (1977)
67. Wasilewska, A., Vigneron, L.: Rough equality algebras. In: Wang, P. P. (ed.) Proceedings of the International Workshop on Rough Sets and Soft Computing (JCIS'95), pp. 26–30 (1995)
68. Wasilewska, A., Vigneron, L.: On generalized rough sets. Preprint
69. Wiweger, A.: On topological rough sets. Bull. Polish Acad. Sc. (Math.) **37**, 51–62 (1988)

70. Xiangjun Li and Fen Rao: An Rough Entropy Based Approach to Outlier Detection. Journal of Computational Information Systems **8**(24), 10501–10508 (2012)
71. Yao, Y.: Constructive and algebraic methods of the theory of rough sets. Information Sciences **109**, 21–47 (1998)
72. Yao, Y.: Generalized rough set models. In: Polkowski, L., Skowron, A. (eds.) Rough Sets in Knowledge Discovery 1: Methodology and Applications, pp. 286–318. Physica-Verlag, Heidelberg (1998)
73. Zheng Zheng and Guoyin Wang: RRIA: A Rough Set and Rule Tree Based Incremental Knowledge Acquisition Algorithm. Fundamenta Informaticae **59**(2-3), 299–313 (2004)

Chapter 10
Other Theories and Rough Set Theory

The reader can find in the literature many papers discussing relationships of rough sets with other approaches. In this chapter, we outline some relationships with the Dempster-Shafer approach. We also mention the relationships of rough sets with fuzzy sets and knn. The role of combination of rough sets with other approaches in applications is emphasised. Moreover, we outline an approach to conflict analysis based on the rough set approach.

10.1 Rough sets and Dempster-Shafer theory

In this section, we present some relationships of rough sets with evidence theory called also Dempster-Shafer theory [**?**]. Our outline is based on [**?**] where are included some generalizations of results from [**?**, **?**] as well as some new results. For more details, the reader is referred to the recent survey on relationships of the Dempster-Shafer theory and the rough set theory [**?**].

We first recall the basic functions used in Dempster-Shafer theory.

By Θ we denote a nonempty set called the *frame of discernment*.

A function $m : \mathcal{P}(\Delta) \longrightarrow [0,1]$, where $\Delta \subseteq \Theta$ is called the *mass function* if $m(\emptyset) = 0$ and $\sum_{\Delta \subseteq \Theta} m(\Delta) = 1$.

There are two more functions important in this theory These are the *belief function* $Bel : \mathcal{P}(\Theta) \longrightarrow [0,1]$ and the *plausibility function* $Pl : \mathcal{P}(\Theta) \longrightarrow [0,1]$. They are defined as follows.

$$Bel(\Delta) = \sum_{\Gamma \subseteq \Delta} m(\Gamma),$$

$$Pl(\Delta) = \sum_{\Gamma \cap \Delta \neq \emptyset} m(\Gamma),$$

where $\Delta \subseteq \Theta$.

These functions have a simple intuitive interpretation in the rough set framework over decision systems [**?**].

Let $\mathbb{A} = (U, C, d)$ be a decision system. We identify the set of decisions V_d with the frame of discernment Θ. Let us recall that by ∂_A we denote the generalized decision of $\mathbb{A}$.

Now, we can define the mass function of the decision system $\mathbb{A}$ by

$$m_{\mathbb{A}}(\Delta) = \frac{|\{x \in U : \partial_A(x) = \Delta\}|}{|U|},$$

where $\Delta \subseteq V_d$. In fact, one can easily check that the function $m_{\mathbb{A}}$ satisfies the requirements for the mass function.

One can ask for characterisation of the set of objects $x \in U$ such that $\partial_A(x) = \Delta$. In the case of decision system $DS = (U, C, d)$ with decision classes $X_i = \{x \in U : d(x) = i\}$, where $i = 1, \ldots, n$ and $n = | V_d | \geq 2$, we obtain a partition of U into the lower approximations $LOW(AS_C, X_1), \ldots, LOW(AS_C, X_n)$ of decision classes $X_1, \ldots, X_n$ and the (relative to C) boundary region

$$BD(AS_C, X_1, \ldots, X_n) = U \setminus \bigcup_{i=1,\ldots,n} LOW(AS_C, X_i)$$

of classification $\{X_1, \ldots, X_n\}$ of U, where $AS_C = (U, IND(C))$.

We have

$$BD(AS_C, X, U \setminus X) = BD(AS_C, X)$$

for non-empty sets $X \subseteq U$ different from U.

Let us now consider a partition of the boundary region $BD(AS_C, X_1, \ldots, X_n)$ created by non-empty components

$$\begin{aligned} BD(AS_C, X_1, \ldots, X_n; \Delta) = \\ \{x \in BD(AS_C, X_1, \ldots, X_n) \ : \\ [x]_C \subseteq \bigcup_{i \in \Delta} X_i \\ \text{and } [x]_C \cap X_i \neq \emptyset \text{ for all } i \in \Delta\}, \end{aligned}$$

where $\Delta \subseteq \{1, \ldots, n\}, | \Delta | \geq 2$.

The following equality holds:

$$\begin{aligned} BD(AS_C, X_1, \ldots, X_n; \Delta) = \\ \bigcap_{i \in J} BD(AS_C, X_i) \cap \bigcap_{i \notin \Delta} U \setminus BD(AS_C, X_i). \end{aligned} \tag{10.1}$$

Each component $BD(AS_C, X_1, \ldots, X_n; \Delta)$ comprises all objects from the union $\bigcup_{i \in \Delta} X_i$ of decision classes sharing the same generalized decision, that is, all objects $x \in U$ for which $\partial_{DS}(x) = \Delta$.

Now, we obtain the following two facts for the belief function $Bel_{\mathbb{A}}$ and the plausibility function $Pl_{\mathbb{A}}$ defined on the basis of the mass function $m_{\mathbb{A}}$ [**?**]:

$$Bel_{\mathbb{A}}(\Delta) = \frac{|\mathsf{LOW}_A(\bigcup_{i\in\Delta} : X_i)|}{|U|}, \tag{10.2}$$

$$PL_{\mathbb{A}}(\Delta) = \frac{|\mathsf{UPP}_A(\bigcup_{i\in\Delta} : X_i)|}{|U|}, \tag{10.3}$$

where $X_i = \{x \in U : d(x) = i\}$ is the decision class related to the decision i, and $\Delta \subseteq V_d$.

Moreover, one can also obtain an interpretation of the so called Dempster-Shafer rule of combination using a relevant operation on decision tables. The Dempster-Shafer rule of combination aggregates two mass functions m_1 and m_2 to a new mass function $m_1 \otimes m_2$ defined by

$$m_1 \otimes m_2(\emptyset) = 0$$

and

$$m_1 \otimes m_2(\Delta) = \frac{\sum_{A\cap B=\Delta} m_1(A)m_2(B)}{1 - \sum_{A\cap B=\emptyset} m_1(A)m_2(B)},$$

where $\emptyset \neq \Delta \subseteq V_d$.

In the case when the mass functions m_1 and m_2 are defined by the decision systems $\mathbb{A}_{\mathbb{1}}$ and $\mathbb{A}$, respectively, one can define a natural operation $\odot$ on these decision systems such that [**?**]:

$$m_{\mathbb{A}_{\mathbb{1}}} \otimes m_{\mathbb{A}} = m_{\mathbb{A}_{\mathbb{1}} \odot \mathbb{A}}. \tag{10.4}$$

Relationships between rough sets and evidence theory lead to different applications. In particular, new methods of inducing rules were developed searching for rules with the large support for unions of few decision classes and eliminating many other decision classes (see, e.g., [**?**]).

It is worthwhile mentioning that research on the further development of the Dempster-Shafer theory, especially related to applications of belief functions in combination with other approaches is intensively going on (see, *e.g.*, [**?**] for the combination with deep learning, proceedings of conferences `www.lgi2a.univ-artois.fr/events/belief2021`, and `bfasociety.org/`).

Problem 10.1. Interpretation of components of the boundary region of classification.
Prove the following equality $BD(AS_C, X_1, \ldots, X_n; \Delta) = \{x \in U : \ \partial_{DS}(x) = \Delta$ for $|\ \Delta \ | \geq 2$.

Problem 10.2. Interpretation of *Bel* function.
Prove that Eqn. 10.2 holds.

Problem 10.3. Interpretation of *Pl* function.
Prove that Eqn. 10.3 holds.

Problem 10.4. Aggregation of decision tables (systems).
Define the operation $\odot$ on decision systems (tables) such that Eqn. 10.4 holds.

10.2 Combination of rough sets with fuzzy sets and other approaches

10.2.1 Combination of rough sets and fuzzy sets

Fuzzy sets have been introduced by Lotfi A. Zadeh in 1965 [**?**]. Since than enormous number of papers and books have been published on fuzzy set theory and applications (see, *e.g.*, [**?**, **?**]).

Any classical set $X \subseteq U$ can be characterised by its charactersistic function

$$\chi_X : U \rightarrow \{0,1\}$$

defined by

$$\chi_X(x) = \begin{cases} 1 \text{ for } x \in X \\ 0 \text{ for } x \notin X. \end{cases}$$

The degree of membership of $x \in U$ to the set X can be either 0 or 1.

A fuzzy set is a class of objects with a continuum of grades of membership. Such a set is characterized by a membership (characteristic) function which assigns to each object a grade of membership ranging between zero and one. More formally, a fuzzy set

$$F : U \rightarrow [0,1]$$

assigns to each element of the universe of objects U the degree of membership $F(x) \in [0,1]$ of x to the fuzzy set F. Hence, the degree of membership of $x \in U$ to the fuzzy set F is a real number from the interval $[0,1]$. The function $F : U \rightarrow [0,1]$ is also called fuzzy membership function (identified with the fuzzy set F).

For more details about theory of fuzzy sets and applications the reader is referred to literature (see, *e.g.*, [**?**, **?**]).

In this section, we consider methods based on combination of rough sets and fuzzy sets. The literature concerning this topic is rich (see, *e.g.*, [**?**, **?**, **?**, **?**, **?**, **?**], [**?**, **?**, **?**, **?**, **?**, **?**]). This section can be treated as an introduction to this topic. Our discussion concentrates on a particular scheme of rough fuzzy approximation called fuzzy rough model. We also provide some comments on applications of combination of rough sets with fuzzy sets and other approaches.

Both fuzzy and rough set theory represent two different approaches to vagueness. Fuzzy set theory addresses *gradualness* of knowledge, expressed by the fuzzy membership, whereas rough set theory addresses *granularity* of knowledge, expressed by the indiscernibility relation. In their combination are taken at least the following aspects: (i) concepts may be fuzzy rather than exact (crisp) [**?**, **?**], (ii) the indiscernibility relation may be fuzzy not strict (crisp), *e.g.*, some objects are more similar to each other than others [**?**, **?**] and (iii) belonging to the lower and upper approximation may be expressed using fuzzy quantifiers (*e.g.*, like 'most') [**?**, **?**].

Rather than competing, the two theories complement each other. Specifically, rough set theory offers tools for approximating fuzzy membership functions by considering, *e.g.*, how the membership degree of an object is influenced by the memberships of similar objects. Rough sets and fuzzy sets can work synergistically, often with other soft computing approaches. The developed systems exploit the tolerance for imprecision, uncertainty, approximate reasoning and partial truth under soft computing framework and is capable of achieving tractability, robustness, and close resemblance with human like (natural) decision making for pattern recognition in ambiguous situations [**?**, **?**]. The developed methods have found applications in different domains such as bioinformatics and medical image processing. Recent applications in data analysis concern feature selection, instance selection, instance-based classification, cognitive networks, imbalanced classification], multi-instance classification and multi-label classification [**?**, **?**].

The objective of the rough-fuzzy integration is to provide a stronger paradigm of uncertainty handing in decision-making. Over the year many methods and applications, in particular in pattern recognition were developed on the basis of rough sets or fuzzy sets and on their combination. The methods based on combination of the approaches exploit different abilities of mixed languages used for generation and expressing patterns by both approaches based on rough set and fuzzy sets, respectively. This is making it possible to discover patterns of the higher quality in comparison with situations when they are used in isolation due to better possibility of approximation of the boundary regions of vague concepts. One should note that in this case the searching space for relevant patters is becoming larger in comparison with cases when single approach is used and developing efficient heuristics searching for relevant patterns is more challenging. The developed methods concern discovery of patterns such as decision rules, clusters and processes or feature selection. The reader can find more details in the literature (see, *e.g.*, [**?**, **?**, **?**]) for the rough set based methods and [**?**, **?**, **?**, **?**, **?**, **?**, **?**, **?**, **?**, **?**, **?**, **?**, **?**, **?**, **?**, **?**, **?**, **?**, **?**, **?**]) for the methods based on combination of rough sets and fuzzy sets.

In the first illustrative example, let us consider a decision system $\mathbb{A} = (U, A, F)$, where F is a decision function with values in the interval of reals $[0,1]$ representing a sample of fuzzy membership function. In the example we assume that the objects from the universe of U are perceived using conditional attributes from A, *i.e.*, objects with the same signatures relative to A are indiscernible. This uncertainty causes that the considered decision system may be in general inconsistent and the value of the decision function should be predicted on the basis of the generalized decision function ∂_A. Hence, approximation of the fuzzy membership function F by the fuzzy lower approximation $LOW(A,F)$ and the fuzzy upper approximation $UPP(A,F)$ of F can be defined as follows.

$$LOW(A,F)(x) = inf_{v \in \partial_A(x)} F(x)$$

and

$$UPP(A,F)(x) = sup_{v \in \partial_A(x)} F(x),$$

respectively, where $x \in U$.

Our second example is related to the case when fuzzy similarity relation is given instead of the crisp indiscernibility relation. Before presenting the second illustrative example of combining rough sets and fuzzy sets, let us consider a general framework for approximation within the fuzzy rough model. In this model, instead of an indiscernibility relation, we employ a fuzzy indiscernibility relation, that is, a function R mapping from $[0,1]\times[0,1]$ to $[0,1]$ that satisfies reflexivity $R(x,x)=1$ and symmetry $R(x,y)=R(y,x)$ for all $x,y\in[0,1]$. Furthermore, we consider fuzzy sets as the objects to be approximated, rather than classical sets.

The general scheme we will discuss is as follows:

$$LOW(A,R,I,F)(x) = inf_{y\in U}(I(R_A(x,y),F(y)),$$
$$UPP(A,R,T,F)(x) = sup_{y\in U}(T(R_A(x,y),F(y))),$$

where LOW and UPP represent approximation operators defined on a fuzzy set F over the universe U. These operators are based on a fuzzy similarity relation $R\subseteq U\times U\rightarrow[0,1]$, a set of attributes A over U (used in definition of R) and operations T,I mapping from $[0,1]\times[0,1]$ to $[0,1]$, which are extensions of the classical conjunction $\wedge$ and implication $\rightarrow$ operators, respectively.

This scheme of approximation, in particular operators in the scheme, will be more elaborated in this section.

Fuzzy rough sets have many relevant properties of classical rough set models, and have been subject to a rich theoretical development in the recent years (see, *e.g.*, [**?**, **?**, **?**, **?**, **?**, **?**, **?**, **?**, **?**, **?**, **?**, **?**, **?**]).

It is worthwhile mentioning that combination of rough sets and fuzzy sets leads to significant improvements of the quality of constructed classifiers in numerous real-life projects (see *e.g.* `https://www.isical.ac.in/~sankar/`).

Here, we would like to emphasize that this approach realizes some potential reasoning strategies for estimating membership degrees under uncertainty. Specifically, to estimate the membership degree of a previously unseen object x, we consider its neighbourhood $\{y\in Y : R(x,y)\}$ (relative to R) and employ a strategy to estimate x's membership based on the memberships of objects within this neighborhood. It is crucial to note that a vast array of strategies can be considered for this purpose, contingent on the chosen t-conorms, t-norms or implicators. This approach diverges from the strategy employed in methods like k-nearest neighbors (knn), as the optimal strategy is dataset-dependent. Consequently, learning the appropriate strategy may necessitate a parallel process to efficiently explore the solution space.

In the following subsection, we present an illustrative example that combines the rough set and fuzzy set approaches.

10.2.1.1 Example of combination of rough and fuzzy approaches

Let us consider a decision system $DS=(U,A,F)$ presented in Table 10.1 with real value decision F.

In Table 10.1 is presented a decision system $DS=(U,A,F)$, where $U=\{x_1,x_2,x_3,x_4,x_5,x_6\}$, $A=\{a_1,a_2,a_3,a_4\}$, $V_{a_1}=V_{a_2}=V_{a_3}=V_{a_4}=\{0,1,2\}$ and $V_F=\{0.0,0.2,0.5,0.7,0.9,1.0\}$. Let us assume, em e.g., that the value od decision expresses a

Table 10.1 Exemplary decision system with real value decision attribute

U	a_1	a_2	a_3	a_4	F
x_1	2	1	1	1	1.0
x_2	1	0	2	2	1.0
x_3	1	2	2	2	0.7
x_4	0	2	2	0	0.5
x_5	1	2	0	2	0.2
x_6	0	1	2	0	0.0

fuzzy concept 'a high risk of perceived situation on the basis of attributes from A'. The situations (objects) are perceived by conditional attributes. The value vectors of conditional attributes in rows of the table are results of perception of situations $x_1,\ldots,x_6$, respectively. By $inf_A(x_i)$ we denote A-signature of x_i defined by $\{(a_i,a_i(x_i)) :\ a_i \in A \ \&\ i \in \{1,\ldots,6\}\}$.

Let us now consider an exemplary fuzzy similarity relation R_A between pairs of objects from U defined by

$$R_A(x_i,x_j) = min_{k=1,\ldots,6}\left(1 - \frac{\mid a_k(x_i) - a_k(x_j) \mid}{range(a_k)}\right),$$

where $i,j \in \{1,\ldots,6\}$ and $range(a_k) = max_{v,v' \in V_{a_k}} \mid v - v' \mid$.

In Table 10.2 are presented values of R_A:

Table 10.2 Similarity relation R_A.

R_A	x_1	x_2	x_3	x_4	x_5	x_6
x_1	1	$\frac{1}{2}$	$\frac{1}{2}$	0	$\frac{1}{2}$	0
x_2	$\frac{1}{2}$	1	0	0	0	0
x_3	$\frac{1}{2}$	0	1	0	0	0
x_4	0	0	0	1	0	$\frac{1}{2}$
x_5	$\frac{1}{2}$	0	0	0	1	0
x_6	0	0	0	$\frac{1}{2}$	0	1

In this way, we constructed fuzzy indiscernibility relation R_A for the considered illustrative example. Now, we would like to illustrate how using this fuzzy relation we can approximate fuzzy decision function F from the example.

Let us recall that in the rough set approach for any subset $X \subseteq U$ of the universe of objects of a given information system $IS = (A,U)$ are defined its lower and upper approximations defined as follows.

$$LOW(A,X) = \{x \in U :\ [x]_A \subseteq X\}$$

and

$$UPP(A,X) = \{x \in U : [x]_A \cap X \neq \emptyset\},$$

respectively.

Observe that these definitions can be also formulated by:

$$x \in LOW(A,X) \Leftrightarrow [x]_A \subseteq X \Leftrightarrow (\forall y \in U)(xIND(A)y \Rightarrow y \in X)$$

and

$$x \in UPP(A,X) \Leftrightarrow [x]_A \cap X \neq \emptyset \Leftrightarrow (\exists y \in U)(xIND(A)y \;\&\; y \in X),$$

respectively.

For our finite domain U, the last formulas can be presented as follows.

$$(\forall y \in U)(xIND(A)y \Rightarrow y \in X) \Leftrightarrow \bigwedge_{i=1,\ldots,6} [\vee((\neg\, xINDx_i), x_i \in X)]$$

and

$$(\exists y \in U)(xIND(A)y \;\&\; y \in X) \Leftrightarrow \bigvee_{i=1,\ldots,6} [\wedge(xINDx_i, x_i \in X)],$$

respectively.

In the considered example we would like to approximate fuzzy decision function F using the fuzzy similarity relation R_A. Hence, the binary membership should be substituted to fuzzy membership in the real interval $[0.1]$, logical connective of disjunction, conjunction and negation should be substituted by appropriate fuzzy connectives with arguments in $[0,1]$. One possible choice is to substitute disjunction by operation *max* and conjunction by *min* and negation of the fuzzy membership function, say μ by $1-\mu$. Then we obtain the following definitions of the lower and upper approximation of F relative to R_A :

$$LOW(A,R,F)(x) = min_{i=1,\ldots,6}(max(1-R_A(x,x_i),F(x_i))),$$

and

$$UPP(A,R,F)(x) = max_{i=1,\ldots,6}(min(R_A(x,x_i),F(x_i))),$$

for $x \in U$.

In our example, we obtain, *e.g.*, the following values:

There are hybrid methods combining rough sets with methods using among others statistics, kernel functions, case-based reasoning, wavelets, EM method, independent component analysis, principal component analysis, rule induction and instance based learning, deep learning (see, *e.g.*, [**?**, **?**, **?**, **?**, **?**, **?**, **?**, **?**, **?**, **?**, **?**, **?**, **?**]).

To illustrate why the combination for different approaches may lead to improvement of the quality of the constructed classifiers, we consider a simple illustrative example where decision rules enhanced by additional information about distribution of objects matched by rules are used. Two situations of a test object matched by a decision rule r are illustrated in Figure 10.1.

In the first case, the test object tst_1 is matched by a decision rule r but it is 'far away' from other objects with decision $*$ matched by this rule. In the second case,

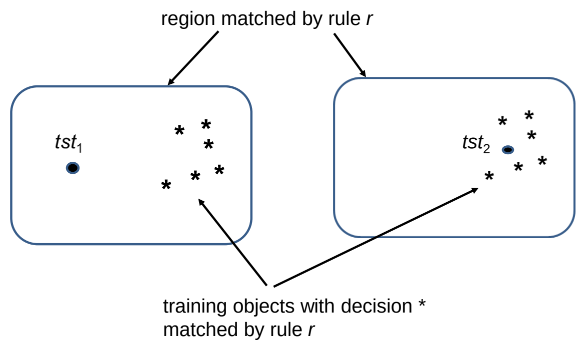


Fig. 10.1 In the standard approach, the same support is assigned to test objects tst_1 and tst_2 by a matching them rule *r*.

the test object tst_2 is matched by the same decision rule r but it is close to other objects with decision $*$ matched by this rule. In the traditional approach the support for the decision $*$ for the both objects tst_1 and Tst_2 measured relative to the training cases supporting the rule r, will be the sane. However, it is visible that the risk of failure in assigning the decision $*$ to tst_1 is much higher that in the case of tst_2. These two cases can be distinguished when together with rules will be used relevant for their classification neighbourhoods (see Figure 10.2). For example, the traditional support may be weighted and the weight assigned to tst_2 can be much higher than in the case of tst_1. In fact, in the first case it will be very risky to treat r as supporting the decision $*$ to tst_1 assuming that the confidence concerning of the relevance of its neighbourhood is high.

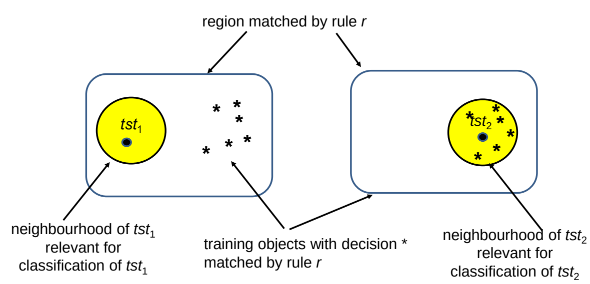


Fig. 10.2 Different supports assigned to test objects tst_1 and tst_2 by a matching them rule r combined with knn neighbourhoods.

For description of some exemplary advanced systems based on combination of the rule-based approach and instance-based approach the reader is referred to the RIONA system [**?**, **?**]and the RIONIDA system [**?**] designed for imbalanced data classification. The classification performed by RIONA on a given test object is based on rules induced only from the neighbourhood of the given test example consisting of some training examples. A small number of rules is enough to use when the lazy approach is applied. A different kind of rules is used in comparison to the commonly used rule-based approaches (where conditions are of the form: attribute equal to its specific value). In RIONA, more general rules are used based on grouping both numerical and symbolic values of attributes (conditions in these rules are of the form: attribute belongs to a set of values). In voting for the decision by rules matching the classified example, the aggregation of the support sets of such rules is used. RIONA constructs object neighbourhoods of the optimal size. The notion of similarity between objects is essential for RIONA for two purposes: construction of the neighbourhood for a given object and grouping values of attributes. The RIONA system was successfully generalised to RIONIDA for dealing with imbalanced data [**?**].

The relevance of the constructed neighbourhood depends on distribution of objects labelled by decisions in this neighbourhood depends. From this example it follows that combination of different approaches can help in designing more relevant for classification.

One should also note that the construction of classifiers is based on reasoning including not only deduction and induction. For example, often the reasoning leading to classifiers is expressed in natural language and next this description is gradually decomposed to be expressed by some mathematical machinery. Moreover, in this reasoning experience is used. Developing methods for such reasoning is one of the biggest challenge (see, *e.g.*, [**?**, **?**, **?**] and `http://people.seas.harvard.edu/~valiant/researchinterests.htm`).

Problem 10.5. Verify the basic properties of the fuzzy approximations of fuzzy membership function considered in our first illustrative example such as: $LOW(A,F)(x) \leq F(x) \leq UPP(A,F)$.

Problem 10.6. Extend our first example by proposing inductive extension of the approximation of F for signatures of objects which are not occurring in the decision system $()U,A,F)$.

Problem 10.7. [**?**] Develop heuristics for generation decision rules with the right hand sides of the form $d \notin V$, where the cardinality V is 'small'.

Problem 10.8. [**?**] Develop heuristics for generation decision rules using a combination of approaches based on rough sets and fuzzy sets.

Problem 10.9. Research problem.
Develop heuristics for selection of the relevant implication and t-norm in the case of rough fuzzy approach.

Problem 10.10. [**?**] Develop heuristics for generation decision rule using a combination of approaches based on rough sets and knn.

10.3 Rough sets and conflict analysis

Since ancient times, conflict analysis and conflict resolution have played an important role in many areas. Nowadays, one can observe the growing research interest in developing of systems supporting conflicts analysis and negotiations for their resolution in, *e.g.*, business, governmental, political and lawsuits disputes, labor-management negotiations, military operations. It was observed (see, *e.g.*, [**?**]) that the concept of conflict has happened to be one of the fundamental concepts of logic. However, due to necessity of carrying out negotiations in natural language still new reasoning methods aiming at supporting negotiations or reaching consensus are under development or. Among examples of important research directions in logic related to conflict analysis are those based on dialogue (see *e.g.*, [**?**]) (especially in human-computer interaction [**?**]) or on paraconsistent logic [**?**]. Quite many mathematical formal models of conflict situations and methods for negotiations as well as reaching of consensus have been proposed and studied (see, e.g., [**?**, **?**]).

Certainly, one can use game theory in developing a mathematical model of conflict situations. Many other mathematical tools from, *e.g.*, graph theory, topology or differential equations have been also used to that purpose. However, there is no, as yet, “universal" theory of conflicts and mathematical models of conflict situations are strongly domain dependent.

In this section, we outline one of this models proposed by Professor Zdzisław Pawlak [**?**, **?**, **?**, **?**, **?**, **?**]. This model was further developed in several directions [**?**, **?**].

The considered model is simple enough for easy computer implementation and seems adequate for many real life applications but to this end more research is needed.

In this model it is assumed that there is a given finite set of agents $Ag = \{ag_1, \ldots, ag_m\}$ representing sides of the conflicts, *e.g.*, countries, customers or managers. There are some issues (from a given finite set of issues) on which the agents can vote and the result of vote of $ag \in Ag$ on a given issue is in the set $\{-1,0,1\}$[1]. The values $-1,0,1$ are representing that the agent is *against*, *neutral* or *favorable*, respectively on the issue. Hence, the information about the agents can be represented by a simplified information system (U,A), where $U = Ag$ and attributes $a : Ag \to \{-1,0,1\}$ from A correspond to the issues and the value of attribute a on $ag \in Ag$, *i.e.*, $a(ag)$ is equal to the result of voting of ag on the issue a.

We will use an illustrative example of the Middle East conflict considered in [**?**, **?**]. The example does not necessarily reflect present-day situation in this region but is used here only as an illustration of the basic ideas considered here. In this example $Ag = \{1,2,3,4,5,6\}$ and $A = \{a,b,c,d,e\}$, where elements of Ag are abbreviations for countries Israel, Egypt, Palestinians, Jordan, Syria, Saudi Arabia, respectively and the issues from A have the following interpretation:

- a – autonomous Palestinian state on the West Bank and Gaza,
- b – Israeli military outpost along the Jordan River,
- c – Israeli retains East Jerusalem,
- d – Israeli military outposts on the Golan Heights,

[1] We will also write - and + instead of -1 and 1, respectively.

- *e* Arab countries grant citizenship to Palestinians who choose to remain within their borders.

The relationship of each agent to a specific issue is shown in Table 10.3.

Table 10.3 Data table for the Middle East conflict

U	a	b	c	d	e
1	$-$	$+$	$+$	$+$	$+$
2	$+$	0	$-$	$-$	$-$
3	$+$	$-$	$-$	$-$	0
4	0	$-$	$-$	0	$-$
5	$+$	$-$	$-$	$-$	$-$
6	0	$+$	$-$	0	$+$

In the table the attitude of six nations of the Middle East region to the above issues is presented: - means, that an agent is against, + means favorable and 0 neutral toward the issue.

Each row of the table characterizes uniquely an agent, by his approach to the disputed issues.

In conflict analysis, primarily we are interested in finding the relationship between agents taking part in the dispute, and investigate what can be done in order to improve the relationship between agents, or in other words how the conflict can be resolved.

In the mentioned above generations of the discussed model there are considered richer models representing, *e.g.*, reasons why agents are voting in a particular way (see, *e.g.*, [**?**]).

In the discussed model, three basic binary relations are used on the universe of agents: *conflict, neutrality* and *alliance*. To define them the following auxiliary function is used:

$$\phi_a(ag,ag') = \begin{cases} 1, & \text{if } a(ag)a(ag') = 1 \text{ or } \; ag = ag', \\ 0, & \text{if } a(ag)a(ag') = 0 \text{ and } ag \neq ag', \\ -1, & \text{if } a(ag)a(ag') = -1. \end{cases}$$

This means that, if $\phi_a(ag,ag') = 1$, agents ag and ag' have the same opinion about issue a (are *allied* on a); $\phi_a(ag,ag') = 0$ means that at least one agent ag or ag' has neutral approach to issue a (is *neutral* on a), and $\phi_a(ag,ag') = -1$, means that the two agents have different opinions about issue a (are in *conflict* on a).

Now can be defined three basic relations R_a^+, R_a^0 and R_a^- over U^2 called *alliance, neutrality* and *conflict* relations respectively:

$$R_a^+(ag,ag') \text{ iff } \phi_a(ag,ag') = 1, \tag{10.5}$$

$$R_a^0(ag,ag') \text{ iff } \phi_a(ag,ag') = 0, \tag{10.6}$$

$$R_a^-(ag, ag') \text{ iff } \phi_a(ag, ag') = -1. \tag{10.7}$$

For example, in the Middle East conflict Egypt, Palestinians and Syria are allied on issue a (autonomous Palestinian state on the West Bank and Gaza), Jordan and Saudi Arabia are neutral to this issue whereas, Israel and Egypt, Israel and Palestinians, and Israel and Syria are in conflict about this issue.

