Chapter 3
Information Granules and Rough-Neurocomputing

Andrzej Skowron[1] and Jarosław Stepaniuk[2]

[1] Institute of Mathematics, Warsaw University, Banacha 2, 02-097 Warsaw, Poland
skowron@mimuw.edu.pl
[2] Department of Computer Science, Białystok University of Technology, Wiejska 45A, 15-351 Białystok, Poland
jstepan@ii.pb.bialystok.pl

Summary. In this chapter we discuss the foundations of rough-neurocomputing (RNC). We introduce information granule systems and information granules in such systems. Information granule networks, called approximate reasoning schemes (AR schemes), are used to represent information granule constructions. We discuss the foundations of RNC using an analogy of information granule networks with neural networks. RNC is a basic paradigm of granular computing (GC). This paradigm makes it possible to tune AR schemes to construct relevant information granules, e.g., satisfying a given specification to a satisfactory degree. One of the goals of our project is to develop methods based on rough-neurocomputing for computing with words (CW).

1 Introduction

Information granules are intuitively described in the literature as collections of entities that are arranged together due to their similarity, functional adjacency, or indiscernibility relation. The process of forming information granules is referred to as information granulation. Information granulation belongs to intensively studied topics in soft computing (see, e.g., [42–44]). One of the recently emerging approaches to deal with information granulation, called granular computing, is based on information granule calculi (see, e.g., [27,35]). The development of such calculi is important for making progress in many areas such as object identification by autonomous systems (see, e.g., [4,40]), web mining (see, e.g., [10]), spatial reasoning (see, e.g., [6]), and sensor fusion (see, e.g., [3,22]). One of the main goals of GC is to achieve computing with words (see, e.g., [42–44]). The granular computing paradigm, as opposed to numeric-computing, is knowledge oriented [14,39]. Computations in granular computing are performed on information granules. Developing methods of GC is also crucial for making progress in knowledge discovery and data mining. The main reason for this is that knowledge-based processing is a cornerstone of knowledge discovery and data mining.

There is a need to develop information granulation tools for constructing complex information granules. For example, in spatial and temporal reasoning, one should be able to determine if a road is safe on the basis of sensor measurements [40] or to classify situations in complex games such as soccer [37]. These complex information granules constitute a form of information fusion. Any calculus of complex information granules should make it possible to

- Deal with the vagueness of information granules.
- Develop strategies for inducing multilayered schemes of complex granule construction.
- Derive robust (stable) information granule construction schemes with respect to deviations of the granules from which they are constructed.
- Develop adaptive strategies for reconstructing induced schemes for complex information granule synthesis.

To deal with vagueness, one can adopt fuzzy set theory [41] or rough set theory [19] either separately or in combination [20]. The second requirement is related to the problem of understanding reasoning from measurements relative to perception [43], to concept approximation learning in layered learning [37], and to fusion of information from different sources [42–44]. Methods of searching for approximate reasoning schemes as schemes of new information granule construction have been investigated using rough mereological tools [24,25,27,29,31]. In general, those methods return hierarchical schemes for new information granule construction. The process of AR schemes construction is related to ideas of cooperation, negotiation, and conflict resolution in multiagent systems [2,9]. Among important topics studied in relation to AR schemes are methods for specifying operations on information granules. In particular, AR schemes are useful in constructing information granules from data and background knowledge and in supplying methods for inducing these hierarchical schemes of information granule construction. One of the possible approaches is to learn such schemes using evolutionary strategies [13]. The robustness of the scheme means that any scheme produces a higher order information granule that is a clump (e.g., a set) of close information granules rather than a single information granule. Such a clump is constructed by means of the scheme from input clumps defined by deviations (up to acceptable degrees) of input information granules from standard (prototype) granules.

It is worthwhile mentioning that modeling complex phenomena requires us to use complex information granules representing local models (perceived by local agents) that are fused. This process involves negotiations between agents [9] to resolve contradictions and conflicts in local modeling. This kind of modeling will become more and more important in solving complex real-life problems that we cannot model using traditional analytical approaches. If the latter approaches can be applied to modeling of such problems, they lead to exact models. However, the necessary assumptions used to build them for complex real-life problems often make the resulting solutions *too far* from reality to be accepted as solutions of such problems.