The alliance relation has the following properties:

(i) $R_a^+(ag, ag)$,
(ii) $R_a^+(ag, ag')$ implies $R_a^+(ag', ag)$,
(iii) $R_a^+(ag, ag')$ and $R_a^+(ag', ag")$ implies $R_a^+(ag, ag")$.

Hence, for any $a \in A$, R_a is an equivalence relation. Each equivalence class of alliance relation will be called *coalition* on a[2]. Let us note that the condition (iii) can be expressed as "friend of my friend is my friend".

For the conflict relation we have the following properties:

(iv) non $R_a^-(ag, ag)$,
(v) $R_a^-(ag, ag')$ implies $R_a^-(ag', ag)$,
(vi) $R_a^-(ag, ag')$ and $R_a^-(ag', ag")$ implies $R_a^+(ag, ag")$,
(vii) $R_a^-(ag, ag')$ and $R_a^+(ag', ag")$ implies $R_a^-(ag, ag")$.

Conditions (vi) and (vii) refer to well known sayings "enemy of my enemy is my friend" and "friend of my enemy is my enemy".

For the neutrality relation we have:

(viii) non $R_a^0(ag, ag)$,
(ix) $R_a^0(ag, ag') = R_a^0(ag', ag)$ (symmetry).

Let us note that in the conflict and neutrality relations there are no coalitions – they are defined only for alliance relation.

These relations R_a^+, R_a^0, R_a^- are illustrated by a *conflict* graph (see Figure 10.3).

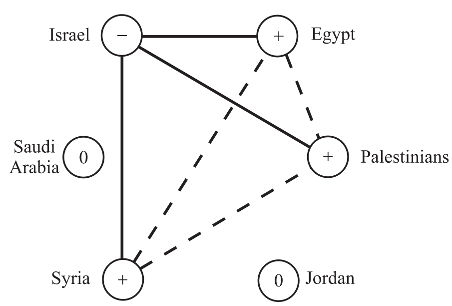


Fig. 10.3 Conflict graph for attribute a.

Nodes of the graph are labelled by agents, whereas branches of the graph represent relations between agents. Besides, opinion of agents $(0, -, +)$ on the disputed

[2] Coalitions which are singletons are called degenerated coalitions.

issue is shown on each node. Solid lines denote conflicts, dotted line – alliance, and neutrality, for simplicity, is not shown explicitly in the graph.

Any conflict graph represents a set of facts. For example, the set of facts represented by the graph in Figure 10.3 consists of the following facts:
R_a^-(Israel, Egypt), R_a^-(Israel, Palestinians), R_a^-(Israel, Syria),
R_a^+(Egypt, Syria), R_a^+(Egypt, Palestinians), R_a^+(Syria, Palestinians),
R_a^0(Saudi Arabia, ag), R_a^0(Jordan, ag), $R_a^0(ag, ag)$
for $ag \in \{$Israel, Egypt, Palestinians, Jordan, Syria, Saudi Arabia$\}$.

If a is conflicting attribute, *i.e.*, $R_a^- \neq \emptyset$, then all agents are divided into two coalitions (blocks) $X_a^+ = \{ag \in U : a(ag) = 1\}$, $X_a^- = \{ag \in U : a(ag) = -1\}$. Any two agents belonging to two different coalitions are in conflict, and the remaining (if any) agents are neutral to the issue a.

The graph shown in Figure 10.3 can be presented as a *coalition* graph (see Figure 10.4).

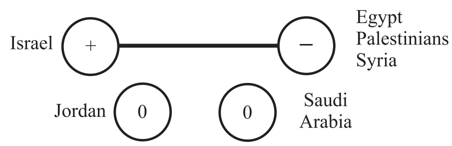


Fig. 10.4 Coalition graph for attribute a.

Analysis of coalition graphs corresponding to various attributes leads to a better understanding of the structure of the Middle East conflict and allows us to infer facts important for negotiations between agents. For example, the attribute c induces partition in which Israel is in conflict with all remaining agents, whereas attribute e leads to alliance of Israel and Saudi Arabia against Egypt, Jordan and Syria with Palestinians being neutral.

An important measure of *conflict degree* of issue a can be defined as follows.

$$Con(a) = \frac{|X_a^+| \cdot |X_a^-|}{[\frac{n}{2}] \cdot (n - [\frac{n}{2}])} = \frac{|R_a^-|}{[\frac{n}{2}] \cdot (n - [\frac{n}{2}])}, \tag{10.8}$$

where $|X|$ denotes cardinality of the set X, $n \geq 2$ is the number of agents involved in the conflict (*i.e.*, the number of nodes of the conflict graph) and $[\frac{n}{2}]$ denotes whole part of $\frac{n}{2}$.

For example, $Con(b) = 2/3$, $Con(c) = 5/9$.

Some interesting hints for negotiations can be obtained by analysing changes of structures of conflict graphs for particular issues caused when agents are changing their votes, *e.g.*, from neutral to alliance or negative. This analysis may concern changes of coalitions caused by changes of votes (see Problem 10.18).

Another interesting problem of conflict analysis is related to *incomplete conflict graphs* for particular issues, *i.e.*, conflict graphs with unknown votes (marked by *) of some agents for a given issue. Then, one can consider *consistent extensions* of such graphs in which unknown votes are replaced by votes from $\{+, -, 0\}$ in such a way that obtained graphs are consistent with properties (i)–(vii). One of these problems, is related to computing of *the conflict degree uncertainty* related to a given incomplete coalition graph. This uncertainty of the conflict degree is defined as the maximal difference between conflict degrees of maximal consistent extensions of a given incomplete conflict graph (see Problem 10.19).

The degree of conflict for a given issue can be generalised to a *tension* on the set of attributes $\emptyset \neq B \subseteq A$ as follows.

$$Con(B) = \frac{\sum_{a \in B} Con(a)}{|B|}. \tag{10.9}$$

Tension for the Middle East Conflict is $Con(A) \cong 0.51$.

Another issue related to agents involved in conflict is their dissimilarity. For agents ag, ag' from Table 10.3 their dissimilarity can be defined by

$$\rho_B(ag, ag') = \frac{|\delta_B(ag, ag')|}{|A|}, \tag{10.10}$$

where $B \subseteq A$ and $\delta_B(ag, ag') = \{a \in B : a(ag) \neq a(ag')\}$.

Before we start negotiations we have to understand better the relationship between different issues being discussed. To this end onme can define a concept of a tr-*reduct* of attributes, where $tr \in (0, 0.5)$ is a given threshold. For example, one can restrict analysis of conflicts defined by dissimilarities of agents, *e.g.*, one can drop a set of 'not very conflicting' issues. In this case, the concept of reduct relative to a given threshold tr may be used.

A tr-reduct of A (see Table 10.3) is any minimal subset B of A satisfying for any ag, ag' the following condition:

$$\rho_B(ag, ag') \geq \rho_A(ag, ag') - tr.$$

Hence, if B is a tr-reduct of A then $A \setminus B$ is a set of 'not very conflicting' issues (relative to tr).

By computing reducts relative to a given threshold (see Problem 10.21) one can obtain sets issues on which negotiations can be concentrated.

More details on conflict analysis based on rough sets the reader can find in the literature cited at the beginning of this section.

At the end of our short introduction to conflict analysis based on rough sets we would like to add some comments related to further research. It is worthwhile mentioning that the further development of methods for negotiations toward conflict resolving will require, in particular development of new reasoning methods on which dialogue can be based (for some attempts in this direction see, *e.g.*, [**?**]). In particular, the rather primitive dialogues between different classifiers from their ensembles used so far in Machine Learning will be substituted by more advanced dialogues between agents inducing classifiers from local sources (see, *e.g.*, [**?**, **?**]). The new methods of reasoning should support dialogues of intelligent systems systems with human experts carried out in (fragments) of natural language [**?**]. One should also consider that the reasoning methods of intelligent systems supporting decisions related to complex situations of the real physical world should be based on a relevant computing model, different from the classical one. An attempt to develop such a model is discussed in the last part of the book related to Interactive Granular Computing (IGrC).

Problem 10.11. Please check that for any a, $R_a \subseteq U \times U$ is an equivalence relation.

Problem 10.12. Calculate all coalitions of R_a for any attribute (issue) a from Table 10.3.

Problem 10.13. Please check that for any $a \in A$, $\{R_a^+, R_a^0, R_a^-\}$ is a partition of $U \times U$.

Problem 10.14. Please draw the conflict graph for any attribute $a \in A$ from Table 10.3.

Problem 10.15. Check that if a is a conflicting attribute, then $R_a^-(x,y)$ *iff* $x \in X_a^+$ and $y \in X_a^-$ or $y \in X_a^+$ and $x \in X_a^-$, for every $x, y \in U$.

Problem 10.16. Please draw the coalition graph for any attribute $a \in A$ from Table 10.3.

Problem 10.17. Please show that conflict degree belongs to the interval [0,1].

Problem 10.18. How the structure of the conflict graph for the issue a in Table 10.3 will change when Jordan would change the neutrality vote to this issue to support?

Problem 10.19. Design an incomplete conflict graph without consistent extension.

Problem 10.20. Design heuristics for estimation of the conflict degree uncertainty for a given incomplete coalition graph.

Problem 10.21. Design heuristics based on Boolean reasoning for computing reducts relative to a given threshold for a given data table describing conflict situation between agents from a given set of agents having their votes on issues from a given set of issues.

References

1. Albanese, A., Pal, S.K., Petrosino, A.: Rough sets, kernel set, and spatiotemporal outlier detection. IEEE Transactions on Knowledge and Data Engineering **26**, 194–207 (2014). doi:10.1109/TKDE.2012.234
2. An, S., Shi, H., Hu, Q., Li, X., Dang, J.: Fuzzy rough regression with application to wind speed prediction. Information Sciences **282**, 388–400 (2014). doi:10.1016/j.ins.2014.03.090
3. Aruda, I.A.: Aspects of the historical development of paraconsistent logic. In: G. Priest, R. Routley, J. Norman (eds.) Paraconsistent Logic: Essays on the Inconsistent, pp. 99–129. Philosophia Verlag, München (1989). doi:10.2307/j.ctv2x8v8c7

4. Bashir, Z., Mahnaz, S., Malik, M.G.A.: Conflict resolution using game theory and rough sets. International Journal of Intelligent Systems **36**(1), 237–259 (2021). doi:10.1002/int.22298
5. Bello, R., Falcón, R., Pedrycz, W.: Granular Computing: At the Junction of Rough Sets and Fuzzy Sets, *Studies in Fuzziness and Soft Computing*, vol. 234. Springer, Heidelberg (2010). doi:10.1007/978-3-540-76973-6
6. Campagner, A., Ciucci, D., Denœux, T.: Belief functions and rough sets: Survey and new insights. Journal of Approximate Reasoning **143**, 192–215 (2022). doi:10.1016/j.ijar.2022.01.011
7. Casti, J.L.: Alternate Realities? Mathematical Models of Nature and Man. John Wiley & Sons, New York, NY (1989)
8. Chakraborty, D., Pal, S.: Granular Video Computing Granular Video Computing: with Rough Sets, Deep Learning and in IoT. World Scientific, Singapore (2021). doi:10.1142/12013
9. Chakraborty, M.K., Banerjee, M.: Rough dialogue and implication lattices. Fundamenta Informaticae **75**(1-4), 123–139 (2007)
10. Chakraborty, M.K., Samanta, P.: Fuzzy sets and rough sets: A mathematical narrative. In: T. Som, O. Castillo, A.K. Tiwari, S. Shreevastava (eds.) Fuzzy, Rough and Intuitionistic Fuzzy Set Approaches for Data Handling. Theory and Applications, Forum for Interdisciplinary Mathematics, pp. 1–21. Springer Nature, Singapore (2023). doi:10.1007/978-3-540-73723-0_14
11. Coombs, C.H., Avruin, G.S.: The Structure of Conflicts. Lawrence Erlbaum, London (1988)
12. Cornelis, C.: Hybridization of Fuzzy Sets and Rough Sets: Achievements and Opportunities. In: Proceedings of FedCSIS 2022, pp. 7–14 (2022). doi:0.15439/2022F302
13. Cornelis, C., Bollaert, H.: Combination of fuzzy sets and rough sets for machine learning purposes (tutorial lecture - extended abstract). In: M. Ganzha, L.A. Maciaszek, M. Paprzycki, D. Slezak (eds.) Proceedings of the 18th Conference on Computer Science and Intelligence Systems, FedCSIS 2023, Warsaw, Poland, September 17-20, 2023, *Annals of Computer Science and Information Systems*, vol. 35, p. 67 (2023). doi:0.15439/2023F0001
14. Cornelis, C., Cock, M.D., Radzikowska, A.M.: Vaguely quantified rough sets. In: A. An, J. Stefanowski, S. Ramanna, C.J. Butz, W. Pedrycz, G. Wang (eds.) Rough Sets, Fuzzy Sets, Data Mining and Granular Computing, 11th International Conference, RSFDGrC 2007, Toronto, Canada, May 14-16, 2007, Proceedings, *Lecture Notes in Computer Science*, vol. 4482, pp. 87–94. Springer (2007). doi:10.1007/978-3-540-72530-5_10
15. Deja, R.: Conflict analysis, rough set methods and applications. In: L. Polkowski, T.Y. Lin, S. Tsumoto (eds.) Rough Set Methods and Applications: New Developments in Knowledge Discovery in Information Systems, *Studies in Fuzziness and Soft Computing*, vol. 56, pp. 491–520. Physica-Verlag, Heidelberg (2000). doi:10.1007/978-3-7908-1840-6
16. Deja, R.: Conflict analysis. International Journal of Intelligent Systems **17**(2), 235–253 (2002). doi:10.1002/int.10019
17. Deja, R., Skowron, A.: On some conflict models and conflict resolution. Romanian Journal of Information Science and Technology **5**(1-2), 69–82 (2002)
18. Dubois, D., Prade, H.: Rough fuzzy sets and fuzzy rough sets. International Journal of General Systems **17**, 191–209 (1990). doi:10.1080/03081079008935107
19. Duong, T.H., Nguyen, N.T., Kozierkiewicz-Hetmanska, A., Jo, G.: Fuzzy ontology integration using consensus to solve conflicts on concept level. In: N.T. Nguyen, B. Trawinski, J.J. Jung (eds.) New Challenges for Intelligent Information and Database Systems [original works presented during a poster session organized within the 3rd Asian Conference on Intelligent Information and Database Systems, 20-22 April 2011, Daegu, Korea], *Studies in Computational Intelligence*, vol. 351, pp. 33–42. Springer (2011). doi:10.1007/978-3-642-19953-0_4
20. Farinas del Cerro, L., Prade, H.: Rough sets, twofold fuzzy sets and modal logic - Fuzziness in indiscernibility and partial information. In: A.D. Nola, A. Ventre (eds.) The Mathematics of Fuzzy Systems, pp. 103–120. Verlag TUV Rheinland, Köln (1986)
21. Ganivada, A., Ray, S., Pal, S.: Fuzzy rough sets, and a granular neural network for unsupervised feature selection. Neural Networks **48**, 91–108 (2013). doi:10.1016/j.neunet.2013.07.008

22. Ganivada, A., Ray, S.S., Pal, S.: Fuzzy rough granular self-organizing map and fuzzy rough entropy. Theoretical Computer Science **466**, 37–63 (2012). doi:10.1016/j.tcs.2012.08.021
23. Golunska, D., Kacprzyk, J.: A consensus reaching support system for multi-criteria decision making problems. In: G.D. Tré, P. Grzegorzewski, J. Kacprzyk, J.W. Owsinski, W. Penczek, S. Zadrozny (eds.) Challenging Problems and Solutions in Intelligent Systems, *Studies in Computational Intelligence*, vol. 634, pp. 219–235. Springer (2016). doi:10.1007/978-3-319-30165-5_11
24. Góra, G.: Combining instance-based learning and rule-based methods for imbalanced data. Phd thesis, University of Warsaw, Warsaw (2022). https://www.mimuw.edu.pl/sites/default/files/gora_grzegorz_rozprawa_doktorska.pdf
25. Góra, G., Wojna, A.: RIONA: A new classification system combining rule induction and instance-based learning. Fundamenta Informaticae **51**(4), 369–390 (2002)
26. Hadrani, A., Guennoun, K., Saadane, R., Wahbi, M.: Fuzzy rough sets: Survey and proposal of an enhanced knowledge representation model based on automatic noisy sample detection. Cognitive Systems Research **64**, 37–56 (2020). doi:10.1016/j.cogsys.2020.05.001
27. Hart, H.: Structures of influence and cooperation-conflict. Interactions **1**, 141–162 (1974). doi:10.1080/03050627408434397
28. Herrera-Viedma, E., García-Lapresta, J.L., Kacprzyk, J., Fedrizzi, M., Nurmi, H., Zadrozny, S. (eds.): Consensual Processes, *Studies in Fuzziness and Soft Computing*, vol. 267. Springer (2011). doi:10.1007/978-3-642-20533-0
29. Hu, Q., Yu, D., Pedrycz, W., Chen, D.: Kernelized fuzzy rough sets and their applications. IEEE Transactions on Knowledge and Data Engineering **23**, 1649–1667 (2011). doi:10.1109/TKDE.2010.260
30. Inuiguchi, M., Wu, W.Z., Cornelis, C., Verbiest, N.: Fuzzy-rough hybridization. In: J. Kacprzyk, W. Pedrycz (eds.) Springer Handbook of Computational Intelligence, pp. 425–451. Springer, Heidelberg (2015). doi:doi.org/10.1007/978-3-662-43505-2_26
31. Jensen, R., Shen, Q.: Computational Intelligence and Feature Selection: Rough and Fuzzy Approaches. IEEE Press Series on Cmputationa Intelligence. IEEE Press and John Wiley & Sons, Hoboken, NJ (2008)
32. Joshi, M., Lingras, P., Rao, C.R.: Correlating fuzzy and rough clustering. Fundamenta Informaticae **115**(2-3), 233–246 (2012). doi:10.3233/FI-2012-652
33. Kacprzyk, J., Merigó, J.M., Nurmi, H., Zadrozny, S.: Multi-agent systems and voting: How similar are voting procedures. In: M. Lesot, S.M. Vieira, M.Z. Reformat, J.P. Carvalho, A. Wilbik, B. Bouchon-Meunier, R.R. Yager (eds.) Information Processing and Management of Uncertainty in Knowledge-Based Systems - 18th International Conference, IPMU 2020, Lisbon, Portugal, June 15-19, 2020, Proceedings, Part I, *Communications in Computer and Information Science*, vol. 1237, pp. 172–184. Springer (2020). doi:10.1007/978-3-030-50146-4_14
34. Kacprzyk, J., Nurmi, H., Zadrozny, S.: Towards a comprehensive similarity analysis of voting procedures using rough sets and similarity measures. In: A. Skowron, Z. Suraj (eds.) Rough Sets and Intelligent Systems - Professor Zdzisław Pawlak in Memoriam - Volume 1, *Intelligent Systems Reference Library*, vol. 42, pp. 359–380. Springer (2013). doi:10.1007/978-3-642-30344-9_13
35. Kacprzyk, J., Zadrozny, S.: Reaching consensus in a group of agents: Supporting a moderator run process via linguistic summaries. In: M. Collan, J. Kacprzyk (eds.) Soft Computing Applications for Group Decision-making and Consensus Modeling, *Studies in Fuzziness and Soft Computing*, vol. 357, p. 465—485. Springer, Cham (2018). doi:10.1007/978-3-319-60207-3
36. Kacprzyk, J., Zadrozny, S., Fedrizzi, M., Nurmi, H.: On group decision making, consensus reaching, voting and voting paradoxes under fuzzy preferences and a fuzzy majority: A survey and some perspectives. In: H.B. Sola, F. Herrera, J. Montero (eds.) Fuzzy Sets and Their Extensions: Representation, Aggregation and Models - Intelligent Systems from Decision Making to Data Mining, Web Intelligence and Computer Vision, *Studies in Fuzziness and Soft Computing*, vol. 220, pp. 263–295. Springer (2008). doi:10.1007/978-3-540-73723-0_14

37. Klein, M.: Conflict resolution in cooperative design. International Journal for AI in Engineering **4**, 168–180 (1990). doi:10.1016/0954-1810(89)90013-7
38. Kowalski, R.: A logic-based approach to conflict resolution. Report, Department of Computing, Imperial College pp. 1–28 (2003). URL http://www.doc.ic.ac.uk/\char126rak/papers/conflictresolution.pdf
39. Kraus, S.: Strategic Negotiations in Multiagent Environments. The MIT Press, Cambridge, MA (2001)
40. Lai, G., Li, C., Sycara, K., Giampapa, J.A.: Literature review on multi-attribute negotiations. Technical Report CMU-RI-TR-04-66 pp. 1–35 (2004). URL https://www.researchgate.net/publication/246704687_Literature_review_on_multi-attribute_negotiations
41. Lang, G., Luo, J., Yao, Y.: Three-way conflict analysis: A unification of models based on rough sets and formal concept analysis. Knowledge Based Systems **194**, 105,556 (2020). doi:10.1016/j.knosys.2020.105556. URL https://doi.org/10.1016/j.knosys.2020.105556
42. Li, Y., Shiu, S.C.K., Pal, S.K., Liu, J.N.K.: A rough set-based case-based reasoner for text categorization. International Journal of Approximate Reasoning **41**(2), 229–255 (2006). doi:10.1016/j.ijar.2005.06.019
43. Lingras, P., Jensen, R.: Survey of rough and fuzzy hybridization. In: FUZZ-IEEE 2007, IEEE International Conference on Fuzzy Systems, Imperial College, London, UK, 23-26 July, 2007, Proceedings, pp. 1–6. IEEE (2007). doi:10.1109/FUZZY.2007.4295352
44. Lingras, P., Peters, G.: Rough clustering. Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery **1**(1), 64–72 (2011). doi:10.1002/widm.16
45. Maeda, Y., Senoo, K., Tanaka, H.: Interval density function in conflict analysis. In: A. Skowron, S. Ohsuga, N. Zhong (eds.) Proceedings of the 7-th International Workshop on Rough Sets, Fuzzy Sets, Data Mining, and Granular-Soft Computing (RSFDGrC'1999), Yamaguchi, November 9-11, 1999, *Lecture Notes in Artificial Intelligence*, vol. 1711, pp. 382–389. Springer-Verlag, Heidelberg (1999). doi:10.1007/978-3-540-48061-7_46
46. Maji, P., Pal, S.: RFCM: A hybrid clustering algorithm using rough and fuzzy sets. Fundamenta Informaticae **80**(4), 477–498 (2007). doi:10.5555/2367421.2367428
47. Maji, P., Pal, S.: Rough set based generalized fuzzy c-means algorithm and quantitative indices. IEEE Transactions on Systems, Man and Cyberneticss. Part B: Cybernetics **37**(6), 1529–1540 (2007). doi:10.1109/TSMCB.2007.906578
48. Maji, P., Pal, S.: Fuzzy-rough sets for information measures and selection of relevant genes from microarray data. IEEE Transactions on Systems, Man and Cyberneticss. Part B: Cybernetics **40**(3), 741–752 (2010). doi:10.1109/TSMCB.2009.2028433
49. Maji, P., Pal, S.K.: Rough-Fuzzy Pattern Recognition: Application in Bioinformatics and Medical Imaging. Wiley Series in Bioinformatics. John Wiley & Sons, Hoboken, NJ (2012)
50. Marszał-Paszek, B., Paszek, P.: Classifiers based on nondeterministic decision rules. In: A. Skowron, Z. Suraj (eds.) Rough Sets and Intelligent Systems - Professor Zdzisław Pawlak in Memoriam (vol. 2), *Intelligent Systems Reference Library*, vol. 43, pp. 445–454. Springer, Heidelberg (2013). doi:10.1007/978-3-642-30341-8_23
51. Mehera, S.K., Pal, S.K.: Rough-wavelet granular space and classification of multispectral remote sensing image. Applied Soft Computing **11**, 5662–5673 (2011). doi:10.1016/j.asoc.2011.03.027
52. Mitra, P., Pal, S., Siddiqi, M.A.: Nonconvex clustering using expectation maximization algorithm with rough set initialization. Pattern Recognition Letters **24**, 863–873 (2003). doi:10.1016/S0167-8655(02)00198-8
53. Nakamura, A.: Conflict logic with degrees. In: S.K. Pal, A. Skowron (eds.) Rough Fuzzy Hybridization: A New Trend in Decision-Making, pp. 136–150. Springer-Verlag, Singapore (1999)
54. Nguyen, H.S.: Approximate boolean reasoning: Foundations and applications in data mining. Transactions on Rough Sets: Journal Subline **5**, 344–523 (2006). doi:10.1007/11847465_16. (LNCS 4100; Springer, Heidelberg)

55. Nguyen, N.T.: Using consensus methods for solving conflicts of data in distributed systems. In: V. Hlavác, K.G. Jeffery, J. Wiedermann (eds.) SOFSEM 2000: Theory and Practice of Informatics, 27th Conference on Current Trends in Theory and Practice of Informatics, Milovy, Czech Republic, November 25 - December 2, 2000, Proceedings, *Lecture Notes in Computer Science*, vol. 1963, pp. 411–419. Springer (2000). doi:10.1007/3-540-44411-4_30
56. Nguyen, N.T.: Processing collective knowledge - conflict resolution and integration aspects - (keynote speech). In: K. Saeed, V. Snásel (eds.) Computer Information Systems and Industrial Management - 13th IFIP TC8 International Conference, CISIM 2014, Ho Chi Minh City, Vietnam, November 5-7, 2014. Proceedings, *Lecture Notes in Computer Science*, vol. 8838, pp. 1–4. Springer (2014). doi:10.1007/978-3-662-45237-0_1
57. Nguyen, N.T., Sobecki, J.: Consensus versus conflicts - methodology and applications. In: G. Wang, Q. Liu, Y. Yao, A. Skowron (eds.) Rough Sets, Fuzzy Sets, Data Mining, and Granular Computing, 9th International Conference, RSFDGrC 2003, Chongqing, China, May 26-29, 2003, Proceedings, *Lecture Notes in Computer Science*, vol. 2639, pp. 565–572. Springer (2003). doi:10.1007/3-540-39205-X_92
58. Nurmi, H., Kacprzyk, J., Fedrizzi, M.: Probabilistic, fuzzy and rough concepts in social choice. European Journal of Operational Research **95**(2), 264–277 (1996). doi:10.1016/0377-2217(95)00295-2
59. Pal, S.: Computational theory of perception (CTP), rough-fuzzy uncertainty analysis and mining in bioinformatics and web intelligence: A unified framework. Transactions on Rough Sets XI: Journal Subline **11**, 106–129 (2010). doi:10.1007/978-3-642-11479-3_7. (LNCS 5946; Springer, Heidelberg)
60. Pal, S.: Rough Set and Deep Learning: Some Concepts. Academia Letters (Article 1849), 1–6 (July 2021). doi:10.20935/AL1849
61. Pal, S., Meher, S., Dutta, S.: Class-dependent rough-fuzzy granular space, dispersion index and classification. Pattern Recognition **45**, 2690–2707 (2012). doi:10.1016/j.patcog.2011.12.027
62. Pal, S., Shiu, S.: Foundations of Soft Case-Based Reasoning. Wiley, Hoboken, NJ (2004)
63. Pal, S.K.: Soft data mining, computational theory of perceptions, and rough-fuzzy approach. Information Sciences **163**(1-3), 5–12 (2004). doi:10.1016/j.ins.2003.03.014
64. Pal, S.K., Mitra, P.: Multispectral image segmentation using the rough-set-initialized EM algorithm. IEEE Transactions on Geoscience and Remote Sensing **40**, 2495–2501 (2002). doi:10.1109/TGRS.2002.803716
65. Pal, S.K., Mitra, P.: Case generation using rough sets with fuzzy representation. IEEE Trans. Knowledge and Data Engineering **16**(3), 292–300 (2004). doi:10.1109/TKDE.2003.1262181
66. Pal, S.K., Mitra, P.: Pattern Recognition Algorithms for Data Mining. CRC Press, Boca Raton, Florida (2004)
67. Pal, S.K., Peters, J.F. (eds.): Rough Fuzzy Image Analysis Foundations and Methodologies. Chapman & Hall/CRC, Boca Raton, Fl (2010)
68. Pal, S.K., Skowron, A. (eds.): Rough Fuzzy Hybridization: A New Trend in Decision-Making. Springer-Verlag, Singapore (1999)
69. Pawlak, Z.: Rough sets and fuzzy sets. Fuzzy Sets and Systems **17**, 99–102 (1985). doi:10.1016/S0165-0114(85)80029-4
70. Pawlak, Z.: O konfliktach (On Conflicts). PWN, Warszawa (1987). (in Polish)
71. Pawlak, Z.: Anatomy of conflict. Bulletin of the European Association for Theoretical Computer Science (EATCS) **50**, 234–247 (1993)
72. Pawlak, Z.: An inquiry into anatomy of conflicts. Journal of Information Sciences **109**, 65–78 (1998). doi:10.1016/S0020-0255(97)10072-X
73. Pawlak, Z.: Some remarks on conflict analysis. European Journal of Operational Research **166**, 649–654 (2005)
74. Pawlak, Z.: Conflicts and negotations. In: G. Wang, J.F. Peters, A. Skowron, Y. Yao (eds.) Rough Sets and Knowledge Technology, First International Conference, RSKT 2006, Chongqing, China, July 24-26, 2006, Proceedings, *Lecture Notes in Computer Science*, vol. 4062, pp. 12–27. Springer (2006). doi:10.1007/11795131_2