Using multiagent terminology, let us also observe that local agents perform operations on information granules they *understand*. Hence, granules received as operation arguments from other agents should be approximated by properly tuned approximation spaces creating interfaces between agents. The process of tuning the approximation space [29,33] parameters in AR schemes corresponds to the tuning of weights in neural networks. The methods for inducing AR schemes to transform information granules into developed information granules using rough set [12,19] and rough mereological methods [24,29] in hybridization with other soft computing approaches (i.e., neural networks [30], fuzzy sets [20,41,44], and evolutionary programming [13,17]) create a core for rough-neurocomputing. In RNC, computations are performed on information granules by schemes analogous to neural networks. The aim of such schemes is, for example, to construct information granules satisfying, at least to a satisfactory degree, a given specification or to preserve some invariants during computation.

One of the basic research directions in RNC concerns relationships between information granules and words (linguistic terms) in a natural language. Another direction of the research concerns the possibility of using induced AR schemes that match, to a satisfactory degree, reasoning schemes in natural language. Further research in this direction will create strong links between RNC and CW.

In this chapter, we discuss the foundations for RNC. We introduce information granule systems and information granules in such systems. Any granule system consists of a parameterized formula set and a finite parameterized relational system in which the semantics of such formulas is defined. Information granules in a given granule system are elements of a parameterized formula set.

We present several examples of information granule systems and information granules in such systems. Different kinds of information granules and inclusion (closeness) measures between information granules are discussed in the following sections of this chapter. Networks of information granule construction are represented by approximate reasoning schemes. We discuss the foundations of the RNC paradigm using an analogy of information granule networks to neural networks.

The rough-neurocomputing paradigm is a basic paradigm of GC. One of the goals of our project is to develop methods based on RNC for computing with words.

2 Motivation: Illustrative Example

In this section, we consider an example that shows the need for information granulation, construction of AR schemes, and RNC. The example is related to estimating a situation on a road on the basis of sensor measurements made by an unmanned helicopter [40]. Let us assume that we would like to estimate whether the situation on the road is dangerous (see Fig. 1).

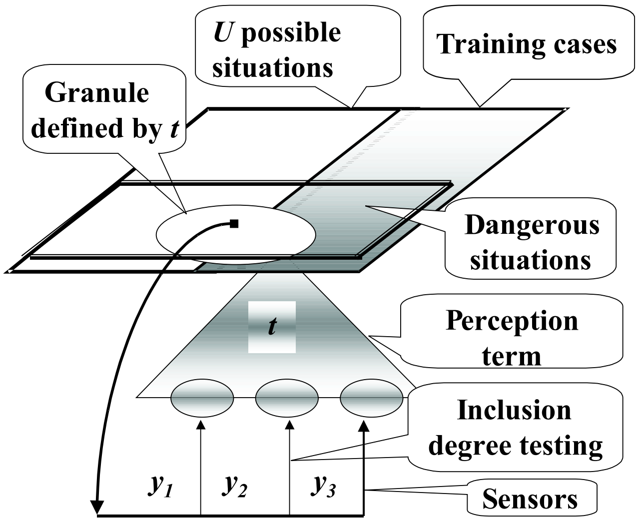

Fig. 1. Classification of situations

The problem is how to induce large patterns, sufficiently included in the concept *dangerous* in terms of sensor measurements. Such patterns can be used to approximate the concept. This can be done on the basis of *perception trees* representing reasoning schemes in natural language, based, for example, on behavior of cars on the road. We expect that AR schemes (or their clusters) can be constructed along such perception trees. They will make it possible to estimate whether sensor measurements corresponding to a given situation are sufficiently close to input information granules of such schemes. If it is true, one can conclude that it is highly probable that the analyzed situation is dangerous.

3 Information Granule Systems

In this section, we present a basic notion of our approach, i.e., the information granule system. Any information granule system is a tuple,

$$S = (G, R, Sem)\,, \tag{1}$$

where

1. G is a set of parameterized formulas, called *information granules*;
2. R is a (parameterized) relational structure; and
3. *Sem* is the semantics of G in R.

We assume that with any information granule system the following are associated:

1. H, a set of *granule inclusion degrees* with a partial order relation $\leq$ that defines on H a structure used to compare the inclusion degrees; we assume that H contains the lowest degree 0 and the largest degree 1.

2. $\nu_p \subseteq G \times G$, a binary relation that is *a part to a degree at least p* between information granules from G, called *rough inclusion*. (Instead of $\nu_p(g, g')$, we also write $\nu(g, g') \geq p$).