75. Pawlak, Z., Skowron, A.: Rough sets and conflict analysis. In: J. Lu, D. Ruan, G. Zhang (eds.) E-Service Intelligence Methodologies, Technologies and Applications, *Studies in Computational Intelligence*, vol. 37, pp. 35–74. Springer, Heidelberg (2006). doi:10.1007/978-3-540-37017-8
76. Pearl, J.: Causal inference in statistics: An overview. Statistics Surveys **3**, 96–146 (2009). doi:10.1214/09-SS057
77. Pedrycz, W.: An Introduction to Computing with Fuzzy Sets. Analysis, Design, and Applications, *Intelligent Systems Reference Library*, vol. 190. Springer Nature, Cham (2021). doi:10.1007/978-3-030-52800-3
78. Peters, G., Crespo, F., Lingras, P., Weber, R.: Soft clustering - fuzzy and rough approaches and their extensions and derivatives. International Journal of Approximate Reasoning **54**(2), 307–322 (2013). doi:10.1016/j.ijar.2012.10.003
79. Przybyla-Kasperek, M.: Three conflict methods in multiple classifiers that use dispersed knowledge. International Journal of Information Technology and Decision Making **18**(2), 555–599 (2019). doi:10.1142/S0219622019500020
80. Przybyla-Kasperek, M., Wakulicz-Deja, A.: A dispersed decision-making system - the use of negotiations during the dynamic generation of a system's structure. Information Systems **288**, 194–219 (2014). doi:10.1016/j.ins.2014.07.032
81. Przybyla-Kasperek, M., Wakulicz-Deja, A.: Global decision-making system with dynamically generated clusters. Information Sciences **270**, 172–191 (2014). doi:10.1016/j.ins.2014.02.076
82. Przybyla-Kasperek, M., Wakulicz-Deja, A.: The strength of coalition in a dispersed decision support system with negotiations. European Journal of Operational Research **252**(3), 947–968 (2016). doi:10.1016/j.ejor.2016.02.008
83. Radzikowska, A., Kerre, E.E.: A comparative study of fuzzy rough sets. Fuzzy Sets and Systems **126**(2), 137–155 (2002). doi:10.1016/S0165-0114(01)00032-X
84. Ramanna, S., Peters, J.F., Skowron, A.: Generalized conflict and resolution model with approximation spaces. In: S. Greco, Y. Hata, S. Hirano, M. Inuiguchi, S. Miyamoto, H.S. Nguyen, R. Slowinski (eds.) Rough Sets and Current Trends in Computing, 5th International Conference, RSCTC 2006, Kobe, Japan, November 6-8, 2006, Proceedings, *Lecture Notes in Computer Science*, vol. 4259, pp. 274–283. Springer (2006). doi:10.1007/11908029_30. URL `https://doi.org/10.1007/11908029_30`
85. Ramanna, S., Peters, J.F., Skowron, A.: Approaches to conflict dynamics based on rough sets. Fundamenta Informaticae **75**(1-4), 453–468 (2007)
86. Ramanna, S., Skowron, A.: Requiremnents interaction an conflicts A rough set approach. In: Proceedings of the IEEE Symposium on Foundations of Computational Intelligence, FOCI 2007, part of the IEEE Symposium Series on Computational Intelligence 2007, Honolulu, Hawaii, USA, 1-5 April 2007, pp. 308–313. IEEE (2007). doi:10.1109/FOCI.2007.372185
87. Ramanna, S., Skowron, A., Peters, J.F.: Approximation space-based socio-technical conflict model. In: J. Yao, P. Lingras, W. Wu, M.S. Szczuka, N. Cercone, D. Slezak (eds.) Rough Sets and Knowledge Technology, Second International Conference,RSKT 2007, Toronto, Canada, May 14-16, 2007, Proceedings, *Lecture Notes in Computer Science*, vol. 4481, pp. 476–483. Springer (2007). doi:10.1007/978-3-540-72458-2_59
88. Rehman, N., Ali, A., Hila, K.: Soft dominance based multigranulation decision theoretic rough sets and their applications in conflict problems. Artificial Intelligence Review **53**(8), 6079–6110 (2020). doi:10.1007/s10462-020-09843-4
89. Roy, A., Pal, S.K.: Fuzzy discretization of feature space for a rough set classifier. Pattern Recognition Letters **24**(6), 895–902 (2003). doi:10.1016/S0167-8655(02)00201-5
90. Saaty, T.L., Alexander, J. (eds.): Conflict Resolution: The Analytic Hierarchy Proces. Praeger, New York (1989)
91. Shafer, G.: Mathematical theory of evidence. Princeton University Press, Princeton, New Jersey (1976)
92. Skowron, A.: The evidence theory and decision tables. Bull. EATCS **39**, 199–204 (1989)
93. Skowron, A.: The rough sets theory and evidence theory. Fundamenta Informaticae **13**, 245–262 (1990). doi:10.3233/FI-1990-13303

94. Skowron, A., Grzymała-Busse, J.W.: From rough set theory to evidence theory. In: R. Yager, M. Fedrizzi, J. Kacprzyk (eds.) Advances in the Dempster-Shafer Theory of Evidence, pp. 193–236. John Wiley & Sons, New York, NY (1994)
95. Skowron, A., Pal, S.K.: Special volume: Rough sets, pattern recognition and data mining. Pattern Recognition Letters **24(6)** (2003). doi:10.1016/S0167-8655(02)00195-2
96. Skowron, A., Pal, S.K., Nguyen, H.S.: Preface: Special issue on Rough sets and Fuzzy sets in Natural computing. Theoretical Computer Science **412(42)**, 5816–5819 (2011). doi:10.1016/j.tcs.2011.05.036
97. Skowron, A., Ramanna, S., Peters, J.F.: Conflict analysis and information systems: A rough set approach. In: G. Wang, J.F. Peters, A. Skowron, Y. Yao (eds.) Rough Sets and Knowledge Technology, First International Conference, RSKT 2006, Chongqing, China, July 24-26, 2006, Proceedings, *Lecture Notes in Computer Science*, vol. 4062, pp. 233–240. Springer (2006). doi:10.1007/11795131_34
98. Swiniarski, R.W., Skowron, A.: Independent component analysis, principal component analysis and rough sets in face recognition. In: J.F. Peters, A. Skowron (eds.) Transactions on Rough Sets I: Journal Subline, *Lecture Notes in Computer Science*, vol. 3100, pp. 392–404. Springer, Heidelberg (2004). doi:10.1007/978-3-540-27794-1_19
99. Sycara, K.: Multiagent systems. AI Magazine **19**, 79–92 (Summer 1998)
100. Tang, X., Zeng, T., Tan, Y., Ding, B.: Conflict analysis based on three-way decision theoretic fuzzy rough set over two universes. Ingénierie des Systèmes d'Information **25**(1), 75–82 (2020). doi:10.18280/isi.250110. URL `https://doi.org/10.18280/isi.250110`
101. Theerens, A., Cornelis, C.: Fuzzy rough sets based on fuzzy quantification. Fuzzy Sets and Systems **473**, 108,704 (2023). doi:10.1016/J.FSS.2023.108704
102. Theerens, A., Lenz, O.U., Cornelis, C.: Choquet-based fuzzy rough sets. International Journal of Approximate Reasoning **146**, 62–78 (2022). doi:10.1016/J.IJAR.2022.04.006
103. Tong, Z., Xu, P., Denœux, T.: An evidential classifier based on Dempster-Shafer theory and deep learning. Neurocomputing **450**, 275–293 (2021). doi:10.1016/j.neucom.2021.03.066
104. Torra, V., Dahlbom, A., Narukawa, Y. (eds.): Fuzzy Sets, Rough Sets, Multisets and Clustering, *Studies in Computational Intelligence*, vol. 671. Springer Nature, Cham (2017). doi:10.1007/978-3-319-47557-8
105. Vluymans, S., D'eer, L., Saeys, Y., Cornelis, C.: Applications of fuzzy rough set theory in machine learning: a survey. Fundamenta Informaticae **142**(1-4), 53–86 (2015). doi:10.3233/FI-2015-1284
106. Wakulicz-Deja, A., Przybyla-Kasperek, M.: Multi-agent decision taking system. Fundamenta Informaticae **101**(1-2), 125–141 (2010). doi:10.3233/FI-2010-280
107. Wakulicz-Deja, A., Przybyla-Kasperek, M.: Pawlak's conflict model: Directions of development. In: M. Ganzha, L.A. Maciaszek, M. Paprzycki (eds.) Proceedings of the 2016 Federated Conference on Computer Science and Information Systems, FedCSIS 2016, Gdańsk, Poland, September 11-14, 2016, *Annals of Computer Science and Information Systems*, vol. 8, pp. 191–197. IEEE (2016). doi:10.15439/2016F003
108. Weyers, B., Bowen, J., Dix, A., Palanque, P. (eds.): The Handbook of Formal Methods in Human-Computer Interaction. Springer, Cham (2017). doi:10.1007/978-3-319-51838-1
109. Wojna, A., Latkowski, R.: Rseslib 3: Library of rough set and machine learning methods with extensible architecture. Transactions on Rough Sets XXI: Journal Subline **21**, 301–323 (2019). doi:10.1007/978-3-662-58768-3_7
110. Wu, H.C.: Mathematical Foundation of Fuzzy Sets. John Wiley & Sons, Hoboken, NJ (2023). doi:10.1002/9781119981558
111. Wu, W., Leung, Y., Mi, J.: On characterizations of (I, J)-fuzzy rough approximation operators. Fuzzy Sets Systems **154**(1), 76–102 (2005). doi:10.1016/j.fss.2005.02.011
112. Yadav, S.P., Bhati, B.S., Mahato, D.P., Kumar, S. (eds.): Federated Learning for IoT Applications. EAI/Springer Innovations in Communication and Computing. Springer, Cham (2022). doi:10.1007/978-3-030-85559-8
113. Yang, Q., Liu, Y., Cheng, Y., Kang, Y., Chen, T., Yu, H. (eds.): Federated Learning, *Synthesis Lectures on Artificial Intelligence and Machine Learning*, vol. 43. Morgan & Claypool (2019). doi:10.2200/S00960ED2V01Y201910AIM043

114. Yao, Y.: A comparative study of fuzzy sets and rough sets. Information Sciences **109**, 227–242 (1998). doi:10.1016/S0020-0255(98)10023-3
115. Zadeh, L.A.: Fuzzy sets. Information and Control **8**, 338–353 (1965). doi:10.1016/S0019-9958(65)90241-X
116. Zadeh, L.A.: Fuzzy logic, neural networks, and soft computing. Communications of the ACM **37**, 77–84 (1994). doi:10.1145/175247.175255
117. Zadeh, L.A.: From computing with numbers to computing with words – from manipulation of measurements to manipulation of perceptions. IEEE Transactions on Circuits and Systems **45**, 105–119 (1999). doi:10.1109/81.739259
118. Zadeh, L.A.: A new direction in AI: Toward a computational theory of perceptions. AI Magazine **22**(1), 73–84 (2001). doi:10.1609/aimag.v22i1.1545

Chapter 11
Rough Sets in the Perspective of Logics and Computations

This chapter is prepared on the basis of several articles, in particular [**?**, **?**]. Wr concentrate of the relationships of rough sets, approximate Boolean reasoning and scalability issues.

Mining large data sets is one of the biggest challenges in KDD. In many practical applications, there is a need of data mining algorithms running on terminals of a client–server database system where the only access to database (located in the server) is enabled by SQL queries.

Unfortunately, the proposed so far data mining methods based on rough sets and Boolean reasoning approach are characterized by high computational complexity and their straightforward implementations are not applicable for large data sets. The critical factor for time complexity of algorithms solving the discussed problem is the number of simple SQL queries like

```
SELECT COUNT FROM a_Table WHERE a_Condition
```

In this section, we present some efficient modifications of these methods to solve out this problem. We consider the following issues:

- Searching for short reducts from large data sets;
- Searching for best partitions defined by cuts on continuous attributes;

11.1 Reduct calculation

Let us again illustrate the idea of reduct calculation using discernibility matrix (Table 11.2).

Example 11.1. Let us consider the "weather" problem, which is represented by decision table (see Table 11.1). Objects are described by four conditional attributes and are divided into 2 classes. Let us consider the first 12 observations. In this example, $U = \{1,2,\ldots,12\}$, $A = \{a_1,a_2,a_3,a_4\}$, $CLASS_{no} = \{1,2,6,8\}$, $CLASS_{yes} = \{3,4,5,7,9,10,11,12\}$.

Table 11.1 The exemplary "weather" decision table

date	outlook	temperature	humidity	windy	play
ID	a_1	a_2	a_3	a_4	*dec*
1	sunny	hot	high	FALSE	no
2	sunny	hot	high	TRUE	no
3	overcast	hot	high	FALSE	yes
4	rainy	mild	high	FALSE	yes
5	rainy	cool	normal	FALSE	yes
6	rainy	cool	normal	TRUE	no
7	overcast	cool	normal	TRUE	yes
8	sunny	mild	high	FALSE	no
9	sunny	cool	normal	FALSE	yes
10	rainy	mild	normal	FALSE	yes
11	sunny	mild	normal	TRUE	yes
12	overcast	mild	high	TRUE	yes

Table 11.2 The compact form of discernibility matrix (right)

$\mathscr{M}$	1	2	6	8
3	a_1	a_1, a_4	a_1, a_2, a_3, a_4	a_1, a_2
4	a_1, a_2	a_1, a_2, a_4	a_2, a_3, a_4	a_1
5	a_1, a_2, a_3	a_1, a_2, a_3, a_4	a_4	a_1, a_2, a_3
7	a_1, a_2, a_3, a_4	a_1, a_2, a_3	a_1	a_1, a_2, a_3, a_4
9	a_2, a_3	a_2, a_3, a_4	a_1, a_4	a_2, a_3
10	a_1, a_2, a_3	a_1, a_2, a_3, a_4	a_2, a_4	a_1, a_3
11	a_2, a_3, a_4	a_2, a_3	a_1, a_2	a_3, a_4
12	a_1, a_2, a_4	a_1, a_2	a_1, a_2, a_3	a_1, a_4

The discernibility matrix can be treated as a board containing $n \times n$ boxes. Noteworthy is the fact that discernibility matrix is symmetrical with respect to the main diagonal, because $M_{i,j} = M_{j,i}$, and that sorting all objects according to their decision classes causes a shift off all empty boxes nearby to the main diagonal. In case of decision table with two decision classes, the discernibility matrix can be rewritten in a more compact form as shown in Table 11.2. The discernibility function is constructed from discernibility matrix by taking a conjunction of all discernibility clauses. After reducing of all repeated clauses we have:

$$\begin{aligned}
f(a_1,a_2,a_3,a_4) \equiv &(a_1) \wedge (a_1 \vee a_4) \wedge (a_1 \vee a_2) \wedge (a_1 \vee a_2 \vee a_3 \vee a_4) \wedge \\
&(a_1 \vee a_2 \vee a_4) \wedge (a_2 \vee a_3 \vee a_4) \wedge (a_1 \vee a_2 \vee a_3) \wedge \\
&(a_4) \wedge (a_2 \vee a_3) \wedge (a_2 \vee a_4) \wedge (a_1 \vee a_3) \wedge (a_3 \vee a_4) \wedge \\
&(a_1 \vee a_2 \vee a_4).
\end{aligned}$$

One can find relative reducts of the decision table by searching for its prime implicants. The straightforward method calculates all prime implicants by translation to

DNF (using absorbtion rule $p(p+q) \equiv p$ another rules for boolean algebra). One can do it as follow:

$$f \equiv (a_1) \wedge (a_4) \wedge (a_2 \vee a_3) = a_1 \wedge a_4 \wedge a_2 \vee a_1 \wedge a_4 \wedge a_3.$$

Thus we have 2 reducts: $R_1 = \{a_1, a_2, a_4\}$ and $R_2 = \{a_1, a_3, a_4\}$.

Every heuristic algorithm for the prime implicant problem can be applied to the discernibility function to solve the minimal reduct problem. One of such heuristics was proposed in [**?**] and was based on the idea of greedy algorithm, where each attribute is evaluated by its discernibility measure, *i.e.*, the number of pairs of objects which are discerned by the attribute, or, equivalently, the number of its occurrences in the discernibility matrix.

- First we have to calculate the number of occurrences of each attributes in the discernibility matrix:

$$eval(a_1) = disc_{dec}(a_1) = 23 \qquad eval(a_2) = disc_{dec}(a_2) = 23$$
$$eval(a_3) = disc_{dec}(a_3) = 18 \qquad eval(a_4) = disc_{dec}(a_4) = 16$$

  Thus a_1 and a_2 are the two most preferred attributes.
- Assume that we select a_1. Now we are taking under consideration only those cells of the discernibility matrix which are not containing a_1. There are 9 such cells only, and the number of occurrences are as the following:

$$eval(a_2) = disc_{dec}(a_1, a_2) - disc_{dec}(a_1) = 7$$
$$eval(a_3) = disc_{dec}(a_1, a_3) - disc_{dec}(a_1) = 7$$
$$eval(a_4) = disc_{dec}(a_1, a_4) - disc_{dec}(a_1) = 6$$

- If this time we select a_2, then the are only 2 remaining cells, and, both are containing a_4;
- Therefore, the greedy algorithm returns the set $\{a_1, a_2, a_4\}$ as a reduct of sufficiently small size.

There is another reason for choosing a_1 and a_4, because they are *core attributes*[1]. It has been shown that an attribute is a core attribute if and only if occurs in the discernibility matrix as a singleton [**?**]. Therefore, core attributes can be recognized by searching for all single cells of the discernibility matrix. The pseudo-code of this algorithm is presented in Algorithm 2.

The reader may have a feeling that the greedy algorithm for reduct problem has quite a high complexity, because two main operations:

- $disc(B)$ – number of pairs of objects discerned by attributes from B;
- $isCore(a)$ – check whether a is a core attribute;

[1] An attribute is called core attribute if and only if it occurs in every reduct.

Algorithm 2: Searching for short reduct

```
begin
  B := ∅;
    // Step 1. Initializing B by core attributes
    for a ∈ A do
        if isCore(a) then
          B := B ∪ {a};
        end
    end
    // Step 2. Including attributes to B
    repeat
        a_max := arg max_{a∈A−B} disc_dec(B ∪ {a});
         eval(a_max) := disc_dec(B ∪ {a_max}) − disc_dec(B);
         if eval(a_max) > 0 then
              B := B ∪ {a};
         end
    until (eval(a_max) == 0) OR (B == A);
    // Step 3. Elimination
    for a ∈ B do
        if disc_dec(B) = disc_dec(B − {a}) then
          B := B − {a};
        end
end end
```

are defined by the discernibility matrix which is a complex data structure containing $O(n^2)$ cells, and each cell can contain up to $O(m)$ attributes, where n is the number of objects and m is the number of attributes of the given decision table. This suggests that the two main operations need at least $O(mn^2)$ computational time.

Fortunately, both operations can be performed more efficiently. It has been shown [**?**] that both operations can be calculated in time $O(mn\log n)$ without the necessity to store the discernibility matrix.

Now we will show that this algorithm can be efficiently implemented in DBMS using only simple SQL queries.

Let $\mathbb{A} = (U, A \cup \{dec\})$ be a decision table. By "*counting table*" of a set of objects $X \subset U$ we denoted the vector:

$$CountTable(X) = \langle n_1, \ldots, n_d \rangle,$$

where $n_k = card(X \cap CLASS_k)$ is the number of objects from X belonging to the k^{th} decision class. We define a conflict measure of X by

$$conflict(X) = \sum_{i<j} n_i n_j = \frac{1}{2}\left[\left(\sum_{k=1}^{d} n_k\right)^2 - \sum_{k=1}^{d} n_k^2\right].$$

In other words, $conflict(X)$ is the number of pairs of different class objects.

By *counting table* of a set of attributes B we mean the two-dimensional array $Count(B) = [n_{v,k}]_{v \in INF(B), k \in V_{dec}}$, where

$$n_{v,k} = card(\{x \in U : inf_B(x) = v \text{ and } dec(x) = k\}).$$

Thus $Count(B)$ is a collection of counting tables of equivalence classes of the indiscernibility relation IND_B. It is clear that the complexity time for the construction of counting table is $O(nd \log n)$, where n is the number of objects and d is the number of decision classes. It is clear that counting table can be easily constructed in data base management systems using simple SQL queries.

The discernibility measure of a set of attributes B can be easily calculated from the counting table as follows:

$$disc_{dec}(B) = \frac{1}{2} \sum_{v \neq v', k \neq k'} n_{v,k} \cdot n_{v',k'}.$$

The disadvantage of this equation relates to the fact that it requires $O(S^2)$ operations, where S is the size of the counting table $Count(B)$.

The discernibility measure can be understood as a number of unresolved (by the set of attributes B) conflicts. One can show that:

$$disc_{dec}(B) = conflict(U) - \sum_{[x] \in U/IND_B} conflict([x]_{IND_B}). \tag{11.1}$$

Thus, the discernibility measure can be determined in $O(S)$ time:

$$disc_{dec}(B) = \frac{1}{2}\left(n^2 - \sum_{k=1}^{d} n_k^2\right) - \frac{1}{2} \sum_{v \in INF(B)} \left[\left(\sum_{k=1}^{d} n_{v,k}\right)^2 - \sum_{k=1}^{d} n_{v,k}^2\right], \tag{11.2}$$

where $n_k = |CLASS_k| = \sum_v n_{v,k}$ is the size of k^{th} decision class.

Moreover, one can show that attribute a is a core attribute of decision table $=(U, A \cup \{dec\})$ if and only if

$$disc_{dec}(A - \{a\}) < disc_{dec}(A). \tag{11.3}$$

Thus both operations $disc_{dec}(B)$ and $isCore(a)$ can be performed in linear time with respect to the counting table.

Example 11.2. The counting table for a_1 is as follows:

$Count(a_1)$	$dec = no$	$dec = yes$
$a_1 = sunny$	3	2
$a_1 = overcast$	0	3
$a_1 = rainy$	1	3

We illustrate Eqn. (11.2) by inserting some additional columns to the counting table:

$Count(a_1)$	$dec = no$	$dec = yes$	$\sum$	$conflict(.)$
$a_1 = sunny$	3	2	5	$\frac{1}{2}(5^2 - 2^2 - 3^2) = 6$
$a_1 = overcast$	0	3	3	$\frac{1}{2}(3^2 - 0^2 - 3^2) = 0$
$a_1 = rainy$	1	3	4	$\frac{1}{2}(4^2 - 1^2 - 3^2) = 3$
U	4	8	12	$\frac{1}{2}(12^2 - 8^2 - 4^2) = 32$

Thus $disc_{dec}(a_1) = 32 - 6 - 0 - 3 = 23$.

Another scalable method for generation of reducts is presented in [**?**]. The reader is also referred to the book [**?**] for the rough set based scalable methods with applications in bioinformatics, to [**?**] for application of *dynamic reducts*, to [**?**] for application of *random reducts* to the feature selection problem, to [**?**] parallel attribute reduction with application of MapReduce or to [**?**] for scalable rough set based methods using FPGA.

11.2 Mining of Large Data Sets Stored in Relational Databases

Mining large data sets is one of the biggest challenges in Knowledge Discovery and Data Mining. In many practical applications, there is a need of data mining algorithms running on terminals of possibly distributed database systems where the only access to data is enabled by SQL queries or NoSQL operations.

Let us consider two illustrative examples of problems for large data sets: (i) searching for short reducts, (ii) searching for best partitions defined by cuts on continuous attributes. In both cases the traditional implementations of rough sets and Boolean reasoning based methods are characterized by the high computational cost. The critical factor for time complexity of algorithms solving the discussed problems is the number of data access operations. Fortunately some efficient modifications of the original algorithms were proposed by relying on concurrent retrieval of higher level statistics which are sufficient for the heuristic search of reducts and partitions (see, *e.g.*, [**?**, **?**, **?**, **?**]). The rough set approach was also applied in development of other scalable big data processing techniques (*e.g.*, Infobright `http://www.infobright.com/`).

In this section (see, *e.g.*, [**?**, **?**, **?**]), we discuss an application of approximate Boolean reasoning to efficient searching for cuts in large data sets stored in relational databases. Searching for relevant cuts is based on simple statistics which can be efficiently extracted from relational databases. This additional statistical knowledge is making it possible to perform the searching based on Boolean reasoning much more efficient. It can be shown that the extracted cuts by using such reasoning are quite close to optimal.

Searching algorithms for optimal partitions of real-valued attributes, defined by cuts, have been intensively studied. The main goal of such algorithms is to discover cuts which can be used to synthesize decision trees or decision rules of high quality wrt some quality measures (*e.g.*, quality of classification of new unseen objects, quality defined by the decision tree height, support and confidence of decision rules).

In general, all those problems are hard from computational point of view (*e.g.*, the searching problem for minimal and consistent set of cuts is NP-hard). In consequence, numerous heuristics have been developed for approximate solutions of these problems. These heuristics are based on approximate measures estimating the quality of extracted cuts. Among such measures *discernibility measures* are relevant for the rough set approach.

We outline an approach for solution of a searching problem for optimal partition of real-valued attributes by cuts, assuming that the large data table is represented in a relational database. In such a case, even the linear time complexity with respect to the number of cuts is not acceptable because of the time needed for one step. The critical factor for time complexity of algorithms solving that problem is the number of SQL queries of the form

```
SELECT COUNT
  FROM a Table
     WHERE (an attribute BETWEEN value1 AND value2)
           AND (additional condition)
```

necessary to construct partitions of real-valued attribute sets. We assume the answer time for such queries does not depend on the interval length[2]. Using a straightforward approach to optimal partition selection (wrt a given measure), the number of necessary queries is of order $O(N)$, where N is the number of preassumed cuts. By introducing some optimization measures, it is possible to reduce the size of searching space. Moreover, using only $O(\log N)$ simple queries, suffices to construct a partition very close to optimal.

Let $\mathbb{A} = (U, A, d)$ be a decision system with real-valued condition attributes. Any cut (a, c), where $a \in A$ and c is a real number, defines two disjoint sets given by

$$U_L(a,c) = \{x \in U : a(x) \leq c\};$$
$$U_R(a,c) = \{x \in U : a(x) > c\}.$$

If both $U_L(a,c)$ and $U_R(a,c)$ are non-empty, then c is called a *cut on attribute* a. The cut (a,c) discerns a pair of objects x, y if either $a(x) < c \leq a(y)$ or $a(y) < c \leq a(x)$.

Let $\mathbb{A} = (U, A, d)$ be a decision system with real-valued condition attributes and decision classes X_i, for $i = 1, \ldots, r(d)$. A *quality of a cut* (a,c), denoted by $W(a,c)$, is defined by

$$\begin{aligned} W(a,c) &= \sum_{i \neq j}^{r(d)} L_i(a,c) * R_j(a,c) \qquad (11.4) \\ &= \left(\sum_{i=1}^{r(d)} L_i(a,c)\right) * \left(\sum_{i=1}^{r(d)} R_i(a,c)\right) - \sum_{i=1}^{r(d)} L_i(a,c) * R_i(a,c), \end{aligned}$$

[2] This assumption is satisfied in some existing database management systems.

where $L_i(a,c) = card(X_i \cap U_L(a,c))$ and $R_i(a,c) = card(X_i \cap U_R(a,c))$, for $i = 1,\ldots,r(d)$.

In the sequel, we will be interested in finding cuts maximizing the function $W(a,c)$.

The following definition will be useful. Let $\mathscr{C}_a = \{(a,c_1),\ldots,(a,c_N)\}$ be a set of cuts on attribute a, over a decision table $\mathbb{A}$ and assume $c_1 < c_2 ... < c_N$. By a *median of the i^{th} decision class*, denoted by $Median(i)$, we mean the minimal index j for which the cut $(a,c_j) \in \mathscr{C}_a$ minimizes the value $|L_i(a,c_j) - R_i(a,c_j)|$,[3] where L_i and R_i are defined before.

One can use only $O(r(d) * \log N)$ SQL queries to determine the medians of decision classes by using the well-known binary search algorithm.

Then one can show that the quality function $W_a(i) \stackrel{\text{def}}{=} W(a,c_i)$, for $i = 1,\ldots,N$, is increasing in $\{1,\ldots,min\}$ and decreasing in $\{max,\ldots,N\}$, where *min* and *max* are defined by

$$min = \min_{1\le i\le N} Median(i);$$
$$max = \max_{1\le i\le N} Median(i).$$

In consequence, the search space for maximum of $W(a,c_i)$ is reduced to $i \in [min, max]$.

Now, one can apply the divide and conquer strategy to determine the best cut, given by $c_{Best} \in [c_{min}, c_{max}]$, wrt the chosen quality function. First, we divide the interval containing all possible cuts into k intervals. Using some heuristics, one then predict the interval which most probably contains the best cut. This process is recursively applied to that interval, until the considered interval consists of one cut. The problem which remains to be solved is how to define such approximate measures which could help us to predict the suitable interval.

Let us consider a simple probabilistic model. Let (a,c_L), (a,c_R) be two cuts such that $c_L < c_R$ and $i = 1,\ldots,r(d)$. For any cut (a,c) satisfying $c_L < c < c_R$, we assume that $x_1,\ldots,x_{r(d)}$, where $x_i = card(X_i \cap U_L(a,c) \cap U_R(a,c))$ are independent random variables with uniform distribution over sets $\{0,...,M_1\}$, ..., $\{0,...,M_{r(d)}\}$, respectively, that

$$M_i = M_i(a,c_L,c_R) = card(X_i \cap U_L(a,c_R) \cap U_R(a,c_L)).$$

Under these assumptions the following fact holds. For any cut $c \in [c_L, c_R]$, the mean $E(W(a,c))$ of quality $W(a,c)$, is given by

$$E(W(a,c)) = \frac{W(a,c_L) + W(a,c_R) + conflict((a,c_L),(a,c_R))}{2}, \tag{11.5}$$

where $conflict((a,c_L),(a,c_R)) = \sum_{i\neq j} M_i * M_j$.

[3] The minimization means that $|L_i(a,c_j) - R_i(a,c_j)| = \min_{1\le k\le N} |L_i(a,c_k) - R_i(a,c_k)|$.

In addition, the standard deviation of $W(a,c)$ is given by

$$D^2(W(a,c)) = \sum_{i=1}^{n} \left[\frac{M_i(M_i+2)}{12} \left(\sum_{j \neq i} (R_j(a,c_R) - L_j(a,c_L)) \right)^2 \right]. \tag{11.6}$$

Formulas (11.5) and (11.6) can be used to construct a predicting measure for the quality of the interval $[c_L, c_R]$:

$$Eval\,([c_L,c_R],\alpha) = E(W(a,c)) + \alpha\sqrt{D^2(W(a,c))}, \tag{11.7}$$

where the real parameter $\alpha \in [0,1]$ can be tuned in a learning process.

To determine the value $Eval\,([c_L,c_R],\alpha)$, we need to compute the numbers

$$L_1(a,c_L),\ldots,L_{r(d)}(a,c_L),M_1,\ldots,M_{r(d)},R_1(a,c_R),\ldots,R_{r(d)}(a,c_R).$$

This requires $O(r(d))$ SQL queries of the form

```
SELECT COUNT
FROM DecTable
WHERE (attribute a BETWEEN value1 AND value2)
       AND (dec = i).
```

Hence, the number of queries required for running this algorithm is

$$O(r(d)k\log_k N).$$

In practice, we set $k = 3$, since the function $f(k) = r(d)k\log_k N$ over positive integers is taking minimum for $k = 3$.

Numerous experiments on different data sets have shown that the proposed solution allows one to find a cut which is very close to the optimal one. For more details the reader is referred to the literature (see [**?**, **?**]).

Problem 11.1. Prove that the equality in Eqn. 11.1 holds.

Problem 11.2. Prove that the equality in Eqn. 11.2 holds.

Problem 11.3. Prove that the inequality 11.3 holds.

Problem 11.4. Develop efficient heuristics for computing the Core of the set of reducts of information systems.

Problem 11.5. Decision reduct relative to the generalized decision function [**?**].
For a given decision system $\mathbb{A} = (U,A,\{d\}$, a subset $R \subseteq A$ is called $POS_A(d)$-reduct if an only if $POS_A(d) = POS_R(d)$ and R is a minimal subset of A satisfying this condition.
Develop efficient heuristics for computing $POS_A(d)$-reducts.
Is this definition equivalent to the definition of d-reduct?

Problem 11.6. Develop a decision procedure checking if a given attribute does not belong to any reduct of a given information system.

Problem 11.7. r-reducts [**?**]
For a given decision system $\mathbb{A} = (U, A, \{d\}$, a subset $S \subseteq A$ is called an r-superreduct (where $1 \leq r < |S|$) if and only if for any $R \subseteq S$ with at most r elements the set $S \setminus R$ is a superreduct (*i.e.,* consists of d-reduct) of $\mathbb{A}$. We say that R is an r-reduct of $\mathbb{A}$, if and only if it is an r-superreduct of $\mathbb{A}$ and there is no proper subset $R' \subset R$, which is an r-superreduct of $\mathbb{A}$.
Develop efficient heuristics for computing r-reducts.

References

1. Bazan, J.G.: A comparison of dynamic and non-dynamic rough set methods for extracting laws from decision tables. In: L. Polkowski, A. Skowron (eds.) Rough Sets in Knowledge Discovery 1: Methodology and Applications, *Studies in Fuzziness and Soft Computing*, vol. 18, pp. 321–365. Physica-Verlag, Heidelberg (1998)
2. Grzegorowski, M.: Governance of the redundancy in the feature selection based on rough sets' reducts. In: V. Flores, F. Gomide, A. Janusz, C. Meneses, D. Miao, G. Peters, D. Ślęzak, G. Wang, R. Weber, Y. Yao (eds.) Rough Sets - International Joint Conference, IJCRS 2016, Santiago de Chile, Chile, October 7-11, 2016, Proceedings, *Lectures Notes in Computer Science*, vol. 9920, pp. 548–557. Springer, Heidelberg (2016). doi:10.1007/978-3-319-47160-0_50
3. Kopczynski, M., Grzes, T., Stepaniuk, J.: FPGA in rough-granular computing: Reduct generation. In: 2014 IEEE/WIC/ACM International Joint Conferences on Web Intelligence (WI) and Intelligent Agent Technologies (IAT), Warsaw, Poland, August 11-14, 2014 - Volume I, pp. 364–370. IEEE Computer Society (2014). doi:10.1109/WI-IAT.2014.120
4. Kruczyk, M., Baltzer, N., Mieczkowski, J., Draminski, M., Koronacki, J., Komorowski, J.: Random reducts: A Monte Carlo rough set-based method for feature selection in large datasets. Fundamenta Informaticae **127**(1-4), 273–288 (2013). doi:10.3233/FI-2013-909
5. Kryszkiewicz, M., Cichoń, K.: Towards scalable algorithms for discovering rough set reducts. In: Transactions on Rough Sets II. Rough sets and fuzzy sets: Journal Subline, pp. 120–143. doi:10.1007/978-3-540-27794-1_5
6. Maji, P., Paul, S.: Scalable Pattern Recognition Algorithms. Applications in Computational Biology and Bioinformatics. Springer, Berlin (2014). doi:10.1007/978-3-319-05630-2
7. Nguyen, H.S.: Efficient SQL-learning method for data mining in large data bases. In: T. Dean (ed.) Sixteenth International Joint Conference on Artificial Intelligence IJCAI, pp. 806–811. Morgan-Kaufmann Publishers, Stockholm, Sweden (1999)
8. Nguyen, H.S.: On efficient handling of continuous attributes in large data bases. Fundamenta Informaticae **48**(1), 61–81 (2001)
9. Nguyen, H.S.: Approximate boolean reasoning: Foundations and applications in data mining. In: J.F. Peters, A. Skowron (eds.) Transactions on Rough Sets V: Journal Subline, *Lecture Notes in Computer Science*, vol. 4100, pp. 344–523. Springer, Heidelberg (2006). doi:10.1007/11847465_16
10. Nguyen, S.H., Nguyen, H.S.: Some efficient algorithms for rough set methods. In: Sixth International Conference on Information Processing and Management of Uncertainty on Knowledge Based Systems IPMU'1996, vol. III, pp. 1451–1456. Granada, Spain (1996)
11. Skowron, A., Nguyen, H.S.: Rough sets: From rudiments to challenges. In: A. Skowron, Z. Suraj (eds.) Rough Sets and Intelligent Systems. Professor Zdzisław Pawlak in Memoriam, *Series Intelligent Systems Reference Library*, vol. 42, pp. 75–173. Springer, Heidelberg (2013). doi:10.1007/978-3-642-30344-9_3

12. Skowron, A., Rauszer, C.: The discernibility matrices and functions in information systems. In: Intelligent Decision Support - Handbook of Applications and Advances of the Rough Sets Theory, pp. 331–362
13. Skowron, A., Suraj, Z. (eds.): Rough Sets and Intelligent Systems. Professor Zdzisław Pawlak in Memoriam. Series Intelligent Systems Reference Library. Springer (2013). doi:10.1007/978-3-642-30344-9; 10.1007/978-3-642-30341-8
14. Xi, D., Wang, G., Zhang, X., Zhang, F.: Parallel attribute reduction based on MapReduce. In: Proceedings of the 9th International Conference on Rough Sets and Knowledge Technology RSKT 2014, Shanghai, China, October 24-26, 2014, pp. 631–641 (2014). doi:10.1007/978-3-319-11740-9_58

Chapter 12
Granular Computing and Rough Set Theory

This chapter presents a discussion of the Granular Computing (GrC) model and its extension, the Interactive Granular Computing (IGrC) model. Additionally, we present the rough set approach within these frameworks and provide a roadmap for the further development of rough sets based on IGrC.

This chapter builds upon previous works related to IGrC (see, *e.g.*, [**?**, **?**, **?**, **?**] as well as [**?**, **?**] and papers referred at `https://dblp.org/pid/s/AndrzejSkowron.html`).

This chapter presents a discussion of the potential for the IGrC model to serve as a foundation for the design of Intelligent Systems (IS's) capable of addressing complex phenomena. IS's are regarded as a specific category of basic objects in IGrC, designated as complex granules (c-granules) with control. The linking of abstract and physical objects is a crucial aspect of modeling the perception of situations in the physical world by IS's. The control of c-granules is responsible for steering the computations generated over granular networks in a manner that achieves the desired target goals (*i.e.*, satisfies the specified requirements). It is essential that the control of c-granules has a comprehensive understanding of the perceived situations in the physical world. This enables the relevant transformations of configurations of physical objects to be realized, which in turn facilitates the generation of high-quality computations over granular networks. For example, the optimization of traffic lights can facilitate the smooth passage of cars through urban areas (see [**?**] and Section 5.3 in Chapter 5).

We present a generalization of the rough set approach in this context.

In essence, the control of c-granules is based on an efficient and online judgment mechanism of c-granules, thereby enabling c-granules to respond to the following queries:

- How to construct a list of the most pressing issues that require resolution in the context of the current situation?
- How might one identify high-quality approximate solutions to the most pressing problems, given the constraints of real-world scenarios?
- What is the optimal subsequent action, that is, which transformations should be activated in the current situation (*e.g.*, facilitating a more comprehensive

understanding of the situation and/or taking a step towards achieving the desired outcome)?

One of the primary challenges in developing IS's that address complex phenomena is the creation of satisfactory computational models that implement adaptive judgment. This judgment is based on the granular computations generated by control mechanisms. The following sections will provide a more detailed discussion of these computations. In particular, we will examine how the approximations of concepts in IGrC are anchored in c-granule with control, rather than relying on the a priori defined approximation spaces delineated by relational systems, as is typical in the current rough set approach.

The objective of adaptive judgment is to develop the capacity for sound judgment and the ability to select the optimal course of action based on knowledge, experience, and understanding (see [**?**]).

Another challenge in the development of IS's is in the discovery of methods for approximate reasoning from measurements to perception, *i.e.*, from concepts derived from sensor concepts derived from sensor measurements to expressions expressed in natural language that express perception understanding. Today, new emerging computing paradigms are being explored to to make progress in solving problems related to this challenge in Perception Based Computing (PBC).

There is a huge literature dedicated to perception (see, *e.g.*, [**?**, **?**]). Here we will only mention that the essence of perception offered by the IGrC is closely related to the approach presented in the book by Noë [**?**] (p. 1):

> [...] *perceiving is a way of acting. Perception is not something that happens to us, or in us. It is something we do. Think of blind person tap-tapping his or her way around a cluttered space, perceiving the space by touch, not all at once, but through time, by skillful probing and movement. This is, or at least ought to be, our paradigm of what perceiving is. The world makes itself available to the perceiver through physical movement and interaction.*

PBC provides the ability to compute and reason with perceptual information just as humans do, to perform a wide variety of physical and mental tasks without measurement and calculation. Given the finite organs and (ultimately the brain) to resolve details, perceptions are inherently granular. The boundaries of perceived granules (*e.g.*, classes) are fuzzy, and the values of the of the attributes they can take are granular. In general, perception can be seen as understanding of sensory information. This point of view is discussed, for example, in Computing with Words and Perception (CWP) [**?**]:

> *derives from the fact that it opens the door to computation and reasoning with information which is perception – rather than measurement-based. Perceptions play a key role in human cognition, and underlie the remarkable human capability to perform a wide variety of physical and mental tasks without any measurements and any computations. Everyday examples of such tasks are driving a car in city traffic, playing tennis and summarizing a story.*

IGrC enhanced by dialogues with experts becomes closely related to CWP.

It should be noted that for the design of IS's addressing complex phenomena, applications of rough sets or fuzzy sets in the framework of PBC can be very useful. However, this requires changing the approach from one based on objects closed in only abstract space to one based on interacting objects that combine abstract and physical objects. This is the goal of IGrC. This topic is discussed in more detail here.

We emphasize the need for special reasoning techniques, called adaptive reasoning, performed over interactive granular computations aimed at generating high-quality approximate solutions to problems to be solved by IS's. This reasoning is also important for the discovery of complex patterns as computational building blocks for cognition (see [**?**]) in these computations.

One should note that there are several 'white spots' here. This means that we do not have satisfactory formal techniques supporting reasoning necessary for understanding the perceived situation to a degree making it possible to make the right decisions by IS's. Among them are reasoning techniques based on commonsense reasoning, experience based reasoning or analogy based reasoning. It is worthwhile to cite here the following opinion [**?**, **?**] about analogy based reasoning belonging to the experience based reasoning domain:

> *The quest for machines that can make abstractions and analogies is as old as the AI field itself, but the problem remains almost completely open.*

This causes that dialogues of IS's with human experts are unavoidable.

We also outline an approach to dynamic information systems considered in the context of the control of c-granules. This approach differs significantly from dynamic information systems reported in the literature (see, *e.g.*, [**?**, **?**, **?**]), where the dynamics are given a priori.

We hope to convince the reader that there are many challenges requiring development of new reasoning techniques leading from data perceived in interaction with the physical world to perception making it possible to understand the perceived situations, e.g., by constructing high quality approximations of complex vague concepts responsible for lunching the right decisions by IS's. Such complex vague concepts are components of complex games, which are crucial for the control of c-granules in the control of granular computations.

It should also be noted that this chapter leads to opening new research perspectives for rough sets and fuzzy sets. Approximation methods are based on the above-mentioned reasoning approaches, not limited to the existing ones, often based, *e.g.*, on partial inclusion of sets only. In fact, the proposed approach to rough sets and fuzzy sets based on IGrC creates a step towards the development of the foundations of Artificial Intelligence (AI).

Our approach is influenced by the book [**?**], which stimulated our work on the foundations for the design and analysis of IS's not burdened with unnecessary formalisms of partial approaches.

12.1 Introduction to Granular Computing

Granular Computing (GrC) is an emerging paradigm that focuses on processing and managing information at different levels of granularity. The basic concepts of GrC, such as information granularity and information granules, have emerged in various contexts and numerous areas of AI, such as data analytics or computer vision.

GrC is rapidly evolving, with ongoing research exploring its theoretical foundations and practical applications. Its ability to manage complexity and uncertainty positions it as a valuable tool in today's data-driven landscape. GrC effectively manages uncertainty through the use of multi-level representations, approximate reasoning, and the integration of theories such as rough sets and fuzzy logic. These mechanisms enhance the ability to make informed decisions in complex, uncertain environments, making GrC a powerful tool in various applications where uncertainty is prevalent. GrC is directly related to the pioneering work of Zadeh [**?**]. He coined an informal but highly descriptive and compelling concept of an information granule as a cluster of objects (or dots) that are bound together by indistinguishability, similarity, proximity, or functionality.

Information granulation can be viewed as a human way of achieving data compression and it plays a key role in the implementation of the strategy of divide-and-conquer in human problem-solving [**?**]. Objects obtained as the result of information granulation or de-granulation are information granules [**?**, **?**, **?**].

Granulation is inherent in human thinking and reasoning processes. Information granules play an important role in human cognition, system modeling and decision-making activities. GrC provides an information processing framework where computation and operations are performed on information granules, and it is based on the realization that precision is sometimes expensive and not much meaningful in modeling and controlling complex systems. GrC is about representing, constructing, processing, and communicating information granules. The concept of information granules is omnipresent and this becomes well documented through a series of applications [**?**, **?**]. Information granules can be of different type of abstraction and they can be treated as hints for the control of IS's (or control of c-granules we will consider later on) to carry over them reasoning leading to the right decisions. Examples of information granules include indiscernibility or tolerance (similarity) classes, information or decision systems, clusters, decision rules, sets of decision rules, classifiers, and time series along with their various components (see, *e.g.*, [**?**, **?**, **?**]). In reasoning about data and knowledge under uncertainty and imprecision many compound information granules are used (see, *e.g.*, [**?**, **?**, **?**, **?**, **?**]).

The challenges arise in discovering the relevant information granules from the computations generated by IS's in relation to the problems they are aiming to solve. We will discuss this in more detail in the following sections.

In GrC are investigated information granules embedded in the abstract space. There is a large literature dedicated to GrC (see, *e.g.*, [**?**, **?**, **?**, **?**, **?**]). However, in [**?**] it is mentioned the lack of theoretical foundations of GrC. In the following sections we discuss some basic issues related to the foundations of GrC, in particular some issues related to questions like:

- How are information granules defined?
- What are the objects over which information granules are defined?
- How are different types of objects discovered over which information granules are defined?
- How is a language for expressing information granules chosen?
- How to choose measures to estimate the quality of information granules?
- How to control the generation of information granules in order to efficiently obtain high quality granules?
- Are information granules satisfactory for the design and analysis of intelligent systems (IS) dealing with complex phenomena?

12.2 From approximation spaces to networks of granular spaces and granular networks (networks of granules) over them

In this section, we introduce one of the fundamental concepts that is essential for developing the foundations of GrC. This concept concerns the generalization of the approximation space as it is considered in the rough set approach. As a reminder, in section 5.1 of chapter 5, we discussed the multirelational approach to rough sets. Any such approximation space consists of a universe of objects and a set of relations defined over that universe. Structures formed by a set S and an indexed family of relations R_α over S are called relational structures. Furthermore, the generalizations of approximation spaces discussed, *e.g.*, in [**?**] are special cases of relational structures. In GrC, it is necessary to consider a generalization of this concept to granular spaces, which consist of objects called granules. This generalization addresses the following issues:

- Granules in the universe vary in type, understood as properties of sets of granules constructed (or discovered) from basic (generic) granules.
- These types are developed through the control of IS's (or the control of granules, which will be discussed later). This construction is supported by reasoning based on data sets, domain knowledge databases, and interactions with humans.
- The construction of types begins with basic granules, from which more complex types of granules are formed using aggregation operations applied to the basic objects or to granules already generated by these operations.
- Granules of different types are connected to each other through interfaces consisting of relations (functions) that specify the relationships between objects of different types. These interfaces also contain tools for evaluating the quality of granules, such as inference rules that allow the properties of higher-level granules to be inferred from those of the types from which they were constructed, as well as inference rules that justify the robustness of granule construction.
- Certain relationships between objects of a given type are identified, such as inclusion, partial inclusion, or proximity of granules within the considered space.

- The constructions described above require the discovery of appropriate languages to express granules and their properties.
- A granular space is a dynamic structure that is modified by control strategies (of IS's or granules with control). In particular, besides the ability to construct new types of granules, it is essential for control to include operations for decomposing previously used granules. This is necessary, *e.g.*, when control determines that the current granular space is ineffective in finding solutions to specific problems to be solved.

One can observe some analogies in the discussed issues to feature extraction (feature engineering) in Machine Learning (ML), where it is necessary to discover a language for expressing features and then to select the relevant features expressed in that language [**?**, **?**]. The discovery of relevant granular spaces that allow for the generation of high-quality computations over granular networks is an important challenge.

Let us now look at some concepts in more detail.

First, we introduce generic granular spaces. Each generic granular space consists of information granules defined by indiscernibility or similarity classes of indiscernibility or similarity relations, respectively, from given multi-relational approximation spaces or information systems (see Section 5.1 in Chapter 5) together with relations of (partial) inclusion between such granules. More formally, any generic granular space GS is a pair

$$(G, Rel),$$

where G is a set of granules and Rel – a set of relations between granules – is defined relative to a given multi-relational approximation space $AS = (U, R)$ (where U is a finite set of objects and R is a set of indiscernibility or similarity relations over U) as follows.

- G is a set of granules, *i.e.*, pairs $g = (syn(g), sem(g)$, where
  - (i) $syn(g)$ is a syntax of granule g corresponding to $x \in U$ in the form of descriptors defining the signature of x;
  - (ii) $sem(g)$ denotes the semantics of $syn(g)$ in the form $[x]_r$, where $x \in U$, $r \in R$ and $[x]_r = \{y \in U : xry\}$ is the indiscernibility or similarity class defined by x.
- Rel is a set of relations over granules from G representing (partial) inclusion of indiscernibility or similarity classes of granules.

In the sequel we assume that for different moments t different approximation spaces $AS_t = (U_t, R_t)$ are provided and related to them families $F_{gen,t}$ of generic granular spaces and strategies of generation of new granular spaces and interfaces between them.

We assume that a family F_{gen} of generic granular spaces and strategies for generating new granular spaces and interfaces between them are given. Networks of granular spaces are elements of F_{gen} or are generated from F_{gen} by applying a finite number of times to the granular spaces of F_{gen} and already generated networks given transformations for generating new granular spaces and interfaces between them. Thus, any network of granular spaces is a generic granular space or a network generated from generic granular spaces.

Any strategy of generation of new granular network GN' from a given one GN is defined relatively to a distinguished granular space $GS_0 = (G_0, Rel_0)$ already generated from F_{gen}. The granular network GN is transformed to GN' using an extension of GS_0 to a new granular space GS' with granules defined by means of aggregation operations transforming granules from GS_0 (or Cartesian product of GS_0) to granules from GS' with possible use of constraints defined by relations from Rel_0. The interface created by these operations and constraints helps to establish relationships of properties of granules from these two granular networks. In the consequence, the semantics of information granules in the new granular space GS' is defined relative to information granules from the granular space GS_0. Hence, we have

Definition 12.1. Granular space GS constructed from a given already constructed granular space $GS_0 = (G_0, Rel_0)$ is a tuple

$$GS = (G, Rel),$$

where

- G is a set of information granules of the form $g = (syn(g), sem(g))$,
- Rel is a set of relations over G and
- the semantics sem of information granules G is defined in terms of granules from G_0, *i.e.*, $sem(g) \subseteq G_0$ for $g \in G$.

Let us now explain the concept of interfaces between granular spaces. These interfaces provide tools for reasoning from sensory measurements through successive levels of granularity, ultimately reaching the highest level of granularity. This process allows the approximation of complex, vague concepts and facilitates the realization of transformations related to actions and plans.

To simplify the reasoning, we consider two granular spaces GS, GS'. An interface between them is defined by a set of relations and aggregation operations over these two granular spaces, as well as some reasoning rules. More formally, and interface $Inter(GS, GS')$ (or, $Inter$, in short) between GS, GS' is a tuple

$$(Rela, Fun, Rul),$$

where

- $Rela$ is a set of relations over granules, *i.e.*, subsets of Cartesian product of granules from GS or GS'.
- Fun is a set of aggregation operations from the set of granules of GS (or Cartesian product of such sets of granules) into the set of granules of GS'.
- Rul is a set of rules for inferring properties of granules obtained by aggregation from properties of granules being aggregated. These properties are expressed by granules on the corresponding granulation layers.

Hence, this interface defines how granules from GS are related to granules of GS'. In particular, the interface specifies how some granules from GS' are constructed from granules from GS and some properties of these constructed granules from GS' are related to properties of granules from which they have been constructed.

One should note that $syn(g)$ represents a structure of granule g consisting of, *e.g.*, description of granular components of g as well as relations between them. For example, granule g may be constructed from some granules of G_0 consisting of granules representing training signatures of objects together with binary decisions for them and relevant conjunctions of descriptors of the form $a = v$, where a denotes an attribute and v its value as well as relations of inclusion of these granules into decision classes of a given binary decision system. These conjunctions of descriptors and heir links with decisions correspond to decision rules.The constructed new granule g may represent a rule based classifier with binary decision. Semantics *sem* is usually specified by a procedure for checking if granules from (a given subset of) G_0 (or cartesian product of G_0) belong to $sem(g)$. For the considered example $sem(g)$ represents the set of objects represented by signatures accepted by the classifier, *i.e.*, signatures of objects for which the classifier returns the value 1.

Relations from *Rela* are used as constraints in aggregation of granules by aggregation operations from Fun. The rules from *Rul* are making it possible to reason about constructed granules by aggregation operations. Let us consider an illustrative example of such rule:

$$\frac{r_1(g_1), \ldots, r_k(g_k)}{r(f(g_1, \ldots, g_k))}.$$

This rule has the following meaning: if granules $g_1, \ldots, g_k$ from GS have properties $r_1, \ldots, r_k$, respectively then their aggregation $f(g_1, \ldots, g_k)$ by operation $f \in Fun$ has property r. Granules $g_1, \ldots, g_k$ may represent, *e.g.*, clusters and $f(g_1, \ldots, g_k)$ may represent their aggregation. This rule expresses a robustness of the construction performed by $f : G^k \to G'$. This may be explained formally by

$$f(sem(r_1) \times \ldots \times sem(r_k)) \subseteq sem(r),$$

where *sem* defines semantics of relations, *i.e.*, $sem(r_i) \subseteq G$, for $i = 1, \ldots, k$, $sem(r) \subseteq G'$ and $f(sem(r_1) \times \ldots \times sem(r_k))$ is the image of the Cartesian product $sem(r_1) \times \ldots \times sem(r_k)$ under the operation f.

It is worthwhile to mention that the relations $r_1, \ldots, r_k, r$ may be defined by aggregation operations over granules from G and G'. For example, they may be defined using similarity or tolerance over basic granules from G and G' by experts or discovered from data. If $sim \subseteq G \times G$, and $sim \subseteq G' \times G'$ are similarity relations in G, G', respectively and $g_1^0, \ldots, g_k^0 \in G$, $g^0 \in G'$ are distinguished granules then

$$sem(r_i) = \{g \in G : sim(g_i^0, g)\}$$

and

$$sem(r) = \{g \in G' : sim(g^0, g)\}.$$

This means that $r_i(g)$ holds iff $sim(g_i^0, g)$ holds for $i = 1, \ldots, k$ and $g \in G$ as well as $r(g)$ holds iff $sim(g^0, g)$ holds for $g \in G'$.

Rules as described above can be composed under natural constraints. For example (see Fig. 12.1), from rules

$$\frac{r_1(g_1), \ldots, r_k(g_k)}{r(f(g_1, \ldots, g_k))} \text{ and } \frac{r'_1(g'_1), \ldots, r'_l(g'_l)}{r'(h(g'_1, \ldots, g'_l))}$$

and a constraint $r(g) \rightarrow r'_1(g)$ one can derive a rule

$$\frac{r_1(g_1), \ldots, r_k(g_k), r'_2(g'_2), \ldots, r'_l(g'_l)}{r'(h(f(g_1, \ldots, g_k), g'_2, \ldots, g'_l))}.$$

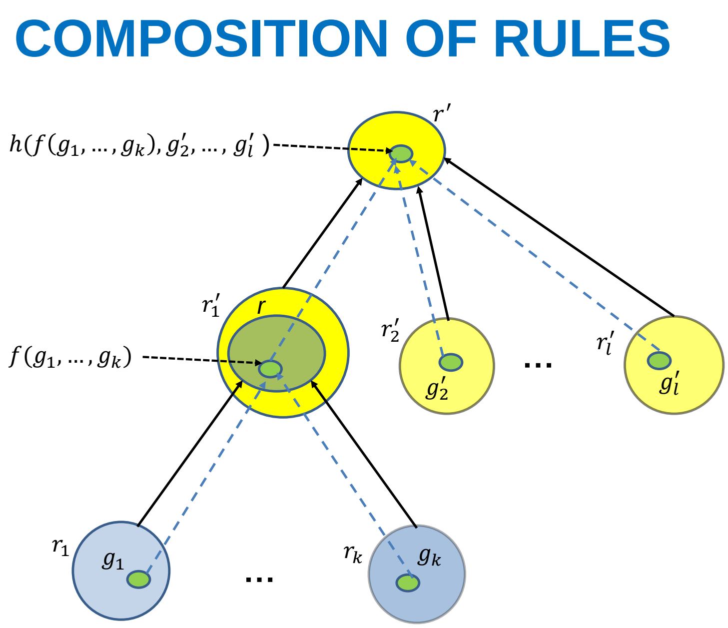


Fig. 12.1 Composition of rules.

One may refer to the distinction between reactive and deliberative control in agents or robots [**?**]. Reactive control relies on rules that provide decisions based on sensory measurements, while deliberative control involves reasoning that leads to decisions. In our case, reasoning is based on searching for rules that, through multiple compositions, yield a decision. This may require the control system to engage in additional reasoning to determine what new measurements are necessary to obtain. The approximate reasoning schemes related to the composition of reasoning rules over granules are discussed in [**?**].

Networks of granular spaces are constructed from a family of granular spaces and interfaces between them.

The existing networks of granular spaces may be extended using strategies making it possible to define a new granular spaces, establishing new interfaces of them with the already defined granular spaces and constructing on this basis new granular networks.

Networks of granules over a given network of granular spaces Net (or network of granules, in short) consists of selected instances of granules from Net together with a set of instances of facts expressed by relations from Net on some selected instances of granules from Net. Facts have the following form $r(g_1,\ldots,g_k)$, where r is a relation form Net, k denotes the arity of r and $g_1,\ldots,g_k$ are instances of granules from Net.

12.2.1 Illustrative example of granular spaces and networks of granules for the Pawlak rough set model and its generalizations

In this section, we recall the Pawlak model [**?**, **?**, **?**] and using this model we distinguish some important components of approximation process related to it. In the abstract setting the approximation space was defined as a pair (U,R), where U is a finite set of objects and $R \subseteq U \times U$ is an equivalence (indiscernibility) relation. This relation defines a partition of U into equivalence classes called indiscernibility classes. These classes $[x]_R$ (for $x \in U$) are elementary granules. The aggregation operation on granules is defined producing new granules from families of elementary granules by taking the union of them. One should note that in the Pawlak model the universe U is treated as given set and the issues how objects form this set are perceived in the physical world are outside of this model.

Together with the indiscernibility relation is considered a partition of U into decision classes (decision granules). Let us denote the corresponding equivalence relation to this partition by R_d. In this way we obtain a triple $AS = (U,R,R_d)$ called approximation space. For any decision granule $X \subseteq U$ is defined its lower approximation relative to R by $LOW(AS,X) = \{x \in U : [x]_R \subseteq X\}$[1] and the upper approximation of X by $UPP(AS,X) = \{x \in U : [x]_R \cap X \neq \emptyset\}$. The set $Bd(AS,X) = UPP(AS,X) \setminus LOW(AS,X)$ is called the boundary region of X relative to R. If $Bd(AS,X) \neq \emptyset$ then X is called rough (relative to R), otherwise X is crisp (relative to R).

Hence, one can observe that in this model we deal with two granular spaces G_R and G_{R_d} corresponding to the partitions defined by relations R and R_d: $G_R = (\{[x]_R\}_{x\in U}, agg)$, where $agg : P(\{[x]_R\}_{x\in U}) \rightarrow P(U)$ and $agg(Z) = \bigcup Z$ for $Z \subseteq \{[x]_R\}_{x\in U}$ and $G_{R_d} = U/R_d$ where $U/R_d = \{[x]_{R_d}\}_{x\in U}$. The first component of G_R contains a family of elementary granules defined by R and the second component is

[1] We will write also $LOW(R,X)$ instead of $LOW(AS,X)$.

an aggregation operation generating definable granules from elementary granules. In the case of G_{R_d}, we have only one component with the family of decision granules.

The granular spaces G_R, G_{R_d} are linked by an interface $Inter(G_R, G_{R_d})$ including some relations between granules from these two granular spaces as well as some reasoning methods making it possible to find relevant relationships between granules from these spaces. In the case of the discussed model this interface has a simple form and is defined by

- the set theoretical inclusion relation between definable granules from G_R and granules from G_{R_d},
- The set theoretical 'non-empty intersection' relation between definable granules from G_R and granules from G_{R_d} and
- *Res* – reasoning module (RM) consisting of reasoning tools supporting, *e.g.*, definition of approximation regions such as lower approximations of decision classes or boundary regions; construction of searching heuristic for the maximal (relative to set inclusion) definable granules included into decision granules for the lower approximation of decision granules and minimal (relative to set inclusion) definable granules including decision granules for the upper approximation of decision granules.

In Fig. 12.2 is presented an example of granular space for the basic Pawlak rough set model.

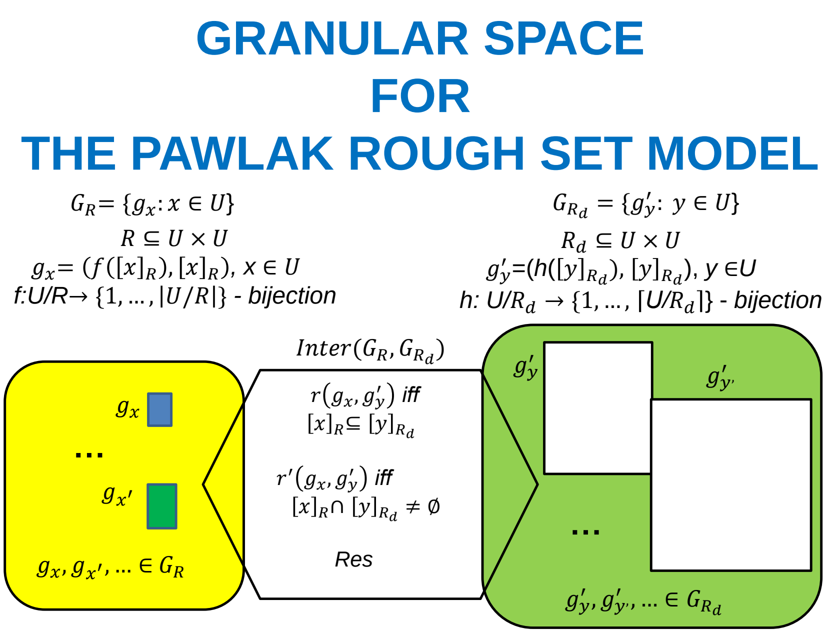


Fig. 12.2 Example of granular space for the Pawlak rough set model.

In the example, two granular spaces G_R and G_d are shown. For simplicity only elementary granules defined by indiscernibility classes are considered. For granules $g_x = (syn(g_x), sem(g_x))$, where $x \in U$, $syn(x)$ is the name of the indiscernibility class defined by x (using any bijection $f : U/R \to \{1, \ldots, | U/R |\}$) and $sem(x)$ is the indiscernibility class defined by .. In the interface two relations are, namely the inclusion relation between semantics of granules as well as the hitting relation making it possible to check if semantics of two granules have non-empty intersection. *Res* consists mechanisms for reasoning leading to construction of approximation regions, corresponding to the lower approximation and upper approximation of decision classes and the boundary region.