The components of an information granule system are parameterized. This means that we deal with parameterized formulas and a parameterized relational system. The parameters are tuned to make it possible to construct finally relevant information granules, i.e., granules satisfying specification or/and some optimization criteria. Parameterized formulas can consist of parameterized subformulas. The value set of parameters labeling a subformula defines a set of formulas. By tuning parameters in an optimization process and/or information granule construction, a relevant subset of parameters is extracted and used for constructing the target information granule.

There are two kinds of computations on information granules: computations on information granule systems and computations on information granules in such systems. The first aim at constructing of relevant information granule systems defining parameterized approximation spaces for concept approximations used on different levels of target information granule construction. The goal of the second is to construct information granules across such information granule systems to obtain target information granules, e.g., satisfying a given specification (at least to a satisfactory degree).

Examples of complex granules are tolerance granules created by similarity (tolerance) relation between elementary granules, decision rules, sets of decision rules, sets of decision rules with guards, information systems, or decision tables [27,31,35]. The most interesting class of information granules consists of information granules that approximate concepts specified in natural language by experimental data tables and background knowledge.

One can consider as an example of the set H of granule inclusion degrees the set of binary sequences of fixed length with the relation ν defined by the lexicographical order. This degree structure can be used to measure the inclusion degree between granule sequences or to measure the matching degree between granules representing classified objects and granules describing the left-hand sides of decision rules in simple classifiers [29]. However, one can consider more complex degree granules assuming the degree of inclusion of granule g_1 in granule g_2 as the granule representing a collection of common parts of these two granules g_1 and g_2.

New information granules can be defined by operations performed on already constructed information granules. Examples of such operations are set theoretical operations (defined by propositional connectives). However, there are other operations widely used in machine learning or pattern recognition ([15]) for constructing classifiers. These are the $Match$ and $Conflict\_res$ operations [29]. The $Match$ operation is used to construct a granule describing the matching result of elementary granules describing classified objects by granules representing the left-hand sides of decision rules. The $Conflict\_res$ is an operation producing from this matching granule the resulting granule, e.g., identifying a relevant decision class for any classified object. It is worthwhile mentioning yet another important class of operations, namely, those defined by data tables called decision tables [35]. From these decision tables, decision rules specifying operations can be induced. More complex operations on information granules are so called transducers [4]. They have been introduced to use background knowledge (not necessarily in the form of data tables) in constructing new granules. One can consider theories or their clusters as information granules. Reasoning schemes in natural language define the most important class of operations on information granules to be investigated. One of the basic problems of such operations and schemes of reasoning is how to approximate them by available information granules, e.g., constructed from sensor measurements.

In an information granule system, the relation ν_p as a part to a degree at least p has a special role. It satisfies some additional natural axioms and, additionally, some axioms of rough mereology [25]. It can be shown that the rough mereological approach, built on the basis of the relation to be part, to a degree, generalizes the rough set and fuzzy set approaches. Moreover, such relations can be used to define other basic concepts, such as closeness of information granules, their semantics, the indiscernibiliy and discernibility of objects, information granule approximation and approximation spaces, the perception structure of information granules as well as the notion of ontology approximation. One can observe that the relation to be part to a degree can be used to define operations on information granules corresponding to a generalization of already defined information granules.

Let us finally note that new information granule systems can be defined using already constructed information granule systems. This leads to a hierarchy of information granule systems.

4 Basic Examples of Information Granule Systems

In the following sections, we present examples of information granule systems. Each of them will be specified by a set of information granules, relational structure, and semantics of information granules. We will also discuss some operations on information granule system that make it possible to construct new relevant information granule systems in searching for the target information granule systems representing approximation spaces.

4.1 Elementary Information Granule Systems

Let us consider an example from [19]. We define a language $L_{\mathcal{A}}$ used for elementary granule description, where $\mathcal{A} = (U, A)$ is an information system that is a relational structure for the example discussed. The syntax of $L_{\mathcal{A}}$ is defined recursively by

1. $(a \in V) \in L_{\mathcal{A}}$, for any $a \in A$ and $V \subseteq V_a$.
2. If $\alpha \in L_{\mathcal{A}}$, then $\neg\alpha \in L_{\mathcal{A}}$.
3. If $\alpha, \beta \in L_{\mathcal{A}}$, then $\alpha \wedge \beta \in L_{\mathcal{A}}$.
4. If $\alpha, \beta \in L_{\mathcal{A}}$, then $\alpha \vee \beta \in L_{\mathcal{A}}$.