From the granular space presented in Fig. 12.2 granular networks are generated. In the considered example, this is realized by selection of some granules from these two granular spaces G_R and G_d and by adding links between these granules pointed out by reasoning mechanism related to construction of approximation regions.

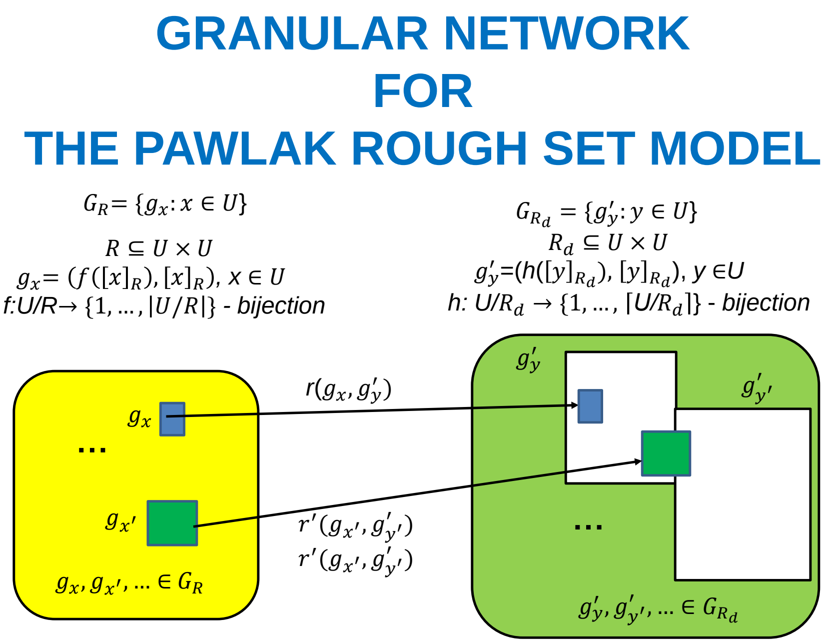


Fig. 12.3 Example of granular network for the Pawlak rough set model generated from the network of granular spaces presented in Figure 12.2.

The illustrative example in Fig. 12.3 showcases a simple granular network summarizing our considerations. The network is built from two granular spaces, G_R and G_{R_d}, which consist of elementary information granules (equivalence classes). These granules are defined by conditional attributes that determine the relation R in the universe of objects U. Additionally, a decision attribute determines the relation

R_d in U. Both the description of these elementary information granules and their meaning within the universe U are defined in an abstract (mathematical) space. The network itself is constructed from two sets of elementary granules, one from each space. Arrows connect these granules, indicating two types of relationships: inclusion or partial inclusion. In general, these links can represent more complex relationships between granules from different, more advanced granular spaces. For instance, information granules might contain details about the location and time of a measurement stored in their informational layers. They could also concern properties of various segments of multi-time series data or the aggregation of this data into clusters or more complex structural objects. Notably, these granules can also include specifications for so-called associations, which allow the control of IS's (c-granules) to generate their meaning in the physical world (see Section **??** for further details). Links can even represent relationships between different granules or their parts that are dependent on time.

Specifications may be in the form like in the case of information granules but also in the form of specification of association. In the case of association the predicted results of physical realization may be different from the real ones obtained in perception of the physical world what causes that control of c-granule should be aware of this and adapt of the currently used complex game or modify accordingly the target goals, if modification can't lead to possibility that the generated computation will be of a satisfactory quality.

In this way, granular computations have states representing granular networks and the discovered links. The IS control specifies the realized transitions between states. In the final state, all elementary granules are linked to decision granules and approximations of decision granules are computed. This simple example illustrates an important aspect of approximation processes running over structures defined by granular spaces and interfaces between them along which are constructed approximations of specified target granules. In the discussed example decision granules play the role of target granules. In different applications different complex objects of the high quality required to be constructed, *e.g.*, learning algorithms, classifiers, clusters, complex physical or abstract objects, or the whole granular computations with the required properties.

The current state of approximation process is represented by granules from G_R and G_{R_d} which have been used in reasoning (and not yet forgotten) and already established links between granules from G_R to granules from G_{R_d} representing inclusion. For example, granules from G_R linked to at least two decision granules from G_{R_d} belong to the boundary region.

In applications, the relation R is defined by a finite set of attributes A, where any $a \in A$ is a function from U into a value set V_a. For any $x \in A$, $Inf_A(x)$ denotes the signature of x relative to A defined by $\{(a, a(x)) : a \in A\}$. For any $x, y \in U$ it is assumed that xRy if and only if $Inf_A(x) = Inf_A(y)$. Then R is denoted by R_A. In the considered case, the definition of granules is modified accordingly. Elementary granules are equivalence classes of R_A labeled by signatures of objects defining them, *i.e.*, for any $x \in U$ the equivalence class $[x]_{R_A}$ is labeled by $Inf_A(x)$. Instead of signature $Inf_A(x)$ one can consider the conjunction $\bigwedge_{a\in A} a = a(x)$. The semantics of

$\bigwedge_{a\in A} a = a(x)$ is defined by $\| \bigwedge_{a\in A}(a = a(x)) \|_U = \{y \in U : a(y) = a(x) \text{ for } a \in A\}$. Obviously,

$$[x]_{R_A} = \| \bigwedge_{a\in A} (a = a(x)) \|_U \ .$$

Hence, the granule g_x corresponding to $x \in U$ can be treated as a pair $(\bigwedge_{a\in A}(a = a(x)), [x]_{R_A})$, with its syntax $\bigwedge_{a\in A}(a = a(x))$ and the semantics $[x]_{R_A}$. It is worthwhile mentioning that this approach is important when one would like to consider inductive extensions of approximation spaces. In this case the aggregation operation in G_{R_A} is restricted to unions of elementary granules defined by formulas obtained by dropping some conditions from Inf_A. Among such formulas the computational building blocks for approximation of decision concepts from G_{R_d} are selected. Any equivalence class $[x]_{R_A}$ is linked to a formula α_x obtained from $\bigwedge_{a\in A}(a = a(x))$ by dropping the maximal number of conditions under assumption that the semantics of α_x has non-empty intersection with exactly the same decision granules from G_d as the semantics of $\bigwedge_{a\in A} a = a(x)$. Then, *e.g.* for $X \in U/R_d$ its lower approximation is obtained by the union of semantics of all formulas α_x included in X. Hence, in this case we consider reasoning supporting linking the elementary granules $[x]_{R_A}$ from G_{R_A} with α_x and α_x with decision classes for $x \in U$.

The process of inducing approximations of decision granules can be realized on tolerance decision systems (for more information about tolerance rough sets see e.g. [**?**, **?**]). The three-way rough set based approach for approximation of decision granules in Intelligent Systems is discussed in [**?**].

An important aspect of approximation is related to the quality of resulting approximation of decision granules from G_{R_d}. In the discussed model, the positive region of G_{R_d} relative to G_R, i.e.,

$$POS_{G_R}(G_{R_d}) = \bigcup_{X\in G_{R_d}} LOW(R,X)$$

is used to define the quality of approximation of G_{R_d} relative to G_R by

$$\gamma = \frac{|POS_{G_R}(G_{R_d})|}{|U|},$$

where $|U|$ denotes the cardinality of U.

However, in applications it may be important also the description length of representation of the constructed approximations. Shorter descriptions may be preferred over longer from the point of view of readability or explainability of approximations. In this case, reasoning methods based on reduction of the length of representation may be applied. In particular, Boolean reasoning can be applied to support search for the minimal description of approximations of decision granules or classification defined by them. Efficient methods for generation of different kinds of reducts have been developed and used for this purpose [**?**, **?**, **?**, **?**]. In the discussed case, one can add a second component to the quality of approximation of decision granules, namely the component related to the description length.

The quality of approximation in the case of Pawlak's model is related to the given universe of objects U. This approach can be extended to the case when U is only a training sample and the derived approximations should be also extended on testing objects (cases) not seen so far, like in Machine Learning. This issue will be discussed shortly in one of the following sections. We would like to only stress here that in such a case one more component related to testing the derived models of approximation should be developed.

Let us illustrate the ideas of constructing granular spaces and granular networks over the by one more example illustrated in Fig. 12.4 and Fig. 12.5, respectively.

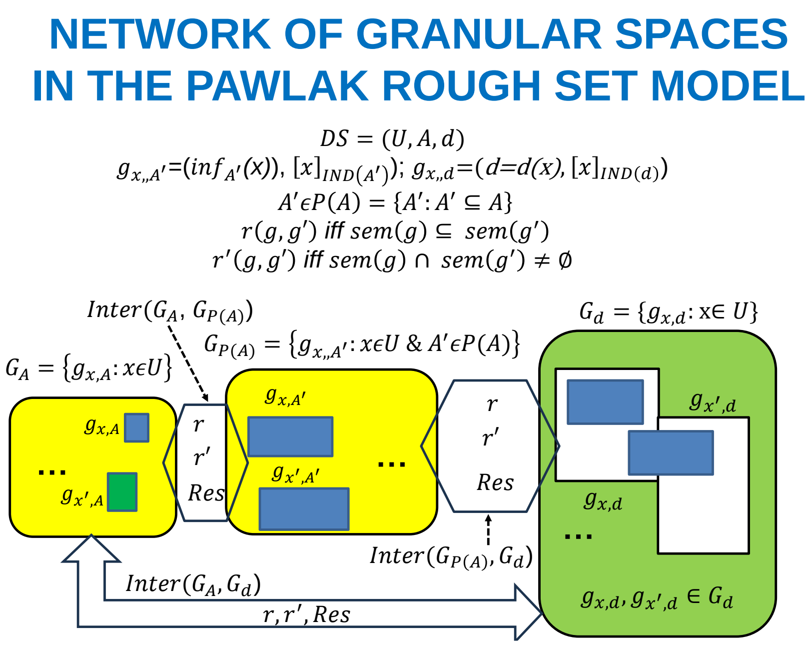


Fig. 12.4 Illustrative example of granular space.

The granular space G_A consists elementary granules defined by indiscernibility classes of the indiscernibility relation $IND(A)$. In the granular space $G_{P(A)}$ we have all indiscernibility classes of indiscernibility relations $IND(A')$, where $A' \subset A$. Hence, the number of granules in $G_{P(A)}$ is exponential relative to the cardinality of A. The interface between the granular spaces G_A and $G_{P(A)}$ consists relation of inclusion of semantics of granules. In the granular space G_d granules are decision classes defined by the indiscernibility relation $IND(d)$. The interface between the granular spaces $G_{P(A)}$ and G_d consists relation of inclusion of semantics of granules. One of the rules of reasoning of the considered network of granular spaces is related to transitivity of inclusion.

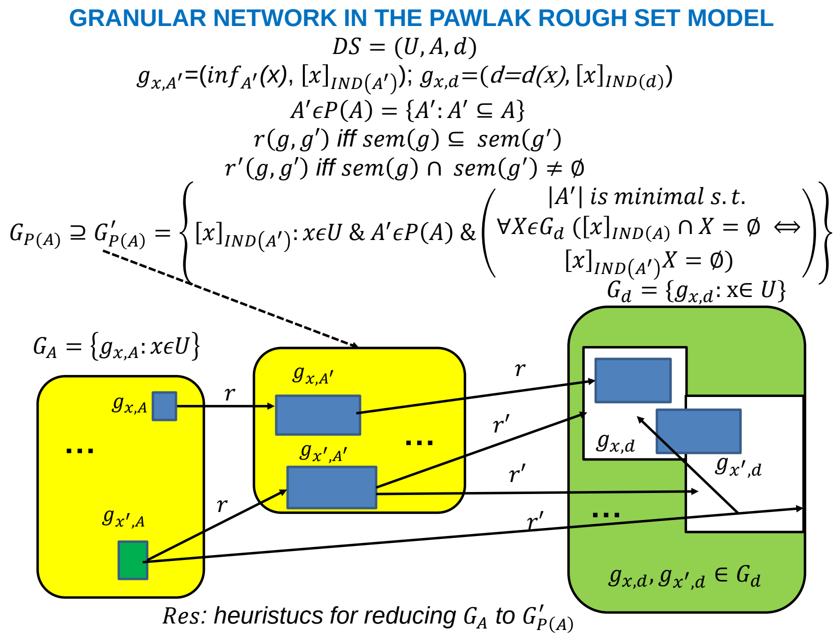


Fig. 12.5 Illustrative example of granular network over granular space presented in Fig. 12.4.

In the granular network presented in Fig. 12.5, the second component corresponding to $G'_{P(A)}$ is obtained by reducing the number of granules to such indiscernibility classes which are maximal relative to set inclusion (minimal with respect to the number of descriptors describing them) and also preserve 'hitting' property expressed by non-empty intersection with decision classes of granules from G_A. Granules from $G'_{P(A)}$ are obtained by generalization (by dropping some descriptors over attributes from A describing them). The precise definition of hitting' property is presented in Fig. 12.5. To the relation of inclusion is added the relation of non-empty intersection of granules. Let us observe that the selection process of granules in the second component can be supported by Boolean reasoning and the discussed example is related to minimal decision rules (see, *e.g.*, [**?**, **?**]). Obtained by Boolean reasoning heuristics may be parts of interfaces in the considered example.

It should be noted that in networks of granular spaces and granular networks in different components may appear granules with semantics in different universes of objects or granules. To explain this we present illustrative examples.

Our first example concerns a role of interface in communication of different granular spaces or networks. Let us consider an information system $IS = (U, A)$ with $a : U \to V_a$, where V_a is a value set of attribute a. We also assume that for any attribute a is given a relational structure $Rel_a = (V_a, R_a)$, where R_a is a set of relations over V_a, *i.e.*, for $r \in R_a$, $r \subseteq V_a^{ar(r)}$, where $ar(r)$ is called the arity of relation r. For example, V_a can be a set of reals and R can consists the relation r of linear order over V_a. Moreover, together with the relational structure Rel_a we consider a set of formulas F_a and its semantics $sem : F_a \to P(U)$ assigning to any $\alpha \in F$ its meaning $sem(\alpha) subsetU$. In this way is defined a granular space GS with the set of granules G equal to a family of granules $(\alpha, sem(\alpha)$ for all attributes $a \in A$, families F_a and semantics sem. As relations between granules from G we consider inclusion relation. One can extend this example by considering instead of a set of formulas F_a a family of such sets. Moreover, one can consider another extension by considering relational structures over Cartesian products of different value sets of attributes and relations over such products. The new defined granular space can be treated as a large potential set of sensory granules from which the next granular space should select attributes used for perceiving objects objects. The next granular space corresponds to a decision system $DS = (U, Atr, d)$, where Atr is a set of attributes selected from the granular space GS. In this new granular space GS' the set of granules Atr is a finite set of formulas selected from granules in GS together with its semantics. In this granular space, one can consider relation inclusion between granules defined by these granules and granules defined by the decision d. From the described granular network consisting GS, GS' granular networks are created by selecting some granules from GS and linking them to granules in GS', in particular to granules defined by the decision attribute d.

Our previous example pertained to sensory granules. The example we are going to present is related to discovery of new attributes from already defined ones. Let us consider a granular space related to an information system $IS = (U, A)$ with the set of granules G defined by indiscernibility classes of $IND(A)$ and the empty set of relations between granules. We assume that there is a one distinguished attribute $t \in A$ representing time with the values set $V_t = \{1, \ldots, n\}$. We define the second granular space GS', in which the set of granules G' consists of granules from G as well as granules generated by operations from a given set Op applied to IS. For our example, we assume that Op has only one operation. First of all, any object of the semantic space U_{time} of these granules is a time window of the length T, *i.e.*, a sequence of the length T of terms being value vectors of attributes obtained from signatures of objects from U such that each next term of the sequence has the value of t one greater than the preceding term. Over such objects are defined granules which are sets of time windows with semantics expressed by formulas from a given set of formulas. Using these new generated granules we would like to approximate decision classes of a decision $d_{time} : U_{time} \to V_{d_{time}}$. These two granular spaces GS, GS' are linked by interface $Inter(GS, GS')$ consisting of relations of inclusion and hitting as well as Res with mechanisms supporting reasoning for selection of the relevant granules from GS' for approximation decision classes of the decision d. Certainly, one can generalize this example by considering more than one operation or by considering new granules obtained by application of a finite number of times operations to IS or already generated granules.

The reader familiar with Machine Learning (ML) will observe that the discussed examples pertain to the main considered in ML problems: feature selection and extraction (feature engineering) [**?**, **?**, **?**, **?**]. The mentioned extension of the second example is corresponding to hierarchical learning (see, *e.g.*, [**?**, **?**]). The main challenge is to provide reasoning mechanisms for developing efficient heuristics searching for relevant granules in granular space *GS* making it possible to construct the high quality approximations of complex vague concepts at the highest level of the hierarchy.

Granular networks are modified by control mechanism of granules to achieve specific target goals. Based on the current granular network, a transformation is selected by control on the basis of its set of rules to change it into a new granular network that represents the next step in granular computation. This control aims to generate granular computation over granular networks that satisfy a given specification (to a satisfactory degree). This specification may pertain to the final state of the computation or encompass the entire computation, depending on the task performed by the granule. The quality of computations is assessed using selected quality measures. Various quality measures have been developed in Machine Learning (ML) and across different application areas, such as risk management, algorithmic trading, and drug discovery [**?**, **?**, **?**, **?**]. For instance, in ML, the Minimum Description Length (MDL) Principle [**?**] and various measures based on the confusion matrix [**?**] are well-known for estimating the quality of learning algorithms and classifiers. In some applications, it may be necessary to maintain certain properties (*e.g.*, the 'safety' of the system) represented by a granule as invariant throughout the computation over granular networks. In other cases, it may be required to enrich a granular network with properties that express, *e.g.*, that a required drug has just been synthesized under constraints related to the costs of executing the computation.

More details on control of granules will be presented in a subsequent section. Before that, we will provide an outline of Interactive Granular Computing (IGrC). In IGrC, we address complex granules (shortened as c-granules) that connect objects from both abstract and physical realms. It is worthwhile to mention that IS's can be viewed as compound c-granules (with control). The extension of GrC to IGrC is essential for addressing the perception of situations in the physical world as perceived by IS's. Understanding perceived situations in the physical world is crucial for IS's, especially those dealing with complex phenomena, for making by them the right decisions pertained to these situations. These decisions concern transformations. The implementations of specification of transformations in the physical world are aiming to accordingly control granular computations of the considered IS (or control of c-granule) for achieving the target goals of IS.

For IS's dealing with complex phenomena two kinds of granules should be considered: (i) information granules defined in GrC by formulas over the relevant language with semantics in the space of granules at a given hierarchical level and (ii) complex granules (c-granules) in IGrC with physical semantics realized in the real-physical world ensuring perceiving some properties of objects and configurations of physical objects as well as their interactions.

It is essential to recognize that the control of IS must continuously interact with the physical environment to respond to changes in perceived situations caused by environmental interactions. Simply collecting data from these interactions once is inadequate for developing a high-quality model. The models created must be adapted to account for any perceived substantial changes in the physical space. Additionally, the control of IS should be equipped with appropriate reasoning techniques to support its behavior. These topics will be explored in greater detail in the following sections.

12.3 Interactive granular computing (IGrC)

12.3.1 IGrC - motivation, basic intuition and concepts

In this section, we present an intuitive explanation of some basic concepts related to IGrC. We start from motivations for generalization of GrC to IGrC.

The GrC model is inadequate for modeling IS's that address complex phenomena. The primary reasons for introducing granules with both informational and physical layers in the IGrC framework can be summarized as follows.

- Any IS dealing with complex phenomena must engage with the physical world to accurately perceive situations within it. The GrC model operates solely in an abstract space, yet interactions with the real physical world are essential for IS's to effectively interpret situations. Relying exclusively on abstract modeling of perception is insufficient. For modeling perception, it is crucial to align language and reasoning with perception and action [**?**]. Modeling perception requires granules that integrate both informational and physical layers, ensuring these layers interact appropriately. This is the focus of IGrC.
- Constructing high-quality data models, granules, and computational building blocks necessary for cognition [**?**] (*i.e.*, understanding perceived situations) based on a predefined information system (data set) is not adequate (see, *e.g.*, the opinion cited below [**?**]). Continuous interaction between IS's and complex phenomena as well as proper steering them in the physical world is essential for understanding perceived situations.
- Engaging in dialogues with domain experts is vital for IS's that address complex phenomena [**?**, **?**].

One of the most important issue related to complex systems as well as IS's dealing with complex phenomena is understanding interactions [**?**]:

> [...] *interaction is a critical issue in the understanding of complex systems of any sorts: as such, it has emerged in several well-established scientific areas other than computer science, like biology, physics, social and organizational sciences.*

Interactions take place between physical objects. Hence, IS (c-granule with control) should have skills to implement in the physical world specifications of configurations of physical objects and perceive properties of these objects as well as interactions between them. These configurations are created using special transformations called associations. Control of c-granule is equipped with the implementational module (IM) for (i) constructing physical semantics of associations consisting of configurations of the physical objects, (ii) initializing interactions of such configurations (by encoding relevant information in the physical objects which are directly accessible by control), (iii) perceiving some properties of objects in these configurations and their interactions (by decoding information from the physical objects directly accessible by control or by judgment (a special kind of reasoning) based on already perceived information, domain knowledge bases or physical laws (computations should be dependent on physical laws!). Hence, information granules from GrC are generalized to complex granules (c-granules, in short) in IGrC with granules consisting of two layers: informational and physical. Control of c-granule provides possibility of interaction between these two layers. It consists of a set of rules which are used for triggering realization of transformations in the abstract layer or associations in the physical layer with the support of IM. In the consequence, computations over granular networks are generalized to granular networks over c-granules. From the discussion above it follows that crucial for the IGrC model is also perception model. Let us recall that in IGrC we follow the already cited idea of perception presented in [**?**].

The IGrC model leverages existing partial results from various fields (including multi-agent systems, perception and action, machine learning, natural language processing, federated learning, cognitive networks, smart cities, cyber-physical systems, complex adaptive systems etc.) by an attempt of putting them into sync.

We emphasize the fact that IS's often are dealing with complex phenomena in the physical world what requires a new kind of modeling for solving problems with the design and analysis of IS. Particularly, this requires grounding the approach on the relevant computing model.

We have selected the IGrC model as the basis for developing theoretical foundations for the design and analysis of IS's dealing with complex phenomena in the physical world. In the considered case, according to opinions of top researchers, classical mathematical modeling is not satisfactory. For example, in [**?**] one can find the following opinion by Frederick Brooks, Turing award winner:

> *Mathematics and the physical sciences made great strides for three centuries by constructing simplified models of complex phenomena, deriving, properties from the models, and verifying those properties experimentally. This worked because the complexities ignored in the models were not the essential properties of the phenomena. It does not work when the complexities are the essence.*

Let us also cite another opinion by Vladimir Vapnik expressing the need for considering the physical world as the basis for computations related to learning of problems in applications [**?**]:

> [...] *further study of this [learning]* phenomenon requires analysis that goes beyond pure mathematical models. As does any branch of natural science, learning theory has two sides:
>
> - *The mathematical side that describes laws of generalization which are valid for all possible worlds and*
> - *The physical side that describes laws which are valid for our specific world, the world where we have to solve our applied tasks.*
>
> [...] *To be successful, learning machines must use structures on the set of functions that are appropriate for problems of our world.* [...] *Constructing the physical part of the theory and*

unifying it with the mathematical part should be considered as one of the main goals of statistical learning theory. [...] *In spite of all results obtained, statistical learning theory is only in its infancy...*

Accordingly to Vapnik [**?**], that there are many many branches of this theory that have not yet been analyzed and that are important both for understanding the phenomenon of learning and for practical applications. Definitely, one of such area of the research should consider the necessity of linking the abstract world of mathematics with the physical world. This is also related to the the grounding problem investigated in psychology [**?**, **?**, **?**, **?**].

It is also worthwhile to cite here explanation of the concept of complex systems [**?**] created by IS's dealing with complex phenomena:

Etymologically: complexity – plexus in Latin (interwoven). Complex system: the elements are difficult to separate. This difficulty arises from the interactions between elements. Without interactions, elements can be separated. But when interactions are relevant, elements co-determine their future states. Thus, the future state of an element can not be determined in isolation, as it co-depends on the states of other elements, precisely of those interacting with it.

This definition and the opinion presented above have significant implications for the design and analysis of IS's and development of the IGrC model. It is particularly important to note that IS's must make decisions about situations in the physical world that involve complex phenomena. Therefore, it is inadequate to model such IS's based solely on single models derived from isolated discoveries, which rely on data and knowledge about fragments of physical reality disconnected from their environmental interactions. For IS's, it is essential to continuously perceive various relevant aspects of different fragments of the physical reality, enabling them to acquire the knowledge necessary for developing adaptive strategies to modify the models currently in use. This highlights the critical importance of perception in the IGrC model, especially concerning c-granules or IS's treated as a special kind of c-granules with control. This requires to develop for IS's attention mechanisms supported by reasoning techniques concerning queries related to *what*, *when*, *where*, *how* etc. to perceive.

Aristotle already was aware of importance of the role of attention (see, *e.g.*, [**?**] and Fig. 12.6) in interaction with the physical world. In particular, in [**?**], one can read:

Spoken words are the symbols (symbola) of mental experience (pathemata) and written words are the symbols of spoken words. Just as all men have not the same writing, so all men have not the same speech sounds, but the mental experiences, which these directly symbolize, are the same for all, as also are those things (pragmata) of which our experiences are the images (homoiomata).

It is worthwhile to mention here opinion from [**?**]:

The Turing test, as originally conceived, focused on language and reasoning; problems of perception and action were conspicuously absent. The proposed tests will provide an opportunity to bring four important areas of AI research (language, reasoning, perception, and action) back into sync after each has regrettably diverged into a fairly independent area of research.

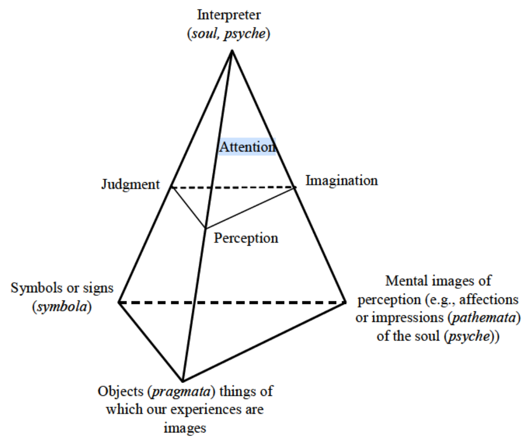


Fig. 12.6 Aristotle tetrahedron [**?**]

One of the consequences of this opinion is that in the discussed IGrC model it is necessary to have objects linking abstract and physical objects. Hence, this model can't be closed only in the abstract space.

The idea of complex granules (c-granules, for short) in IGrC is very well expressed as computational building blocks for cognition by Leslie Valiant [**?**]:

> *A fundamental question for Artificial Intelligence is to characterize the computational building blocks that are necessary for cognition.*

The computing IGrC model should enable continuous interaction between the designed on the basis of this model system and the physical environment, allowing for the collection of relevant data which can be used to infer data models temporarily characterized by the high quality.

The basic objects in IGrC are complex granules (c-granules, for short). They consists two layers: informational and physical. In the informational layer is stored information about perceived situations as well as specifications of tasks realized over them as well as information about the expected results of realization of these tasks in different parts of the physical world. This information is labeling specifications of spatio-temporal windows (addresses) describing regions of the physical space where the information is perceived.

The physical layer of c-granule consists parts like soft_suite, link_suit and hard_suit. Soft_suit consists of physical objects directly accessible, *i.e*, objects which properties can be decoded by measurements into the information layer or objects into which some relevant information from the information layer can be encoded. This is realized by special elementary c-granules generated by control of c-granules. Information about physical objects which are not directly accessible is inferred by reasoning tools using knowledge bases or physical laws. Hence, computations in IGrC depend on physical laws contrary to the Turing model [**?**]:

> *It seems that we have no choice but to recognize the dependence of our mathematical knowledge* [(...)] *on physics, and that being so, it is time to abandon the classical view of computations as purely logical notion independent of that of computation as a physical process.*

In link_suit are physical objects used for transmission of interactions from soft_suit to hard_suit and hard_suit contains physical objects to be perceived according to the specifications labeling spatio-temporal windows represented in the information layer. One should note that, c-granules are under control of other c-granules or their own control.

12.3.2 Control of c-granules

For simplicity of reasoning, we consider here the case when c-granules are under the control of IS which can be treated as a higher order c-granule. This control is responsible for generating computations of IS. The computations are sequences of c-granules (or their networks including information about relationships of other c-granules which are parts of the networks). IS is aiming to generate such computations realizing in the best way the task of IS, *i.e.*, they are aiming to generate computations along which the high quality approximate solutions of problems to be solved by IS are constructed. These approximate solutions of problems may concern classifiers or compound physical objects like sensors, robots or chemical components.

In each step of computation, the control module (CM) of IS verifies whether the information about the current situation is satisfactory to initiate the appropriate transformation of the current c-granule configuration in the form of network of c-granules. This may involve suspending, modifying existing c-granules, or generating new ones. CM includes a special implementation module (IM) responsible for realization the transformation specifications in the physical world. In essence, the IM realizes the so-called physical semantics of the transformations' specifications.

Here's an idea how it works: The specifications of these transformations are included on the right-hand side of rules located in the rule module (RM) of CM. In each step, the control checks if the information about the currently perceived situation matches the left-hand side of any rules. If there's a match, the RM uses reasoning mechanisms to resolve any conflicts among these rules. If the conflicts can be resolved, the c-granule control selects from RM the rule for execution. Otherwise, it suggests to gather more information about the perceived situation. This can be done by:

- realization by IM of some associations in the physical world concerning, *e.g.*, measurement of values of some attributes in some specified regions of the physical space or
- reasoning with the use of some information granules representing a partial knowledge about the physical environment of c-granule; these information granules can be treated as domain databases and reasoning mechanisms of con-

trol applied to these domain databases together with information about the currently perceived situation in the physical world allow the control of c-granule to improve understanding this situation (*e.g.*, it can be knowledge about the properties concerning the transmission of encoded information in configurations of physical objects).

Let us consider an illustrative example concerning the mentioned above reasoning with the use of domain data bases. Fig. 12.7 depicts reasoning performed on information granules, including the knowledge base (denoted by KB). In the first step, based on information about the current situation (inf) perceived in the physical world, the control aims to extract the relevant piece (fr) from KB that could be used to extend inf in a useful way. This requires a proper knowledge representation in KB to support the control's efficient extraction of fr. Next, using inf and fr, the control reasons to return an extended version of the information about the perceived situation, denoted by inf'. This extension is expected to be helpful, *e.g.*, in resolving conflicts between control rules that match inf'. Due to its generality, this type of reasoning may require decomposition through several levels before the control can realize it using accessible information granules.

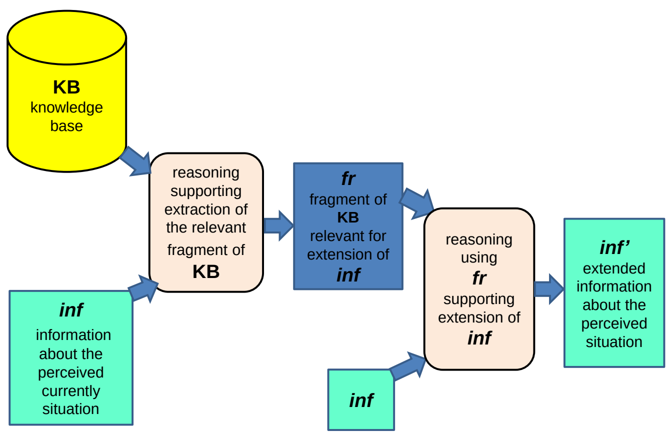


Fig. 12.7 Reasoning performed on information granules, including the knowledge base KB and information about the current situation (inf) perceived in the physical world.

The execution of the rule by CM involves realizing the transformation specification from its right-hand side. The IM is responsible for carrying out this transformation in the physical world. The set of rules in the RM can be viewed as a complex game involving intricate rules composed out of vague concepts (learned by the IS control) labeled by transformations. The rules in RM are adaptively changed according to perceived changes by CM. For more details about CM, the reader is referred to [**?**].

Fig. 12.8 illustrates the concept of the basic cycle of c-granule control behavior. This c-granule consists of two layers: informational (il) and physical (pl). The sym-

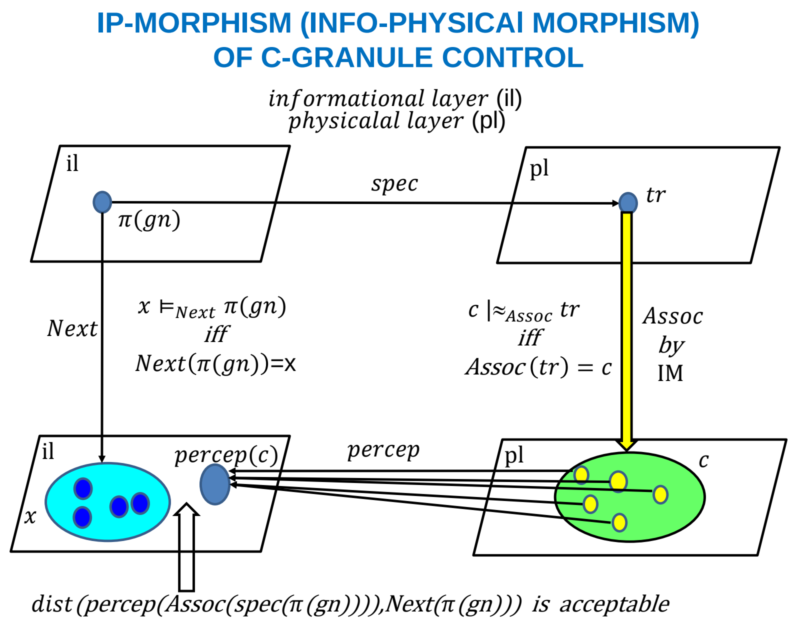


Fig. 12.8 Basic cycle of c-granule control: ip-morphism.

bol gn represents a granular network that reflects the currently perceived situation in physical space. The projection $\pi(gn)$ denotes its representation on the informational layer (il), comprising a family of specifications for spatio-temporal windows (addresses) labeled with information gathered about physical objects in the regions of space indicated by the specifications. Based on $\pi(gn)$, the control mechanism predicts the desired properties of the next granular network in realized computation. This prediction is illustrated in the figure by the operation $Next$, which assigns the desired property to $\pi(gn)$. The subsequent granular network in the computation, generated by c-granule control, is expected to either satisfy this property or be 'close enough' to a network that does.

By drawing an analogy with infomorphisms in information flow (see [**?**]), we can define the satisfiability relation $\models_{Next}$ as follows:

$$x \models_{Next} \pi(gn) \text{ iff } Next(\pi(gn)) = x,$$

where x denotes the property assigned to $\pi(gn)$ by $Next$. This expresses the intended semantics of the transition relation over granular networks realized by c-granule control.

The c-granule control must possess the capabilities necessary to generate the next granular network following gn in the physical world. Based on $\pi(gn)$, the c-granule control generates a specification tr for the association (transformation) to be realized in the physical world using the operation denoted in the figure by $spec$.

This specification tr is then transformed by the implementational module (IM) of c-granule control, using the association *Assoc*, into a configuration c of the physical objects. One should note that we use the term association to emphasize that *Assoc* establishes correspondence between abstract and physical objects.

By drawing an analogy with infomorphisms in information flow (see [**?**]), we can define the satisfiability association $|\approx_{Assoc}$ as follows:

$$c \mid\approx_{Assoc} tr \text{ iff } Assoc(tr) = c,$$

where c is the configuration of physical objects assigned by the association tr.

After initializing interactions in c, the behavior of c is subsequently perceived by c-granule control, and the results of this perception are stored in the informational layer, represented in the figure as $percep(c)$. The information $percep(c)$ describes the real next granular network following *gn*. The c-granule control uses a relevant distance measure ($dist$ in the figure) to check whether $percep(c)$ is 'close enough' to a granular network possessing the property x. If this condition is not met, the c-granule control may choose to modify the *spec* operation to obtain a granular network of the desired quality, or it may adjust the target goal of the c-granule if modification is not feasible.

It is important to note that $percep$ also functions as a type of association operation on physical objects, enabling the decoding of information related to the perceived physical objects and their interactions. This decoding is achieved by accessing properties of physical objects that are 'directly' available to the c-granule control or by reasoning (judgment) based on previously perceived information, data stored in domain knowledge databases, or established physical laws. For example, in [**?**] neural networks can be trained from additional information obtained by enforcing the physical laws.

Our computing IGrC model is influenced by interactions with the physical world, in contrast to the classical computing model that is confined to the abstract space [**?**, **?**]. As a result, the presented IGrC model is not purely mathematical; it requires engagement with both abstract and physical objects. Furthermore, one should be aware that, *e.g.*, the results of the association $percep$ may depend on the interactions of the configuration of physical objects c with the environment.

The control consists several other important modules. More detailed description of control is included in the cited paper on IGrC (see, *e.g.*, [**?**, **?**, **?**, **?**, **?**, **?**, **?**, **?**, **?**]). The RM plays one of the most important role of the control of IS. In the simplest case, the rules are embedded in this module by designers. However, in many cases these rules should be learned and changed according to perceived changes. The central role in the control of IS play reasoning techniques supporting the IS control in its behavior.

It should be mentioned that in description of the c-granule control we restrict our considerations to specification explaining the intended behavior of the control. The realization of the control in the physical world aiming to satisfy a given specification is not discussed. This may be the task of human designer or other c-granule representing *e.g.*, robot.

12.3.3 Summary of comments pertained to realization of associations in IGrC

IGrC goes beyond abstract concepts like information granules in GrC. It also handles granules that interact with the physical world. The control of IS is equipped with the module IM responsible for realization so called physical semantics. The IM module takes specifications of associations (a broader term than specification of mathematical functions) and generates or uses existing configurations of physical objects. It then initializes interactions within these configurations and allows the IS control to perceive properties related to the object interactions. Based on this perceived information, along with knowledge bases and physical laws, the IS control can infer properties of the perceived objects as results of the realized association. It's important to note that these inferred properties might differ from expectations (expressed in specifications) due to environmental interactions (see Fig. 12.9). If the differences between the expected and perceived results of realization of transformations are too large than CM is looking for adaptation of rules stored in RM [2].

ASSOCIATIONS
AND THEIR PHYSICAL SEMANTICS

$f : X \rightsquigarrow_g Y$ where g is a given c-granule

- *X − defined in set theory, elements of X are stored (represented) in informational layer of c-granule g (e.g. control of IcS),*
- *Y − physical space, not definable in set theory,*
- *f − association between X and Y realized by c-granule g using **physical semantics**:*
  - ***implementation**: for a given $x \in X$ and a specification of f control of g is constructing a physical structural object o_x (with dynamics controlled by g relative to its local time) providing a 'physical pointer' from a part of o_x in which x has been encoded to the associated (by f) to x a physical object in o_x (pointed out by a spatio-temporal window specification represented in the physical layer of g),*
  - ***perception**: some properties of parts of o_x and properties of interactions between them (and with the environment) are perceived by control of g (in particular by decoding from some parts of o_x into informational layer of g) and used in **reasoning by g** toward providing representation of information about the object associated to x by f.*

Fig. 12.9 Associations and their physical semantics.

[2] One should note that Aristotle already emphasized the necessity of using adaptation in the decision-making process [**?**].

Referring to our previous discussion on granular spaces and granular networks, it is important to note that associations can occur at different granular levels. These associations suggest that their physical realizations through IM will facilitate the perception of information represented by information granules at the relevant granular levels. For instance, the granular level corresponding to the behavior of an animal's heart can be represented by information granules derived from sensory measurements related to that heart behavior.

In the context of Granular Computing (GrC), information granules can be seen as a specific type of c-granule. This allows us to focus on the specifications of c-granules, which are represented by information granules. The difference between information granules considered in GrC and c-granules from IGrC can be illustrated as follows. Let us assume that information inf specifying an intended change of a given information system IS to IS' is related to updating IS by adding a new attribute to the set of attributes of IS, with values computed according to a given procedure *proc*. If this specification is admissible for IS by its type then the change of IS to IS' is realized by implementation in computer hardware of the procedure *proc* and taking the computed by the procedure value as the value of attribute for each considered object. In this case, assuming that the realization in computer hardware is correct, in particular, not disturbed by the environment one can restrict considerations about the realized transformation to the corresponding informational layer without referring to the physical world. Therefore, the computational building blocks needed for cognition include both information granules and c-granules. C-granules are generated by control of IS using reasoning techniques. This control aims to construct high-quality approximate solutions for problems that IS needs to solve. These computational building blocks can take various forms, including patterns, clusters, information systems, classifiers, and physical objects such as new sensors, robots, or chemical compounds.

The basic concepts related to c-granules and ic-granules are summarized in Table 12.1 and Table 12.2, and illustrated in Fig. 12.10 and Fig. 12.11. In general, the structure of the control may be much compound and contain many other modules (see Fig. 12.13).

12.4 Further comments on behavior of the IS control and reasoning supporting it

Our research explores new directions for applying the generalized rough set approach to IS. This work builds upon the IGrC model and leverages existing partial results from various fields (including multi-agent systems, perception and action, machine learning, natural language processing, federated learning, cognitive networks, smart cities, cyber-physical systems, complex adaptive systems etc.) by putting them into sync. We also emphasized the fact that IS are dealing with complex phenomena in the physical world what requires a new kind of modeling for solving problems with the design and analysis of IS. It was pointed out that dia-

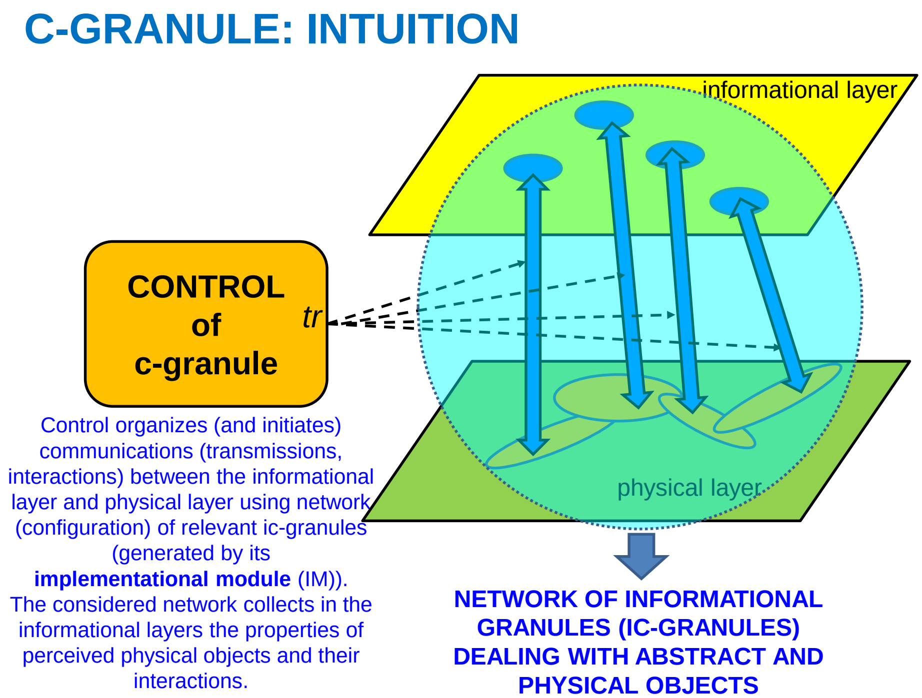


Fig. 12.10 A c-granule and a network of informational granules (ic-granules).

logues with domain experts are unavoidable for IS due to the fact that still we do not have satisfactory formal reasoning techniques making it possible to deal to a satisfactory degree with commonsense reasoning or experience based reasoning. The discussed comprehensive approach has the potential to establish a solid foundation for the design and analysis of IS.

Among reasoning methods in RM of control are methods based on deduction, induction or abduction [**?**, **?**]. However, there are some 'white spots' which require further research. Among them are reasoning (judgment) methods based on experience (*e.g.*, reasoning by analogy) [**?**, **?**] or mechanisms supporting discovery [**?**, **?**]. This causes that in some applications, especially dealing with complex phenomena (*e.g.*, medical) it is not possible to eliminate dialogues with domain experts [**?**, **?**, **?**] and/or chatbots [**?**].

One of the challenges for c-granules with control, in particular for IS's based on IGrC is to develop methods for reasoning over interactive computations performed on granular networks supporting c-granules in controlling such computations toward achieving the target tasks. Such reasoning techniques we sometimes call *adaptive judgment*. *Intuitive judgment* and *rational judgment* are distinguished as different kinds of judgment [**?**]. Adaptive judgment is the basic tool in discovering of relevant patterns of different complexity used for approximation of complex vague concepts

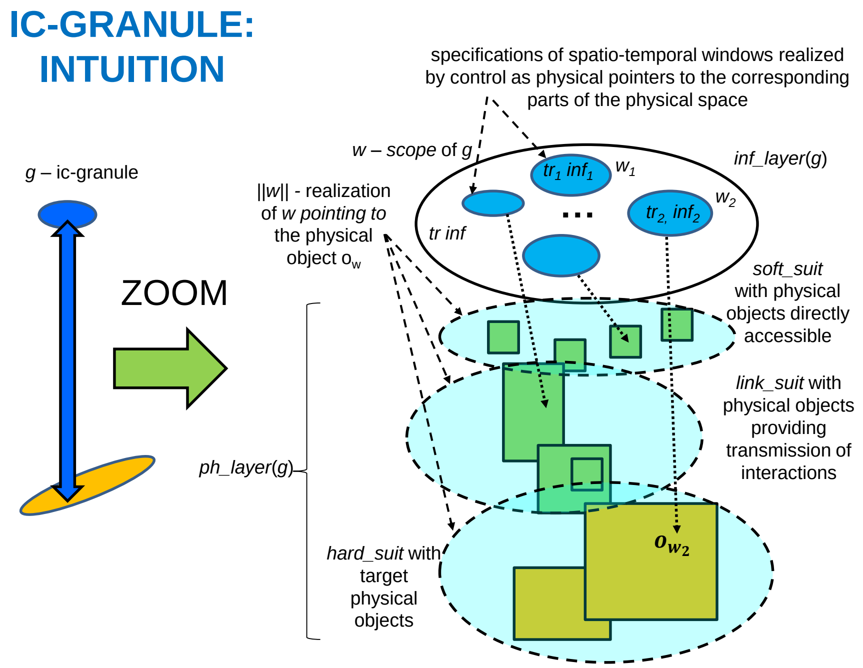


Fig. 12.11 Details of one of ic-granules that compose the network of ic-granules illustrated in Fig. 12.10.

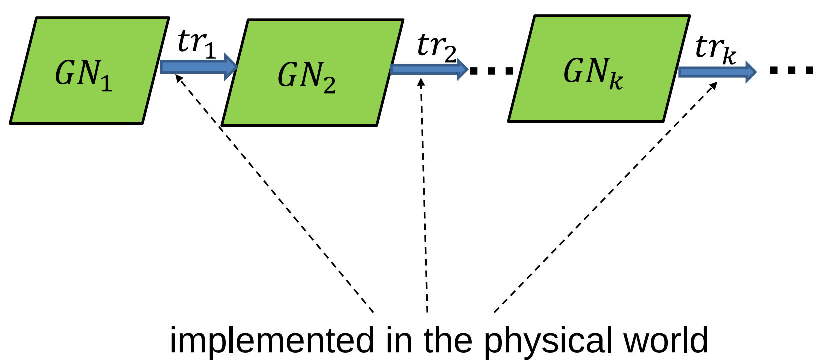


Fig. 12.12 Computation over granular networks.

and inducing approximate reasoning schemes on such approximations, called after Leslie Valiant as computational building blocks [**?**]. Adaptive judgement in IS's is a mixture of reasoning based on deduction, abduction, induction, case based or

Table 12.1 Notation used in this chapter

Name	Interpretation
g_c	complex granule (c-granule) (sometimes called informational granule (ic-granule) when it is enhanced by non-empty informational layer) is a dynamic object characterized (at a given moment of local time of g_c) by the control and a network of subordinate ic-granules (Fig. 12.10); the control is aiming to achieve the goals of g_c transforming the current network *GN* of ic granules into a new one by selecting the specification of network transformation *tr* (relevant to the current network)
GN	network of ic-granules composed out of a finite number of ic-granules
tr	specification of network transformation realized in the physical space by the implementational module (IM) of the control (possibly preceded by decomposition of specification to the form directly realizable by IM in the physical space)
control	contains several modules such as reasoning module (RM), implementational module (IM), attention module (AM), adaptation module (AdM), dialogue module (DM) and transition relation *rel*, goals and specification of family of networks *Fam_net*
rel	if $GN\ rel\ GN'$ then GN' is the real result of realization (in the physical world) of the selected transformation by the control at *GN*
comp	(finite) computation over *Fam_net*: GN_1 rel $GN_2 \ldots$ rel GN_k, where $GN_i \in Fam\_net$ for $i = 1, \ldots, k$ (see Fig. 12.12)
$trace(comp)$	information trace of *comp*: $inf\_l(GN_1), \ldots, inf\_l(GN_k)$, where $inf\_l(GN_i)$ is the information layer of GN_i for $i = 1, \ldots, k$
goal	goal of g_c interpreted by the control as a quality (utility) function over computations with values in $[0, 1]$
w	specification of a spatio-temporal window (in a given language)
$\|w\|$	subset of $\Re^3$, where $\Re$ is the set of reals, defined by *w*
o_w	physical object, *i.e.*, part of the physical space corresponding to $\|w\|$
g	informational granule (ic-granule) composed out of informational layer $inf\_layer(g)$ and physical layer $ph\_layer(g)$

analogy based reasoning, experience, observed changes in the environment, meta-heuristics from natural computing is used (see Fig. 12.14).

It is worthwhile to cite here the opinion about practical judgment from [**?**]:

> *Practical judgment is not algebraic calculation. Prior to any deductive or inductive reckoning, the judge is involved in selecting objects and relationships for attention and assessing their interactions. Identifying things of importance from a potentially endless pool of candidates, assessing their relative significance, and evaluating their relationships is well beyond the jurisdiction of reason.*

This opinion is strongly related to IGrC, in particular, to the role of IM and AM modules in realization in the physical world of associations.

We have already discussed some issues related to reasoning using IS control. Fig. 12.15 illustrates a variety of components of the reasoning module (RM) of IS from the point of view of different control tasks.

Table 12.2 Notation used in this chapter (cd.)

Name	Interpretation
$inf\_layer(g)$	informational layer of g consists of family of tuples (w, tr, inf), where inf is the current information updated by information perceived on o_w by g during realization by IM of the transformation specification tr
$scope(g)$	a distinguished w from $inf\_layer(g)$ where $\|w\|$ is the largest among all $\|w'\|$ from $inf\_layer(g)$
$\alpha \Longrightarrow tr : \beta$	rule, where α is a condition triggering rule defined over (relevant part of) $inf\_layer(g)$, tr is a specification of network transformation and β in the expected property of the resulting network after tr implementation
$Rule\_set$	set of rules (complex game) referring to the physical world
$ph\_layer(g)$	physical layer of g consists of parts: $soft\_suit$, $link\_suit$, $hard\_suit$ creating a dynamical physical system perceived by ic-granule g; perceived information is recorded in $inf\_layer(g)$
$soft\_suit$	consists of o_w, where w is from $inf\_layer(g)$ and o_w is directly accessible for measurement by g *e.g.*, IM can directly realize in the physical space the specifications $enc(inf, w), dec(w)$ of encoding inf in o_w and decoding information from the object after realization of $enc(inf, w)$ such that the realization of enc and next dec (*i.e.*, $dec(enc(inf, w))$) by IM results in inf and after that $enc(dec(o_w), w)$ results in o_w provided that the environment is not disturbing these implementation processes
$hard\_suit$	consists the target objects o_w not necessarily directly accessible; information about such objects recorded in information layer is obtained by the control on the basis of reasoning about objects in $scope$ using the current information in this layer including domain knowledge or physical laws
$link\_suit$	consists objects used for transmission of interactions between $soft\_suit$ and $hard\_suit$, information about such objects recorded in information layer in case they are not directly accessible is obtained by the control on the basis of reasoning about objects from $scope$ using the current information from informational layer including domain knowledge or physical laws

These components are crucial for the IS control in supporting generation of computations over granular networks by IS's aiming to solve specified problems such as

- *Learning algorithms and Classifiers Construction* [**?**, **?**]: The IS takes training data, quality requirements for the classifier, and databases containing information relevant to classifier construction (*e.g.*, search strategies) as input. The IS then returns a classifier that meets the specified quality requirements. This can be seen as searching for a complex structure (granule) that satisfies a given specification to a satisfactory degree, as measured by the quality of the constructed classifier. In other words, the returned classifier should be an example of com-

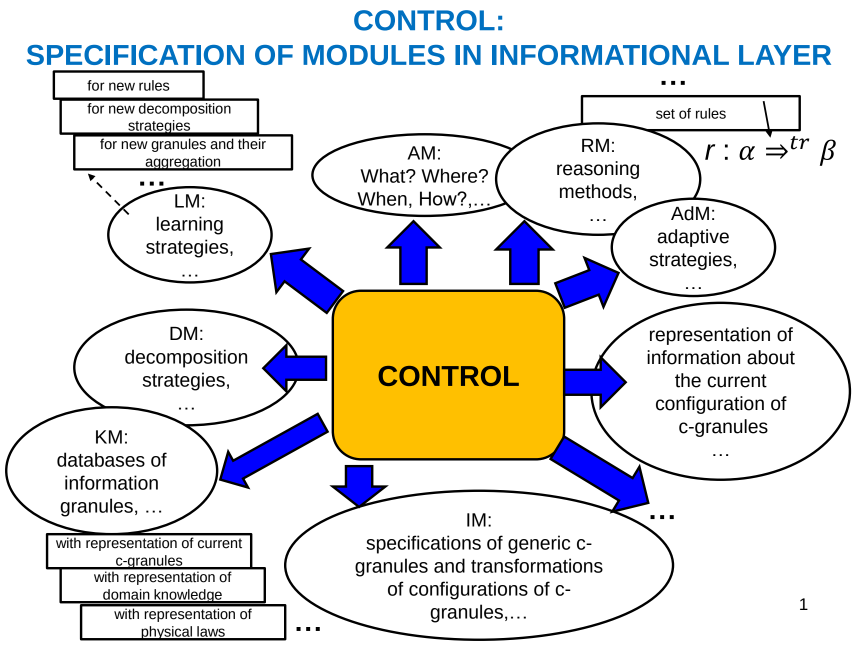


Fig. 12.13 The control of c-granule.

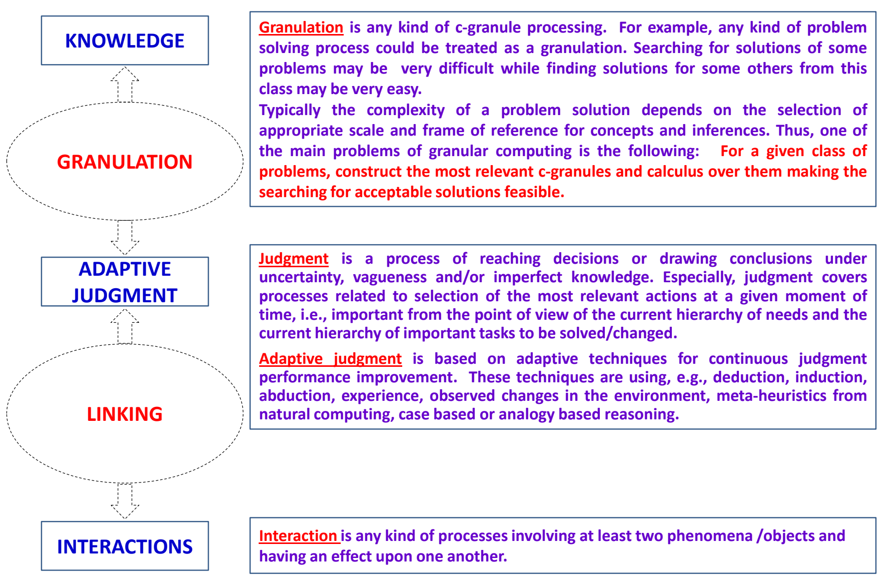


Fig. 12.14 Interactions, adaptive judgement and granulation

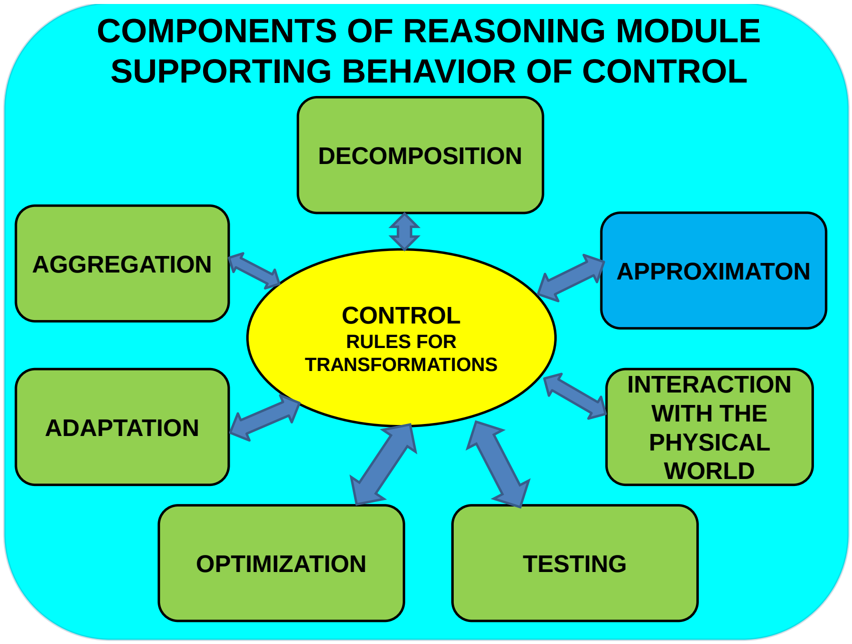


Fig. 12.15 Different components of RM in IS (c-granule with control).

plex granule belonging (with the high degree) to the lower approximation of the concept 'classifiers satisfying the given specification to a satisfactory degree'.

- *Automatic Design of Robots* [**?**]: The IS takes databases of robot parts and a specification for the robot (*e.g.*, concerning desired tasks and performance metrics) as input. It then returns a physical robot that meets the given specification to a satisfactory degree. Similarly to the first example, this represents finding a complex structure (c-granule) that fulfills the desired criteria. In this case, the returned robot as a physical object should be an example of c-granule belonging (with the high degree) to the lower approximation of the concept 'robots satisfying the given specification to a satisfactory degree'.
- *Drug Discovery* [**?**]: The IS takes various relevant knowledge bases (e.g., medical and chemical data) and a specification for the desired medicine (*e.g.*, expected effects) as input. It then returns a physical medicine that meets the specified requirements to a satisfactory degree. Once again, this translates to searching for a complex structure (granule) that adheres to the defined criteria. Here, the returned medicine as a physical object should belong (with the high degree) to the lower approximation of the concept 'medicine satisfying the given specification to a satisfactory degree'.

One can see that in each of the above mentioned applications IS should construct objects, called here c-granules, satisfying to a satisfactory degree a given specification. This can be expressed by requirement that the generated granules should belong, *e.g.*, to the lower approximation of the concept consisting granules representing approximate solutions or granular computations along which these solutions are constructed. Hence, IS should be equipped with advanced reasoning methods for performing approximate reasoning processes along which such complex granules may be constructed during generation of granular computations. In developing models of such approximate reasoning processes, one should answer several important questions. Among them are the following ones:

- How the target complex granules are specified?
- How the target complex granules are processed by IS?
- What are the objects (c-granules) over which approximate reasoning processes are running?
- How approximations of c-granules are defined and constructed along performed reasoning by IS? How calculi of granules on different hierarchical levels are defined and/or discovered? On what kind of reasoning are based strategies for discovery of relevant calculi of granules? What are the basic steps of reasoning supporting construction of the high quality solutions of the given problem from a given c-granule g representing specification?
- What are the methods of reasoning over generated by IS granular computations aiming to control them toward solutions satisfying given specifications to a satisfactory degree? Which methods can provide insightful reasoning for control of IS?
- What kind of quality measures should be used for estimation of the quality of granules in construction of solutions (*e.g.*, classifiers, clusters, plans, and other complex abstract and physical objects) evaluated along the approximate reasoning processes?
- What kind of computing model should be used by IS to guarantee feasibility of construction of the high quality of approximate solutions realized in the physical world?
- What kind of testing methods IS systems should use to verify the quality of provided solutions?

Answering these question requires from the IS control reasoning techniques pertained to different components of RM. In general, reasoning techniques from RM module are supporting decision making about the currently perceived situation and are performed along granular computations over granular networks. Information from c-granules used in reasoning may be obtained as a consequence of interactions with the physical objects (*e.g.*, by measurement or performing actions being consequences of realization of associations in the physical world) as well as aggregation of information granules also with the use of domain knowledge databases or physical laws represented in informational layers of information granules.

This discussed approach focuses on IS that handle complex phenomena. When solving problems with such IS, two key issues arise:

1. the appropriate computing model,
2. the approximate reasoning processes that guide computations within this model.

These problems involve IS generating computations that lead to solutions, *e.g.*, within the lower approximation of concept encompassing all solutions to a given problem. This formally expresses that the generated solution is of high quality or is sufficiently satisfactory.

In particular, we already emphasized the role of reasoning supporting optimization of different tasks performed by IS in searching for the high quality solutions of given problems to be solved, *e.g.*, optimization of parameters of approximation spaces, quality measures, learning algorithms. In this section, we discuss some others of these components. To summarize, let us remind that reasoning by control is performed on complex granules (c-granules) among which are granules linking abstract and physical objects. Control is performing reasoning on granular networks generated by control of IS and along performed reasoning are generated granular computations aiming to lead to the high quality of approximate solutions of considered problems.