The semantics of formulas from $L_{\mathcal{A}}$ with respect to an information system $\mathcal{A}$ is defined recursively by

1. $Sem_{\mathcal{A}}(a \in V) = \{x \in U : a(x) \in V\}$.
2. $Sem_{\mathcal{A}}(\neg\alpha) = U - Sem_{\mathcal{A}}(\alpha)$.
3. $Sem_{\mathcal{A}}(\alpha \wedge \beta) = Sem_{\mathcal{A}}(\alpha) \cap Sem_{\mathcal{A}}(\beta)$.
4. $Sem_{\mathcal{A}}(\alpha \vee \beta) = Sem_{\mathcal{A}}(\alpha) \cup Sem_{\mathcal{A}}(\beta)$.

Elementary granules. In an information system $\mathcal{A} = (U, A)$, elementary granules are defined by $EF_B(x)$, where EF_B is a conjunction of selectors (descriptors) of the form $a = a(x)$, $B \subseteq A$ and $x \in U$. For example, the meaning of an elementary granule $a = 1 \wedge b = 1$ is defined by

$$Sem_{\mathcal{A}}(a = 1 \wedge b = 1) = \{x \in U : a(x) = 1 \;\&\; b(x) = 1\}.$$

In the elementary information granule system discussed, the set of elementary granules consists of a set of conjunctions of selectors, the set H of inclusion degrees is a subset of $[0, 1]$ and $\nu_p(EF_B, EF'_B)$ if and only if

$$\frac{card\,[Sem_{\mathcal{A}}(EF_B) \cap Sem_{\mathcal{A}}(EF'_B)]}{card\,[Sem_{\mathcal{A}}(EF_B)]} \geq p.$$

The number of conjuncts in the granule can be taken as one of parameters to be tuned; this is well known as the drooping condition technique in machine learning [15].

One can extend the set of elementary granules assuming that if α is any Boolean combination of descriptors over A, then $(\overline{B}\alpha)$ and $(\underline{B}\alpha)$ define the syntax of elementary granules, too, for any $B \subseteq A$. The reader can find more details on granules defined by rough set approximations in [36].

One can consider extension of elementary granules defined by a tolerance relation. Let $\mathcal{A} = (U, A)$ be an information system, and let τ be a tolerance relation on elementary granules of $\mathcal{A}$. Any pair (α, τ) is called a τ-*elementary granule*. The semantics $Sem_{\mathcal{A}}[(\alpha, \tau)]$ of (α, τ) is the family $\{Sem_{\mathcal{A}}(\beta) : (\beta, \alpha) \in \tau\}$. Parameters to be tuned in searching for a relevant tolerance granule can be its support (represented by the number of supporting objects) and its degree of inclusion (or closeness) in some other granules as well as parameters specifying the tolerance relation.

Certainly, one can consider other forms of elementary granules. For example, one can define an elementary information granule as any tuple consisting of a Boolean combination of descriptors or its semantics in a given information system. The choice depends on applications.

4.2 Operations on Information Granule Systems

The simplest operations on information granule systems are set theoretical operations. For, example if $S = (G, R, Sem)$ and $S' = (G', R', Sem')$ are two information granule systems, then $pair(S, S')$ is an information granule system in which information granules are pairs (α, α') where $\alpha \in G$, $\alpha' \in G'$ and the semantics of (α, α') is defined by $(Sem(\alpha), Sem'(\alpha'))$. Analogously, other set theoretical operations, such as $tuple(S_1, \ldots, S_k)$ or $set(S_1, \ldots, S_k)$ can be defined. One can also define projection and extension operations, helping to solve feature extraction and selection problems.

First, let us discuss some examples of possible applications of such constructions of new information granule systems. Granules defined by rules in information systems are examples of information granules in $pair(S, S')$. Let $\mathcal{A} = \mathcal{A}'$ be an information system and let (α, β) be a new information granule in $pair(S, S')$ received from the rule *if* α *then* β, where α, β are elementary granules in S, S', respectively. If the right-hand sides of rules represent decision classes, then the among parameters to be tuned in classification is the number of conjuncts on the left-hand sides of rules. A typical goal is to search for the minimal number of such conjuncts (corresponding to the largest generalization), which still guarantees a satisfactory degree of inclusion in a corresponding decision class [12,15]. One can now consider the new information granule system $set[pair(S, S'), \ldots, pair(S, S')]$ in which information granules represent sets of rules. An important problem in machine learning is the problem of searching for a granule of the smallest cardinality sufficiently close to one given in such a system, i.e., a searching problem for representating of a given rule collection by another set of rules of sufficiently small cardinality and sufficiently close to the collection.