The states of IS's are networks of granules from different networks of granular spaces. Granules from different granular spaces are linked by more compound relations than granules linked from the same granular space. Granular networks are storing properties of single granules and properties of computations over granular networks. The links between granules represent different relationships between granules. Some granules represent domain knowledge, some other are used to represent components of IS control. There is also a distinguished granule representing the state of perception of the currently perceived situation.

Control of IS is deciding in each state what kind of transformation of the current granular network to perform. The transformation may concern suspending some measurements, elimination of some granules from the network, updating of the granular network with new granules or updating the existing granules. The control decides which transformation to select by matching rules being at its disposal against the granule representing information about the currently perceived situation. The set of rules can be treated as a complex game (see. *e.g.*, [**?**, **?**, **?**, **?**, **?**, **?**]) consisting of concepts (often, complex vague concepts) labeled by transformations which should de realized if the concept is satisfied (to a satisfactory degree) (see Fig. 12.16).

The complex game illustrated in Fig. 12.16 consists of pairs (C_j, tr_{j_i}) for $j = 1, \ldots, n$, where each C_j is a classifier for a vague concept triggering the transformation specification t_{j_i}. If C_j is satisfied (to a satisfactory degree) in the current perceived situation, the IS control registers this as a match. Subsequently, the control attempts to resolve conflicts among all matched classifiers to select a transformation specification for the IM to execute in the physical world. This is supported by appropriate reasoning methods. If successful, the selected specification is transmitted to the IM; otherwise, the IS control seeks additional information about the perceived situation (*e.g.*, through sensory measurements or knowledge derived from c-granules representing domain knowledge bases). The IM executes the transformation specification in the physical world through the following steps: (i) creating (or organizing) a configuration of physical objects defining the scope of the transformation, (ii) initiating interactions within this configuration, (iii) allowing the IS control to perceive relevant object properties and interactions over the relevant period of time and recording results in the informational layers of appropriate c-granules, and (iv) treating the appropriate recorded values as the transformation's outcomes. It's important to note that these outcomes may deviate from expectations due to environmental interactions.

Both the concept and labeling it transformation are complex granules (c-granules) in IGrC. The satisfiability of the concept to a satisfactory degree (under additional assumption that conflict resolution strategy between rules satisfiable to satisfactory degree points to the concept) initiates interaction with granule activating transformation leading to its realization.

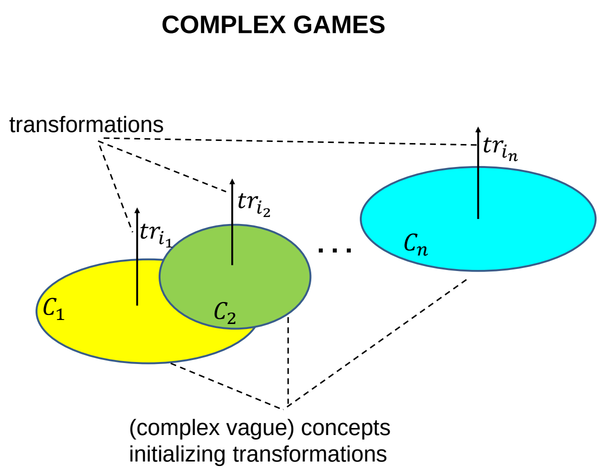


Fig. 12.16 Complex granule consisting of set of rules interpreted as a complex game with granules $C_1, \ldots C_n$ representing vague concepts and their classifiers labeled by transformations $tr_{i_1}, \ldots, tr_{i_n}$.

One should note that the rules set of the control of IS may have a compound structure, *e.g.*, representing distributed control or priorities of rules. Moreover, IS should be equipped with reasoning methods for development strategies for adaptation of rules based, *e.g.*, on optimization of various parameters of entities leading to construction of rules including granular spaces and networks over them, constructions of granular networks and forms of transformations. Fig. 12.17 presents a simplified example of an adaptive complex game. Classifiers for complex vague concepts, denoted as $C_1, \ldots, C_m$, trigger complex games $G_1, \ldots G_m$. Using these classifiers, the IS control identifies concepts $C_1, \ldots, C_m$ that match the current perceived situation and employs appropriate reasoning methods to resolve conflicts among them. If conflicts can be resolved, a complex game from $G_1, \ldots, G_m$ is selected; otherwise, the control seeks additional information about the perceived situation to facilitate conflict resolution. Challenging problems are related to learning the high quality classifiers for $C_1, \ldots, C_m$. One should also note that the *IC* control must determine whether the goals outlined in the problem specification can be achieved. If not, it should adjust the goals accordingly. Hence, it is becoming important reasoning supporting recognition on feasibility of realization of goals, otherwise their modification should be proposed (*e.g.*, with the use of Maslov hierarchy of needs [**?**] or its modification).

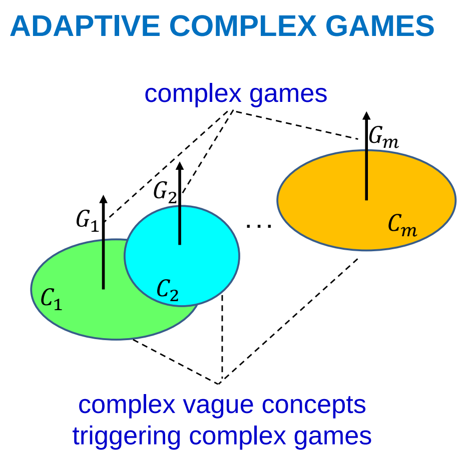


Fig. 12.17 Adaptive complex game consisting of vague concepts with classifiers $C_1,\ldots,C_m$ labeled by complex games $G_1,\ldots,G_m$.

Notably, complex games and adaptive complex games can be viewed as more intricate c-granules. Their behavior arises from interactions between granules representing vague concepts. These interacting granules are labeled by c-granules representing decisions (*e.g.*, plans or complex games relevant to the domains defined by these vague concepts). Additionally, reasoning processes influence these interactions by resolving conflicts between votes on different decisions predicted by the concepts. This reasoning ultimately establishes the interaction deciding which plan or complex game to launch.

More details about these important issues will be presented in our next works. Here, we would put attention of the reader to the importance and richness of the required reasoning methods supporting the control of IS.

Let us consider in more detail two more examples related to reasoning methods related to aggregation and decomposition of granular networks.

Aggregation of granules is widely used in Machine Learning, *e.g.*, in feature engineering or hierarchical learning [**?**, **?**, **?**] and Granular Computing (GrC). In feature engineering, new features are defined over the existing ones. Hence, a partition of the universe of objects defined by a new feature if defined by partitions of some known already features. This may be interpreted as a transformation of granules from one granular space into a new one.

In general, the decomposition problems are challenging. According to the opinion of Judea Pearl [**?**]:

> *Traditional statistics is strong in devising ways of describing data and inferring distributional parameters from sample. Causal inference requires two additional ingredients:*
>
> - *a science-friendly language for articulating causal knowledge, and*
> - *a mathematical machinery for processing that knowledge, combining it with data and drawing new causal conclusions about a phenomenon.*

This transition from 'a science-friendly language for articulating causal knowledge' to 'a mathematical machinery for processing that knowledge' usually requires decomposition through several layers.

Decomposition (degranulation [**?**]) of granules is an important issue in GrC and IGrC. Here, we discuss shortly some important problems related to decomposition of granules, in particular described in natural language. Language of granules consists complex vague concepts expressed in natural language. Their semantics is defined by classifiers induced from data. We emphasize reasoning methods leading to the relevant decomposition. Let us mention two illustrative examples: decomposition of specification of tasks of IS's and decomposition of specifications of transformations in realization of associations.

In [**?**] we have presented examples of such tasks. In each of these examples, the description of task is presented using complex vague concepts what can be hardly understandable by IS without additional support. It is worthwhile to cite here an opinion of Lotfi Zadeh [**?**, **?**]:

> *Information granulation plays a key role in implementation of the strategy of divide-and-conquer in human problem-solving.*

One can ask how to acquire or learn such a strategy. One possible approach is to provide dialogue methods with domain experts. This is especially important for medical applications [**?**, **?**]. Another one is related to applications of chatboots [**?**] and Large Language Models (LLM) [**?**, **?**]. It may be necessary to go through several levels of decomposition, with possible backtracking, before the level with concepts which can be efficiently and with high quality approximated can be reached on the basis of available data. In the case of realization of specifications of associations for this last level a direct realization of this level in the physical world should be possible. This approach proved to be successful for several real-life projects where so called the rough set-based ontology approximation of concepts was applied: after decomposition to the lowest level the obtained ontology of concepts was approximated using bottom-up strategy (see, *e.g.*, [**?**]).

IS also require reasoning mechanisms to handle situations where the current search strategy for relevant granules is unlikely to succeed. In such cases, the reasoning system of IS should guide the backtracking process by appropriately decomposing (degranulating) some already generated granules up to a suitable level. Additionally, more advanced IS should leverage information from unsuccessful trials to learn and adapt their search paths towards discovering the relevant granules.

12.5 Rough sets in IgrC paradigm

12.5.1 Information systems in IGrC

In this section, we present some comments on management of information systems in c-granules by their CM module. Information systems in IGrC, contrary to the existing approaches, are in 'hands' of CM (see Fig. 12.18), *i.e.*, their dynamics depends on CM and environmental interactions.

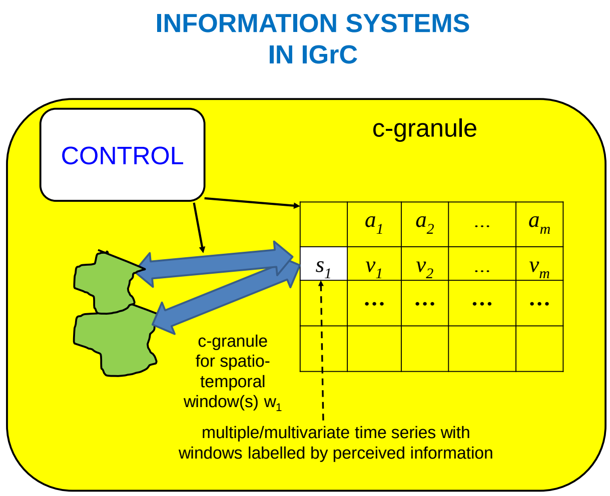


Fig. 12.18 Information system in 'hands' of control of c-granule.

The c-granules (with control) hold specific pieces of information stored in their informational layers. Among these pieces are information systems. In our discussion, we consider IS as a c-granule with control. IS typically manages multiple information systems. Therefore, its control requires a proper addressing mechanism (realized by spatio-temporal windows or addresses) to locate the relevant information system, considering both space and time.

Each information system, identified by its address, stores objects of a predefined type. The type of these objects is defined by a formula that allows the IS control to determine if a piece of information sent by the IS control can modify the system and how. For instance, the type might specify that the system holds objects defined by a spatio-temporal window describing where specific attributes should be measured. It could also include time information, such as the start time for the measurement and the expected duration.

Information systems can store more complex object types. These could include, *e.g.*, properties of segments from different multi-time series that the IS control perceives during measurements. These segments could be aggregated into clusters or even more intricate structures. Additionally, types can hold properties related to interactions with physical and abstract objects. These properties might also include conditions expressing relationships between attributes, such as specifying that certain parts of the observed objects are physically close.

Formally, these types can be represented as formulas $\alpha(x)$ in a specific language. When checking if information inf is relevant for updating a particular information system *IS*, the variable x in the formula is replaced with the information inf. This information describes how the IS control intends to modify *IS* by adding a new object to its collection.

Let us consider an illustrative example in which this new object o is defined by information inf in the format:

$$w : a_1, \ldots, a_m; spec.$$

Here, w is a spatio-temporal window specification, identifying a part of physical space where the values of attributes a_1 to a_m should be measured. The expression *spec* refers to a specification for how the module IM of control should obtain these values.

If this object satisfies the type formula $\alpha(x)$, the IS control follows these steps:

- It expands the universe of objects within *IS* by adding the new object o.
- The IS control sends a request to the IM to realize the specification *spec* in the physical world.
- IM initiates a process in which it perceives the values of attributes a_1 to a_m in the specified part of the physical space (defined by w).
- These values are then stored in the expanded information system IS' as attribute values for the newly created object o.

Before summarizing our considerations in this section, let us make some complementary comments.

Let us consider an illustrative example concerning types of aggregated information systems. We consider an aggregation of two information systems IS_1, IS_2 of type $\alpha_1(x), \alpha_2(y)$, respectively. In aggregation we take the Cartesian product of the universes of objects of IS_1, IS_2 and filter it with constraint $\alpha(x,y)$ defined with the use of some relations between value sets of attributes from IS_1, IS_2 (*e.g.*, specifying closeness of objects x and y). The type of this exemplary aggregation is defined by $\alpha(x,y) \wedge \alpha_1(x) \wedge \alpha_2(y)$.

One can observe that the specification of type of information system *IS* is closely related to specification of a family of admissible changes of *IS*..

Summarizing, we propose the following changes in modeling of information systems in comparison to the Pawlak model:

- *The Key to Dynamics: Open Information Systems.* We propose a generalization of Pawlak's information systems into 'open' information systems. These systems are dynamic entities, and the IS control is responsible for their evolution during computations seeking high-quality approximate solutions for problems the IS needs to solve. Hence, the dynamics of these systems is not defined a priori as it was proposed in papers on dynamic information systems so far (see, *e.g.*, [**?**, **?**, **?**]). Pawlak's information systems can be seen as starting points, or 'seeds.' We need to consider huge spaces of information systems around them. Within these spaces, it is necessary to search for (semi-)optimal information systems (or approximation spaces). The relevant reasoning techniques should be developed supporting this search making it possible to induce the relevant computational building blocks for cognition, like classifiers or clusters. Furthermore, the IS control system must be aware that these vast information spaces are dynamic and change over time.

- *Challenges and Networks.* The intended dynamics may not always be achieved due to unforeseen interactions with the physical environment. Furthermore, the IS control often deals with multiple interconnected information systems rather than a single one, forming networks of information systems. These networks can be viewed as networks of c-granules over which computations are generated by control of IS.
- *System Types and Objects in Information Systems.* Each information system has a type that specifies the allowed type of objects in it, in particular those that can be added in updating. The type is typically specified using some properties of fragments of granular computations generated by the IS. Objects within an information system must be compatible with the system's type. It can be treated as a filter for objects to be stored in information system. Objects in information systems are descriptions of structural objects labeled by specifications of spatio-temporal windows, like bitmaps of images, fragments of time series or their clusters, together with encoded in these descriptions procedures and/or specifications of associations used by control of IS for computing values of the relevant attributes. The control of IS is using

  (i) a procedure encoded in the object to compute necessary attribute values for an object using, *e.g.*, information from other information systems and/or
  (ii) a specification of an association encoded in the object with the specified fragment of the physical space; a process is associated with this specification that is realized in the physical world by IM, allowing IS to perceive, *e.g.*, the desired attribute values and store them in the information system.

  Hence, attributes considered in the paper are not necessarily abstract functions by they may be defined by specifications of associations and their realization in the physical world by IM.
  The type specification can also include information on how the IS control can update the information system, *e.g.*, by adding or removing rows or columns. For example, a type might specify a type of spatio-temporal windows, a list of attributes to measure within the fragments of the physical space corresponding to those windows, additional information like measurement time frames and expected results. Control of IS is responsible for updating the system with the perceived in physical realization measurement values.
- *Dynamic by Design, Not Random.* These generalized information systems are dynamic, not randomly so. Their evolution is determined by the IS control's

intended dynamics, which can be influenced by interactions with the physical environment. This implies that the information systems in this paper are not purely mathematical objects. Their dynamics are defined by c-granules representing the systems themselves and their interactions with other c-granules, working like physical pointers to specific parts of the physical space. These pointers allow the IS control to perceive properties of physical objects in those parts and use this information to update the current state of the systems.

This type of modeling is essential for seriously considering issues related to perceiving complex situations in the physical world and making the right decisions about them. We demonstrated that our approach to information systems can be seen as a constructive way to define dynamic approximation spaces. These spaces are changing accordingly to changes of corresponding to them information systems and they can be used to search for computational building blocks for cognition, such as patterns, clusters, or concept approximations (classifiers) a well as their hierarchical structures relevant to problem-solving by the IS.

Typically, control of IS deals with a family of information systems generated in the perception of situations. Moreover, the information systems from such families are linked by different relations representing relationships between objects, fragments of the whole information systems. In this way are created networks of information systems. They are examples of more compound c-granules (or information granules), corresponding according to our previous discussion, to families of approximation spaces. Here, it is worthwhile mentioning the relationships with information flow [**?**, **?**] attempting to develop logical foundations for distributed computing. One should also refer here Fuzzy(-Rough) Cognitive Networks (see, *e.g.*, [**?**, **?**, **?**, **?**]) as well as Federated Learning (see, *e.g.*, [**?**, **?**]) as examples of techniques aiming to create machine learning models with improved performance on distributed datasets (without sacrificing privacy). On the way to create such models many challenges appear concerning, *e.g.*, creating, designing, operationalizing or maintaining distributed systems. One of the challenges is related to developing reasoning methods supporting solving these challenges.

The proposed approach focuses on developing reasoning methods that support the construction of approximate solutions for specified tasks. These methods must consider additional constraints during construction. These constraints can include privacy requirements, limitations on data aggregation due to resource limitations, or adherence to principles expressed in natural language standards (*e.g.*, ISO standards) that may contain complex and vague concepts. Importantly, these reasoning methods should not only analyze pre-constructed solutions but also actively support the construction process itself, working alongside the granular computations generated by IS. Dialogues with human experts may play a crucial role in this process. Furthermore, at different stages of the IS computations, solved subproblems can be treated as optimization problems within large families of approximation spaces.

Aggregation and decomposition operations as well as filtration (see, *e.g.*, [**?**, **?**]) of information systems enable us to construct new information systems on the basis of which new relevant c-granules being computational building blocks for comprehension of the perceived situations are discovered.

One of the fundamental issue of information systems under the control of IS is that they are open to interaction. They are changed by control of IS. The changes are controlled by reasoning skills of control. We discuss this issue in more detail in the subsequent section.

12.5.2 Approximate solutions generated by IS's based on the rough set approach grounded in IGrC

In this section, we discuss and summarize shortly perspectives of rough sets in the framework of IS's and IGrC.

Fig. 12.19 illustrates a paradigm shift in approximation from the Pawlak model to the model presented in this paper. Approximation in the Pawlak model involves concepts represented as subsets of a given finite universe of objects. These concepts are approximated using so-called definable sets, which are unions of indiscernibility classes derived from an equivalence relation within a specified approximation space. A fundamental goal in IS is to identify high-quality approximate solutions for problems to be addressed by IS. These solutions are constructed along granular computations guided by IS control. This process is facilitated by advanced reasoning techniques and/or expert collaboration. The IS control behavior is based on adaptive models of so-called complex games (see Section 12.3.2), composed of rules that dictate necessary transformations within the current state of granular computation when relevant conditions are satisfied (to a satisfactory degree). The quality of approximate solutions is evaluated using appropriate quality measures. A primary objective is to generate, within the space of approximate solutions for a given problem, examples of approximate solutions constructed along granular computations definitively (with certainty) belonging to the concept of high-quality solutions, *i.e.*, the lower approximation of this concept. It is important to note that such examples are often unavailable beforehand.

In Fig. 12.20 is presented a context in which rough sets should be considered in IS.

In the context of Fig. 12.20, the rough set approach departs from traditional methods that rely on a single information or decision system (data table). Instead, it utilizes perceived data sets with AM. These data sets are used to create multiple multi-relational approximation spaces, denoted as $AS_1, \ldots, AS_k$, each corresponding to a distinct information or decision system. Various techniques are then applied to these spaces to generate a family of multi-relational approximation spaces, denoted as $\mathscr{F}_{AS_1,\ldots,AS_k}$. This family serves as the basis for optimization algorithms searching for high-quality complex games.

In Fig. 12.21 is presented a more general scheme illustrating the role of adaptive rough sets in the c-granule control.

In the upper part of the figure, a learning scheme for a complex game is presented. It begins with generic (basic) granular spaces $GS_1(\Theta_1), \ldots, GS_k(\Theta_k)$, characterized by parameters $\Theta_1, \ldots, \Theta_k$. These spaces consist of c-granules that correspond to

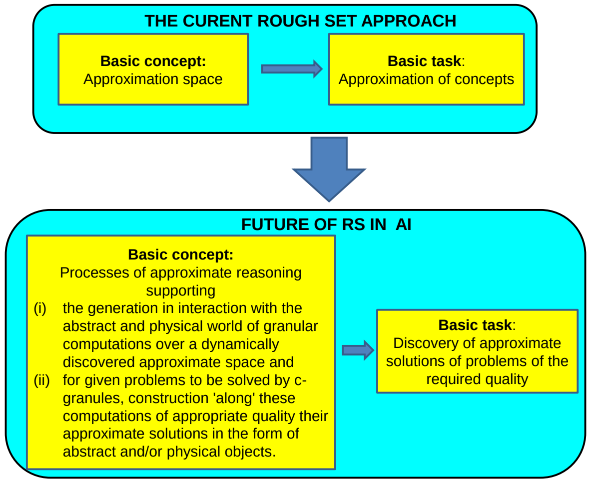


Fig. 12.19 From approximation of concepts to approximate solutions of problems.

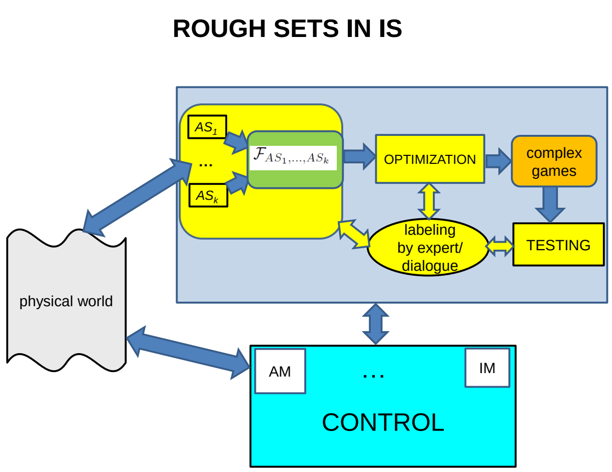


Fig. 12.20 Rough sets in IS (AM-attention module, IM-implementational module).

sensory measurements. The informational layers of these c-granules store data represented by information granules (*e.g.*, information systems or decision systems) perceived in the physical space through sensory measurements. The relationships within these granular spaces are characterized by (partial) inclusion among the information granules. The parameters $\Theta_1,\ldots,\Theta_k$ correspond to factors such as the location of the measurement, the time at which it was taken, the reasoning behind the measurement, and the methods used. Next, a network of granular spaces, denoted as $NGS_{GS_1(\Theta_1),\ldots,GS_k(\Theta_k)}$, is constructed from these granular spaces. This network con-

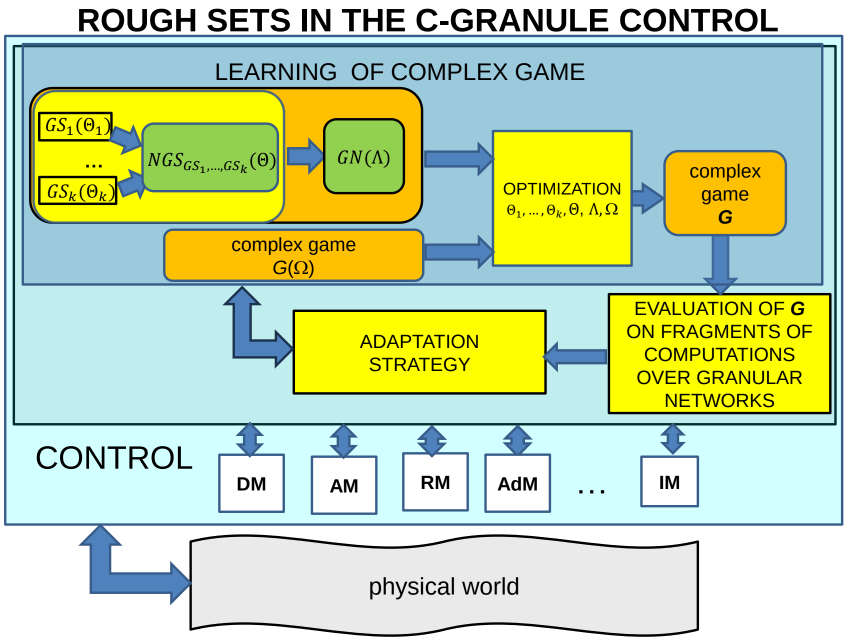


Fig. 12.21 Adaptive rough sets in the c-granule control.

sists of c-granules of various types (*e.g.*, related to different levels of hierarchical modeling), along with interfaces that specify how higher-level c-granules are constructed from simpler ones. From this network, a granular network $GN(\Lambda)$ is formed by selecting relevant c-granules and the relationships between them. The parameters in Λ specify possible selections of these c-granules and their relationships within the network.

Following this, the optimization of the complex game focuses on selecting parameters from $\Theta_1,\ldots,\Theta_k,\Theta,\Lambda$. The constructed complex game is evaluated based on relevant fragments of computations generated from the granular network. Various quality measures, dependent on the target goals of the specific c-granule (*e.g.*, information system), are employed in the evaluation. If the quality is unsatisfactory, the complex game is modified using an adaptive strategy. The adaptive module (AdM) is responsible for managing the adaptation in the control of c-granules.

The figure also illustrates the IM module, previously explained, which is responsible for implementing the specifications of transformations from the complex game in the physical world. Additionally, it shows the reasoning module (RM), attention module (AM) and the dialogue module (DM), which facilitates interaction between the c-granule and experts to support its behavior.

The c-granule with control aims to generate computations over granular networks that meet specific target goals. As previously mentioned, the quality of these computations may depend on either the final state of the computation or the entire computation process, depending on the task specifications realized by the c-granule. Degrees of computation quality correspond to different regions of approximation (*e.g.*, lower approximation region, upper approximation region, or boundary region) of potential solutions for the specified task realized by the c-granule. It is important to note that in some tasks executed by c-granules, positive examples of objects from certain approximation regions may not be available, unlike the scenarios typically encountered in machine learning (ML). In other words, the c-granule must discover examples of such objects by constructing them along granular computations.

In the considered context, the goal is to generate examples of computations that belong to the lower approximation of the concept being considered. These examples, referred to as positive examples, are not provided a priori. The objective is to discover complex games for controlling c-granules that enable the generation of such computations representing positive examples. Often, it is necessary to adapt the initial complex game due to changes in the physical world with which the c-granules interact as well as due to recognized mistakes performed in searching. Constructing positive examples requires conducting experiments that involve interactions with the physical world.

Approximate solutions are constructed along computations over granular networks. In the space of generated computations by control one can distinguish computations of the high quality, *i.e.*, computations generating approximate solutions of the high quality. Such computations belong to the lower approximation of the concept consisting computations of the high quality. Other computations may belong to computations of acceptable quality (creating upper approximation of the concept consisting of the high quality computations) or to complement of such concept (creating complement of the upper approximation) (see Fig. 12.22).

It's important to note that the optimization process involves aggregation and deaggregation, which correspond to granulation and de-granulation of the underlying information and decision systems, respectively. This process is crucial for identifying relevant multi-relational approximation spaces. As Zadeh previously emphasized in his proposal on information granulation [**?**, **?**], in decomposition of vague concepts ongoing dialogue with experts remains essential.

Furthermore, unlike traditional approaches that approximate a single concept, our method aims to construct a family of concepts in the learning phase of complex games. These concepts are labeled by specifications of transformations, which represent the specifications of actions. During testing, the quality of the discovered complex games is evaluated using application-specific quality measures. Additionally, based on the testing results, the games are adapted. AM supported by relevant reasoning techniques and expert collaboration, empowers IS to acquire new data sets. This allows for more efficient adaptation.

In our discussion on GrC and IGrC, we highlighted the importance of developing relevant quality measures (metrics) for constructed granules. We have mentioned various proposals for metrics used in ML, particularly those related to the Minimum Description Length (MDL) principle and confusion matrices. For instance, we can consider quality measures for features treated as granules in machine learning, as

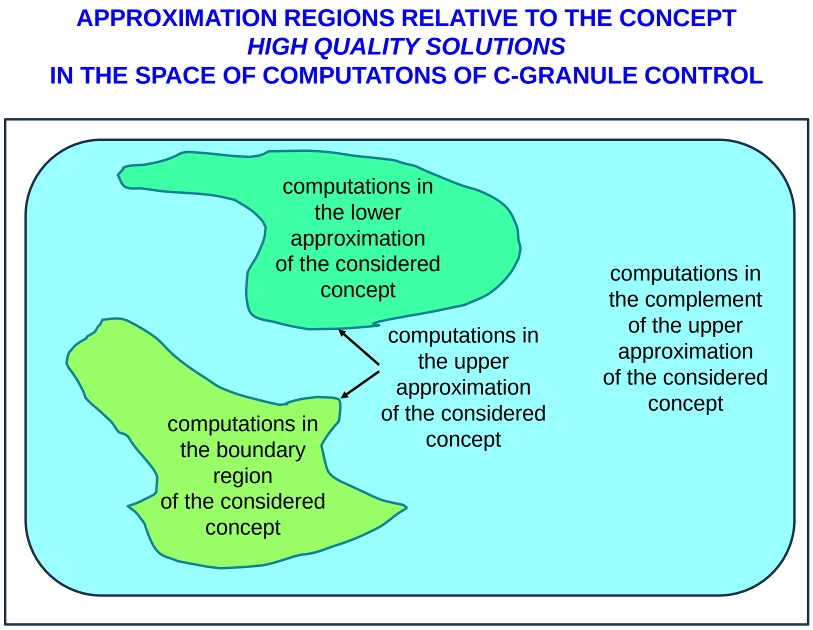


Fig. 12.22 Approximation regions in the space of computations of c-granule control.

well as metrics used in the optimization of parameterized granules, such as meta-parameters in learning algorithms.

In GrC, it is proposed to use the principle of justifiable granularity as a basis for developing quality measures for granules [**?**, **?**]. In IGrC, we suggest constructing these measures based on the relevance of granules as computational building blocks necessary for understanding perceived situations, enabling better decision-making. Consequently, quality measures should be considered in the context of c-granule control. A significant challenge lies in developing measures that support the estimation of the quality of c-granules within this control framework. This will allow for the evaluation of the behaviors of such granules, particularly regarding the quality of their control. Another example of challenges concerns developing measures for estimation of the quality of performed reasoning by c-granule control aiming to support control in making decisions concerning its behavior.

It can be noted that our discussion can also be applied to fuzzy sets [**?**, **?**]. In this case, the modeling of fuzzy membership functions also requires the proper computing model for interactions with the physical environment. The restriction of fuzzy sets to membership functions as transformation of a given set X into the interval of reals $[0,1]$ is not relevant for the discussed problems to be solved by IS. In particular, the problem of learning the set X in interaction with physical space as the relevant representation of perceived situations in the physical space is omitted. The detailed discussion will be presented elsewhere.

12.6 Conclusions

In this chapter, we discussed GrC and its extension to IGrC as the basis for the design and analysis of IS's dealing with complex phenomena. The following conclusions summarize our considerations:

- The presented approach is based on a substantial generalization of the existing rough set approaches, including generalization of approximation spaces to dynamic networks of granular spaces and generated computations over granular networks. These entities are used by control of c-granule to discover the relevant computational building blocks for cognition in the form of c-granules, granular networks and computations over them. By using discovered complex games the c-granule control is aiming to understand the perceived situations for making the right decisions in realization of the specified tasks along generated granular computations.
- The control mechanism of IS based on the rough set theory framework, aims to adaptively learn complex games. This allows IS to generate computations over granular networks leading to high-quality approximate solutions.
- The control system, aided by AM, continuously searches for relevant data sets represented in multi-relational approximation spaces as $AS_1, \ldots, AS_k$, or in more general granular spaces. The AM is leveraging advanced reasoning techniques in collaboration with domain experts.
- These approximation spaces can be extended, in cooperation with experts, into a large family of multi-relational approximation spaces $\mathscr{F}_{AS_1,\ldots,AS_k}$ or networks of granular spaces and extracted from them granular networks. The system then performs optimization to identify high-quality complex games.
- These complex games are adaptively modified based on observed changes in their performance. As previously discussed, complex games consist of sets of rules. The predecessors of these rules are classifiers for often complex, imprecise and vague concepts triggering specifications of transformations appearing on the right hand side of rules to be applied to granular networks when a rule is chosen for execution by the IS control. One should note that IM may require to make multi-level decomposition of the specification of transformation to be realized before it can be directly embedded into the physical world.
- The universes of objects of information (decision) systems constructed by control of IS are formed by expressions satisfying their types expressing properties of fragments of granular computations realized by IS.
- In many cases, IS needs to generate the granular computation leading to high-quality solutions even when no known examples of such solutions exist and only some negative examples are available. This is the case when, *e.g.*, IS is searching for new chemical compounds with some specified properties. The success of this generation process heavily relies on the reasoning techniques supporting the IS control. The quality of the generated solution, determined by the quality of the used complex game, often depends on the behavior of this

complex game over the entire generated granular computation, not its final state only.

In summary, this approach highlights that approximation problems in IS are significantly more complex than those typically encountered in rough set applications, so far. The success of the control system heavily depends on the quality of both reasoning techniques and the dialogue with domain experts.

Our considerations are opening a new directions for the new fascinating research on the generalized rough set approach in IcS. This research based on IGrC, using also many existing partial results from different areas like multi-agent systems, perception and action, machine learning, natural language processing etc., may lead to creation solid foundations for the IcS design and analysis. This research will be important in the realization of of a bit generalized goal formulated in [**?**]:

> *Tomorrow, I believe,* ~~every biologist will use~~ [we will use INTELLIGENT SYSTEMS] *to support our decisions in defining our research strategy and specific aims, in managing our experiments, in collecting our results, interpreting our data, in incorporating the findings of others, in disseminating our observations, in extending (generalizing) our experimental observations through exploratory discovery and modeling – in directions completely unanticipated.*

In the future, we will continue the research on the foundations of IGrC aiming to extend the current results to societies of c-granules with control as well as dialogues with experts as well as chatbots, LLM and Agentic Systems. Designing societies of c-granules with control providing the required behaviour of society in the environment of another given society of c-granules is a challenge (*e.g.*, discovery of learning algorithms and classifiers, new medicine or chemical compounds, control of autonomous vehicles or design of robots [**?**, **?**, **?**, **?**]).

We will continue our research on the foundations of IGrC, aiming to expand our current findings to include societies of c-granules. This will involve both further studies on control of c-granules and dialogues with experts, chatbots, and large language models (LLMs) as well as Agentic Systems.

One of the challenges is to develop the granulated deep learning approach on the basis of the IGrC model [**?**]. Moreover, also a combination of the rough set approach based on IGrC with the Lifelong Learning (LL) [**?**] is a challenge. LL calls for techniques of learning in the dynamic and open world or environment in a self-supervised manner what is consistent with the aims of IGrC. Many recent applications (*e.g.*, chatbots, self-driving cars, or AI systems interacting with humans/physical environments) require to cope with their dynamic and open environments. Hence, they should continuously learn new things in order to function well. Techniques supporting realization of this goal should be based on the relevant computing model.

The proposed IGrC model may also be treated as a step toward realization of combination of the physical structure and thinking behavior of the brain [**?**]:

> *We should combine the physical structure and thinking behavior of the brain, add physical priors, break through the bottleneck of computing power, realize low-power, low-parameter, high-speed, high-precision, non-depth AI models, and develop more efficient artificial intelligence technology.*

Problem 12.1. Please propose a generator of (parameterized) granular spaces from a given basic granular space.

Problem 12.2. Please propose examples of (parameterized) interfaces between granular spaces.

Problem 12.3. Please give examples of rules from interfaces between granular spaces.

Problem 12.4. Please give examples of reasoning based on composition of rules from interfaces between granular spaces.

Problem 12.5. Please design an example of (parameterized) network of granular spaces.

Problem 12.6. Please design an example of (parameterized) granular network from a given (parameterized) network of granular spaces.

Problem 12.7. Please propose examples of optimization heuristics for generation of complex games.

Problem 12.8. Please provide illustrative examples of reasoning rules for interfaces between granular spaces.

Problem 12.9. Please discuss challenges for experience and common sense reasoning.

Problem 12.10. Please develop a dialogue with a selected chatbot supporting multi-level decomposition of a complex vague concept.

References

1. Alam, M., Shakil, K.A., Khan, S. (eds.): Internet of Things (IoT): Concepts and Applications. Springer, Cham, Switzerland (2020). doi:10.1007/978-3-030-37468-6
2. Anderson, J.R.: How Can the Human Mind Occur in the Physical Universe? Oxford University Press, New York, NJ (2007). doi:10.1093/acprof:oso/9780195324259.001.0001
3. Aristotle: On interpretation. `http://classics.mit.edu//Aristotle/interpretation.html` (2024). Accessed: 2024.09.20
4. Bargiela, A., Pedrycz, W.: Granular Computing: An Introduction. Kluwer Academic Publishers (2003)
5. Bargiela, A., Pedrycz, W.: Granular Computing. An Introduction, *The Springer International Series in Engineering and Computer Science (SECS)*, vol. 717. Springer, New York, NY (2003). doi:10.1007/978-1-4615-1033-8
6. Barsalou, L.W.: Perceptual symbol systems. Behavioral and Brain Sciences **22**, 577–660 (1999). doi:10.1017/S0140525X99002149
7. Barwise, J., Seligman, J.: Information Flow: The Logic of Distributed Systems. Cambridge University Press, Cambridge (1997). doi:10.1017/CBO9780511895968
8. Barwise, J., Seligman, J.: Information Flow: The Logic of Distributed Systems. Cambridge University Press (1997). doi:10.1023/A:1008350722315

9. Bazan, J.: Hierarchical classifiers for complex spatio-temporal concepts. In: J.F. Peters, A. Skowron, H. Rybiński (eds.) Transactions on Rough Sets **IX**: Journal Subline, *Lecture Notes in Computer Science*, vol. 5390, pp. 474–750. Springer, Heidelberg (2008). doi:10.1007/978-3-540-89876-4_26
10. Bazan, J.: Hierarchical classifiers for complex spatio-temporal concepts. In: J.F. Peters, A. Skowron, H. Rybiński (eds.) Transactions on Rough Sets IX: Journal Subline, *Lecture Notes in Computer Science*, vol. 5390, pp. 474–750. Springer, Heidelberg (2008). doi:10.1007/978-3-540-89876-4_26
11. Bower, J.M., Bolouri, H. (eds.): Computational Modeling of Genetic and Biochemical Networks. MIT Press (2001)
12. Brooks, Jr., F.P.: The Mythical Man-Month: Essays on Software Engineering, Anniversary Edition. Addison-Wesley (1995)
13. Brown, F.: Boolean Reasoning. Kluwer Academic Publishers, Dordrecht (1990)
14. Brown, N. (ed.): Artificial Intelligence in Drug Discovery, *Drug Discovery Series*, vol. 75. Royal Society of Chemistry, London (2021)
15. Campagner, A., Ciucci, D., Dorigatti, V.: Uncertainty representation in dynamical systems using rough set theory. Theoretical of Computer Science **908**, 28–42 (2022). doi:10.1016/J.TCS.2021.11.009
16. Chen, Z., Liu, B.: Lifelong Machine Learning, *Synthesis Lectures on Artificial Intelligence and Machine Learning*, vol. 38. Springer Nature, Cham (2018). doi:10.1007/978-3-031-01581-6
17. Ciesla, R.: The Book of Chatbots. From ELIZA to ChatGPT. Springer, Cham (2024). doi:10.1007/978-3-031-51004-5
18. Deutsch, D., Ekert, A., Lupacchini, R.: Machines, logic and quantum physics. Neural Computation **6**, 265–283 (2000). doi:10.2307/421056
19. Dong, L., Wang, R., Chen, D.: Incremental feature selection with fuzzy rough sets for dynamic data sets. Fuzzy Sets Syst. **467**, 108,503 (2023). doi:10.1016/J.FSS.2023.03.006
20. Dutta, S., Skowron, A.: Interactive granular computing model for intelligent systems. In: Z. Shi, M.K. Chakraborty, S. Kar (eds.) Intelligence Science III: 4th IFIP TC 12 International Conference, ICIS 2020, Durgapur, India, February 24-27, 2021, Revised Selected Papers, *IFIP Advances in Information and Communication Technology*, vol. 623, pp. 37–48. Springer International Publishing (2020). doi:10.1007/978-3-030-74826-5_4
21. Dutta, S., Skowron, A.: Interactive granular computing connecting abstract and physical worlds: An example. In: H. Schlingloff, T. Vogel (eds.) Proceedings of the 29th International Workshop on Concurrency, Specification and Programming (CS&P 2021), Berlin, Germany, September 27-28, 2021, *CEUR Workshop Proceedings*, vol. 2951, pp. 46–59. CEUR-WS.org (2021). URL `http://ceur-ws.org/Vol-2951/paper18.pdf`
22. Dutta, S., Skowron, A.: Interactive Granular Computing Connecting Abstract and Physical Worlds: An Example. In: H. Schlingloff, T. Vogel (eds.) Proceedings of the 29th International Workshop on Concurrency, Specification and Programming (CS&P 2021), Berlin, Germany, September 27-28, 2021, *CEUR Workshop Proceedings*, vol. 2951, pp. 46–59. CEUR-WS.org (2021). URL `ceur-ws.org/Vol-2951/paper18.pdf`
23. Dutta, S., Skowron, A.: Toward a computing model dealing with complex phenomena: Interactive granular computing. In: N.T. Nguyen, L. Iliadis, I. Maglogiannis, B. Trawinski (eds.) Computational Collective Intelligence - 13th International Conference, ICCCI 2021, Rhodes, Greece, September 29 - October 1, 2021, Proceedings, *Lecture Notes in Computer Science*, vol. 12876, pp. 199–214. Springer, Heidelberg (2021). doi:10.1007/978-3-030-88081-1_15
24. Dutta, S., Skowron, A.: Toward a computing model dealing with complex phenomena: Interactive granular computing. In: N.T. Nguyen, L. Iliadis, I. Maglogiannis, B. Trawinski (eds.) Computational Collective Intelligence - 13th International Conference, ICCCI 2021, Rhodes, Greece, September 29 - October 1, 2021, Proceedings, *Lecture Notes in Computer Science*, vol. 12876, pp. 199–214. Springer (2021). doi:10.1007/978-3-030-88081-1_15
25. Dutta, S., Skowron, A., Chakraborty, M.K.: Information flow in logic for distributed systems: Extending graded consequence. Information Sciences **491**, 232–250 (2019). doi:10.1016/J.INS.2019.03.057

26. Gerrish, S.: How Smart Machines Think. MIT Press, Cambridge, MA (2018)
27. Gershenson, C., Heylighen, F.: How can we think the complex? In: K. Richardson (ed.) In Managing Organizational Complexity: Philosophy, Theory and Application, pp. 47–61. Information Age Publishing, Greenwich, CT (2005)
28. Giabbanelli, P.J., Nápoles, G. (eds.): Fuzzy Cognitive Maps: Best Practices and Modern Methods. Springer, Cham (2024). doi:10.1007/978-3-031-48963-1
29. Gora, P.: Metaheuristics in optimization of complex processes. Phd thesis, University of Warsaw, Warsaw (2024). `https://www.mimuw.edu.pl/media/uploads/doctorates/thesis-pawel-gora.pdf`
30. Guégan, D., Hassani, B.K.: Risk Measurement. From Quantitative Measures to Management Decisions. Springer Nature, Cham (2019). doi:10.1007/978-3-030-02680-6
31. Harnad, S.: Categorical Perception: The Groundwork of Cognition. Cambridge University Press, New York, NY (1987)
32. Harnad, S.: The symbol grounding problem. Physica **D 42**, 335–346 (1990). doi:10.1016/0167-2789(90)90087-6
33. Hastie, T., Tibshirani, R., Friedman, J.H.: The Elements of Statistical Learning: Data Mining, Inference, and Prediction. Springer-Verlag, Heidelberg (2009). doi:10.1007/b94608
34. Hastie, T., Tibshirani, R., Friedman, J.H.: The Elements of Statistical Learning: Data Mining, Inference, and Prediction. Springer-Verlag, Heidelberg (2009). doi:10.1007/b94608
35. Jankowski, A.: Interactive Granular Computations in Networks and Systems Engineering: A Practical Perspective, *Lecture Notes in Networks and Systems*, vol. 17. Springer, Heidelberg (2017). doi:10.1007/978-3-319-57627-5
36. Jankowski, A.: Interactive Granular Computations in Networks and Systems Engineering: A Practical Perspective, *Lecture Notes in Networks and Systems*, vol. 17. Springer, Heidelberg (2017)
37. Jankowski, A., Skowron, A.: Wisdom technology: A rough-granular approach. In: M. Marciniak, A. Mykowiecka (eds.) Bolc Festschrift, *Lectures Notes in Computer Science*, vol. 5070, pp. 3–41. Springer, Heidelberg (2009). doi:10.1007/978-3-642-04735-0_1
38. Jansen, S.: Machine Learning for Algorithmic Trading. Packt, Birmingham - Mumbai (2020)
39. Japkowicz, N., Shah, M.: Evaluating Learning Algorithms: A Classification Perspective. Cambridge University Press, New York, NY (2011)
40. Jiao, L., Song, X., You, C., Liu, X., Li, L., Chen, P., Tang, X., Feng, Z., Liu, F., Guo, Y., Yang, S., Li, Y., Zhang, X., Ma, W., Wang, S., Bai, J., Hou, B.: AI meets physics: a comprehensive survey. Artiicial Intelligence Review **57**(9), 256 (2024). doi:10.1007/S10462-024-10874-4
41. Jing, T., Wang, C., Pedrycz, W., Li, Z., Succi, G., Zhou, M.: Granular models as networks of associations of information granules: A development scheme via augmented principle of justifiable granularity. Applied Soft Computing **115**, 108,062 (2022). doi:10.1016/j.asoc.2021.108062
42. Kahneman, D.: Maps of bounded rationality: Psychology for behavioral economics. The American Economic Review **93**, 1449–1475 (2002). doi:10.1257/000282803322655392
43. Kamath, U., Keenan, K., Somers, G., Sorenson, S.: Large Language Models: A Deep Dive. Bridging Theory and Practice. Springer, Cham (2024). doi:10.1007/978-3-031-65647-7
44. Karniadakis, G.E., Kevrekidis, I.G., Lu, L., Perdikaris, P., Wang, S., Yang, L.: Physics-informed machine learning. Nature Reviews Physics **3**, 422–440 (2021). doi:10.1038/s42254-021-00314-5
45. Kosko, B.: Fuzzy cognitive maps. International journal of Man-machine Studies **24**, 65–75 (1986). doi:10.1016/S0020-7373(86)80040-2
46. Krishnan, S., Anand, A.J., Srinivasan, R., Kavitha, R., Suresh, S. (eds.): Handbook on Federated Learning: Advances, Applications and Opportunities. CRC Press,Taylor & Francis Group, Boca Raton, FL (2024). doi:10.1201/9781003384854
47. Martin, W.M. (ed.): Theories of Judgment. Psychology, Logic, Phenomenology. Cambridge University Press, New York (2006)
48. Maslov, A.H.: A theory of human motivation. Psychological Review **50**, 370–396 (1943). doi:10.1037/h0054346

49. Matthews, D., Spielberg, A., Rus, D., Kriegman, S., Bongard, J.: Efficient automatic design of robots. PNAS **120**, 1–7 (2023). doi:10.1073/pnas.2305180120
50. Mitchell, M.: Artificial Intelligence: A Guide for Thinking Humans. Farrar, Straus, Giroux, New York, NY (2019)
51. Mitchell, M.: Abstraction and analogy-making in artificial intelligence. Annals Reports of the New York Academy of Sciences **1505**, 79–101 (2021). doi:10.1111/nyas.14619
52. Mitchell, M.: Frontiers in Collective Intelligence: A Workshop Report 2021. Santa Fe Institute, Santa Fe, NM (2021). URL `arxiv.org/abs/2112.06864`
53. Monarch, R.M.: Human-in-the-Loop Machine Learning. Active Learning and Annotation for Human-Centered AI. MANNING, Shelter Island, NY (2021)
54. Monarch, R.M.: Human-in-the-Loop Machine Learning. Active Learning and Annotation for Human-Centered AI. MANNING, Shelter Island, NY (2021)
55. Nanay, B.: Perception: The Basics. Routledge, Taylor& Francis, New York, NY (2024)
56. Nápoles, G., Falcon, R., Papageorgiou, E., Bello, R., Vanhoof, K.: Rough cognitive ensembles. International Journal of Approximate Reasoning **85**, 79–96 (2017). doi:https://doi.org/10.1016/j.ijar.2017.03.011
57. Nápoles, G., Grau, I., Papageorgiou, E., Bello, R., Vanhoof, K.: Rough cognitive networks. Knowledge Based Systems **91**, 46–61 (2016). doi:10.1016/J.KNOSYS.2015.10.015
58. Nguyen, H.S.: Approximate boolean reasoning: Foundations and applications in data mining. Transactions on Rough Sets V: Journal Subline **5**, 344–523 (2006). doi::10.1007/11847465_16
59. Nguyen, H.S., Skowron, A.: Rough sets: From rudiments to challenges. In: Skowron and Suraj [**?**], pp. 75–173. doi:10.1007/978-3-642-30344-9_3
60. Nguyen, H.S., Skowron, A.: Rough sets: From rudiments to challenges. In: A. Skowron, Z. Suraj (eds.) Rough Sets and Intelligent Systems. Professor Zdzislaw Pawlak in Memoriam, Series Intelligent Systems Reference Library, pp. 75–173. Springer (2013). doi:10.1007/978-3-642-30344-9; 10.1007/978-3-642-30341-8
61. Noë, A.: Action in Perception. MIT Press (2004)
62. Omicini, A., Ricci, A., Viroli, M.: The multidisciplinary patterns of interaction from sciences to computer science. In: D. Goldin, S. Smolka, P. Wegner (eds.) Interactive Computation: The New Paradigm, pp. 395–414. Springer (2006). doi:10.1007/3-540-34874-3
63. Ortiz Jr., C.L.: Why we need a physically embodied Turing test and what it might look like. AI Magazine **37**, 55–62 (2016). doi:10.1609/aimag.v37i1.2645
64. Pal, S.: Rough Set and Deep Learning: Some Concepts. Academia Letters (Article 1849), 1–6 (July 2021). doi:10.20935/AL1849
65. Pawlak, Z.: Rough sets. International Journal of Computer and Information Sciences **11**, 341–356 (1982). doi:10.1007/BF01001956
66. Pawlak, Z.: Rough Sets: Theoretical Aspects of Reasoning about Data, *System Theory, Knowledge Engineering and Problem Solving*, vol. 9. Kluwer Academic Publishers, Dordrecht, The Netherlands (1991). doi:10.1007/978-94-011-3534-4
67. Pawlak, Z., Skowron, A.: Rough sets and boolean reasoning. Information Sciences **177**(1), 41–73 (2007)
68. Pawlak, Z., Skowron, A.: Rudiments of rough sets. Information Sciences **177**(1), 3–27 (2007). doi:10.1016/j.ins.2006.06.003
69. Pearl, J.: Causal inference in statistics: An overview. Statistics Surveys **3**, 96–146 (2009). doi:10.1214/09-SS057
70. Pedrycz, W.: Granular Computing. Analysis And Design of Intelligent Systems. CRC Press, Taylor & Francis (2013)
71. Pedrycz, W.: Granular computing for data analytics: a manifesto of human-centric computing. IEEE/CAA Journal of Automatica Sinica **5**, 1025–1034 (2018). doi:10.1109/jas.2018.7511213
72. Pedrycz, W.: An Introduction to Computing with Fuzzy Sets. Analysis, Design, and Applications, *Intelligent Systems Reference Library*, vol. 190. Springer Nature, Cham (2021). doi:10.1007/978-3-030-52800-3

73. Pedrycz, W.: An Introduction to Computing with Fuzzy Sets: Analysis, Design, and Applications, vol. 190, chap. Granulation-Degranulation Processes, pp. 161–173. Springer International Publishing, Cham (2021). doi:10.1007/978-3-030-52800-3_11
74. Pedrycz, W., Homenda, W.: Building the fundamentals of granular computing: a principle of justifiable granularity. Applied Soft Computing **13**, 4209–4218 (2013). doi:10.1016/j.asoc.2013.06.017
75. Pedrycz, W., Skowron, S., Kreinovich, V. (eds.): Handbook of Granular Computing. John Wiley & Sons, Hoboken, NJ (2008). doi:10.1002/9780470724163
76. Polkowski, L., Skowron, A.: Towards adaptive calculus of granules. In: L.A. Zadeh, J. Kacprzyk (eds.) Computing with Words in Information/Intelligent Systems, pp. 201–227. Physica-Verlag, Heidelberg (1999). doi:10.1007/978-3-7908-1873-4
77. Polkowski, L., Skowron, A.: Rough mereological calculi of granules: A rough set approach to computation. Computational Intelligence: An International Journal **17**(3), 472–492 (2001). doi:10.1111/0824-7935.00159
78. Povalej, P., Verlic, M., Stiglic, G.: Discovery systems. In: R.A. Meyers (ed.) Encyclopedia of Complexity and Systems Science, pp. 1982–2002. Springer New York, New York, NY (2009). doi:10.1007/978-0-387-30440-3_125
79. Qin, J., Martínez, L., Pedrycz, W., Ma, X., Liang, Y.: An overview of granular computing in decision-making: Extensions, applications, and challenges. Information Fusion **98**, 101,833 (2023). doi:10.1016/j.inffus.2023.101833
80. Rissanen, J.: Minimum-description-length principle. In: S. Kotz, N. Johnson (eds.) Encyclopedia of Statistical Sciences, pp. 523–527. John Wiley & Sons, New York, NY (1985)
81. Russell, S.J., Norvig, P.: Artificial Intelligence A Modern Approach. Pearson Education, Inc., Upper Saddle River, NJ (2021). 4th edition
82. Russell, S.J., Norvig, P.: Artificial Intelligence A Modern Approach. Pearson Education, Inc., Upper Saddle River, NJ (2021). 4th edition
83. Santos, J.C., Santos, M.S., Abreu, P.H.: An interpretable human-in-the-loop process to improve medical image classification. In: I. Miliou, N. Piatkowski, P. Papapetrou (eds.) Advances in Intelligent Data Analysis XXII - 22nd International Symposium on Intelligent Data Analysis, IDA 2024, Stockholm, Sweden, April 24-26, 2024, Proceedings, Part I, *Lecture Notes in Computer Science*, vol. 14641, pp. 179–190. Springer (2024). doi:10.1007/978-3-031-58547-0_15
84. Shneiderman, B.: Human – Centered AI. Oxford University Press, Oxford, UK (2022). doi:10.1093/oso/9780192845290.001.0001
85. Shneiderman, B.: Human – Centered AI. Oxford University Press, Oxford, UK (2022). doi:10.1093/oso/9780192845290.001.0001
86. Skowron, A.: Toward intelligent systems: Calculi of information granules. Bulletin of the International Rough Set Society **5**(1-2), 9–30 (2001)
87. Skowron, A., Dutta, S.: Information system in the light of interactive granular computing. In: M. Hu, C. Cornelis, Y. Zhang, P. Lingras, D. Ślęzak, J. Yao (eds.) Rough Sets - International Joint Conference, IJCRS 2024, Halifax, Canada, May 17-20, 2024, Proceedings, Part I, *Lecture Notes in Computer Science*, vol. 14839, pp. 223–237. Springer, Cham (2024). doi:10.1007/978-3-031-65665-1_14. (in print)
88. Skowron, A., Jankowski, A.: Rough Sets and Interactive Granular Computing. Fundamenta Informaticae **147**(2-3), 371–385 (2016). doi:10.3233/FI-2016-1413
89. Skowron, A., Jankowski, A., Dutta, S.: Interactive granular computing. In: Granular Computing, pp. 95–113. Springer, Heidelberg (2016). (DOI: 10.1007/s41066-015-0002-1)
90. Skowron, A., Rauszer, C.: The discernibility matrices and functions in information systems. In: R. Słowiński (ed.) Intelligent Decision Support - Handbook of Applications and Advances of the Rough Sets Theory, *System Theory, Knowledge Engineering and Problem Solving*, vol. 11, pp. 331–362. Kluwer Academic Publishers, Dordrecht, The Netherlands (1992)
91. Skowron, A., Ślęzak, D.: Rough sets in interactive granular computing: Toward foundations for intelligent systems interacting with human experts and complex phenomena. In: Proceedings of the International Joint Conference on Rough Sets (IJCRS 2023), Cracow, October

5-8, 2023, *Lecture Notes in Computer Science*, vol. 14481, pp. xv–xxxvii. Springer, Berlin, Heidelberg (2023). doi:doi.org/10.1007/978-3-031-50959-9
92. Skowron, A., Ślęzak, D.: A rough set perspective on intelligence systems (2024). (submitted)
93. Skowron, A., Stepaniuk, J.: Tolerance approximation spaces. Fundam. Informaticae **27**(2/3), 245–253 (1996). doi:10.3233/FI-1996-272311
94. Skowron, A., Stepaniuk, J.: Information granules: Towards foundations of granular computing. International Journal of Intelligent Systems **16**(1), 57–86 (2001). doi:10.1002/1098-111X(200101)16:1<57::AID-INT6>3.0.CO;2-Y
95. Skowron, A., Stepaniuk, J.: Information granules: Towards foundations of granular computing. International Journal of Intelligent Systems **16**(1), 57–86 (2001). doi:10.1002/1098-111X(200101)16:1<57::AID-INT6>3.0.CO;2-Y
96. Skowron, A., Stepaniuk, J.: Information granules and rough-neural computing. In: S.K. Pal, L. Polkowski, A. Skowron (eds.) Rough-Neural Computing: Techniques for Computing with Words, Cognitive Technologies, pp. 43–84. Springer-Verlag, Heidelberg (2004)
97. Skowron, A., Stepaniuk, J.: Information granules and rough-neural computing. In: S.K. Pal, L. Polkowski, A. Skowron (eds.) Rough-Neural Computing: Techniques for Computing with Words, Cognitive Technologies, pp. 43–84. Springer-Verlag, Heidelberg (2004). doi:10.1007/978-3-642-18859-6
98. Skowron, A., Stepaniuk, J.: Toward rough set based insightful reasoning in intelligent systems p. 2 (2024). (submitted)
99. Skowron, A., Suraj, Z. (eds.): Rough Sets and Intelligent Systems. Professor Zdzislaw Pawlak in Memoriam. Series Intelligent Systems Reference Library. Springer (2013). doi:10.1007/978-3-642-30344-9; 10.1007/978-3-642-30341-8
100. Stepaniuk, J.: Rough – Granular Computing in Knowledge Discovery and Data Mining, *Studies in Computational Intelligence*, vol. 152. Springer (2008). doi:10.1007/978-3-54070801-8
101. Stepaniuk, J. (ed.): Rough-Granular Computing in Knowledge Discovery and Data Mining. Studies in Computational Intelligence. Springer, Heidelberg (2008)
102. Stepaniuk, J., Skowron, A.: Three–way approximation of decision granules based on the rough set approach. International Journal of Approximate Reasoning **155**, 1–16 (2023). doi:10.1016/j.ijar.2023.01.003
103. Stone, P.: Layered Learning in Multi-Agent Systems: A Winning Approach to Robotic Soccer. The MIT Press, Cambridge, MA (2000)
104. Stone, P.: Layered Learning in Multi-Agent Systems: A Winning Approach to Robotic Soccer. The MIT Press, Cambridge, MA (2000)
105. Thakur, K., Barker, H.G., Pathan, A.S.K. (eds.): Artificial Intelligence and Large Language Models: An Introduction to the Technological Future. Chapman and Hall/CRC, Boca Raton, FL (2024). doi:10.1201/9781003474173
106. Thiele, L.P.: The Heart of Judgment: Practical Wisdom, Neuroscience, and Narrative. Cambridge University Press, Cambridge, UK (2010). doi:10.1017/CBO9780511498718
107. Valiant, L.: Research directions. `https://people.seas.harvard.edu/~valiant/researchinterests.htm` (2024). Accessed: 2024.05.28
108. Vapnik, V.: Statistical Learning Theory. John Wiley & Sons, New York, NY (1998)
109. Veale, T., Cardoso, F.A. (eds.): Computational Creativity. The Philosophy and Engineering of Autonomously Creative Systems. Computational Synthesis and Creative Systems. Springer, Cham (2019). doi:10.1007/978-3-319-43610-4
110. Webster's New World College Dictionary: Wisdom. `https://www.yourdictionary.com/wisdom` (2024). Accessed: 2024.11.20
111. Wolski, M.: Rough sets in terms of discrete dynamical systems. In: M.S. Szczuka, M. Kryszkiewicz, S. Ramanna, R. Jensen, Q. Hu (eds.) Rough Sets and Current Trends in Computing - 7th International Conference, RSCTC 2010, Warsaw, Poland, June 28-30,2010. Proceedings, *Lecture Notes in Computer Science*, vol. 6086, pp. 237–246. Springer (2010). doi:doi.org/10.1007/978-3-642-13529-3_26
112. Zadeh, L.A.: Fuzzy sets. Information and Control **8**, 338–353 (1965). doi:10.1016/S0019-9958(65)90241-X

113. Zadeh, L.A.: Toward a theory of fuzzy information granulation and its centrality in human reasoning and fuzzy logic. Fuzzy Sets Systems **90**(2), 111–127 (1997). doi:10.1016/S0165-0114(97)00077-8
114. Zadeh, L.A.: A new direction in AI: Toward a computational theory of perceptions. AI Magazine **22**(1), 73–84 (2001). doi:10.1609/aimag.v22i1.1545
115. Zadeh, L.A.: A new direction in AI: Toward a computational theory of perceptions. AI Magazine **22**(1), 73–84 (2001). doi:10.1609/aimag.v22i1.1545
116. Zadeh, L.A.: Foreword. In: S.K. Pal, L. Polkowski, A. Skowron (eds.) Rough-Neural Computing: Techniques for Computing with Words, Cognitive Technologies, pp. IX–XI. Springer-Verlag, Heidelberg (2004). doi:doi.org/10.1007/978-3-642-18859-6