In examples presented, we have discussed parameterized information granules. We have pointed out that the process of parameter tuning is used to induce relevant (for a given task) information granules. In particular, the process of parameter tuning is performed to obtain a satisfactory degree of inclusion (closeness) of information granules.

We have not yet discussed how the rough inclusion relations in the resulting information granule systems are defined. In the following section, we discuss inclusion and closeness relations for information granules.

4.3 Examples of Granule Inclusion and Closeness

In this section, we will discuss inclusion and closeness of different information granules. Inclusion and closeness are basic concepts related to information granules [27,35]. Using them, one can measure the closeness of a constructed granule to a target granule and the robustness of the construction scheme with respect to deviations of information granules that are components of the construction. For details and examples of closeness relations, refer to [27,35].

The choice of inclusion or closeness definition depends very much on the area of application and data analyzed. This is the reason that we have decided to introduce a separate section with this more subjective (or task oriented) part of granule semantics.

The inclusion relation between granules g, g' to the degree at least p (i.e., $\nu(g, g') \geq p$), will be denoted by $\nu_p(g, g')$. By $\underline{\nu}_p(g, g')$, we denote the inclusion of g in g' to the degree at most p, i.e., that $\nu(g, g') \leq p$ holds. Similarly, the closeness relation between granules g, g', to the degree at least p, will be denoted by $cl_p(g, g')$. By p, we denote a vector of parameters (e.g., from the interval [0,1] of real numbers).

A general scheme for constructing of hierarchical granules and their closeness can be described by the following recursive metarule: if granules of order $\leq k$ and their closeness have been defined, then the closeness $cl_p(g, g')$ (at least to the degree p) between granules g, g' of order $k+1$ can be defined by applying an appropriate operator F to closeness values of components of g, g', respectively.

Elementary granules. We have introduced the simplest case of granules in information system $\mathcal{A} = (U, A)$. They are defined by $EF_B(x)$, where EF_B is a conjunction of selectors of the form $a = a(x)$, where $a \in B \subseteq A$ and $x \in U$. Let

$$G_{\mathcal{A}} = \{EF_B(x) : \emptyset \neq B \subseteq A \ \&\ x \in U\}. \tag{2}$$

In the standard rough set model [19], elementary granules describe indiscernibility classes with respect to some subsets of attributes. In a more general setting [33], tolerance (similarity) classes are described. The crisp inclusion of α in β is defined by

$$Sem_{\mathcal{A}}(\alpha) \subseteq Sem_{\mathcal{A}}(\beta), \tag{3}$$

where $\alpha, \beta \in \{EF_B(x) : B \subseteq A \ \&\ x \in U\}$ and $Sem_{\mathcal{A}}(\alpha)$ and $Sem_{\mathcal{A}}(\beta)$ are sets of objects from $\mathcal{A}$ satisfying α and β, respectively. The noncrisp inclusion, known in KDD [1] for association rules, is defined by two thresholds t and t':

$$support_{\mathcal{A}}(\alpha, \beta) = card\,[Sem_{\mathcal{A}}(\alpha \wedge \beta)] \geq t, \tag{4}$$

$$accuracy_{\mathcal{A}}(\alpha, \beta) = \frac{support_{\mathcal{A}}(\alpha, \beta)}{card\,[Sem_{\mathcal{A}}(\alpha)]} \geq t'. \tag{5}$$

Elementary granule inclusion in a given information system $\mathcal{A}$ can be defined using different schemes, e.g., by

$$\nu^{\mathcal{A}}_{t,t'}(\alpha, \beta) \text{ if and only if } support_{\mathcal{A}}(\alpha, \beta) \geq t \quad \text{and} \quad accuracy_{\mathcal{A}}(\alpha, \beta) \geq t' \tag{6}$$

or

$$\nu^{\mathcal{A}}_{t}(\alpha, \beta) \quad \text{if and only if} \quad accuracy_{\mathcal{A}}(\alpha, \beta) \geq t. \tag{7}$$

The closeness of granules can be defined by

$$cl^{\mathcal{A}}_{t,t'}(\alpha,\beta) \text{ if and only if } \nu^{\mathcal{A}}_{t,t'}(\alpha,\beta) \text{ and } \nu^{\mathcal{A}}_{t,t'}(\beta,\alpha) \text{ hold.} \tag{8}$$

Decision rules as granules. One can define inclusion and closeness of granules corresponding to rules of the form **if** α **then** β by using accuracy coefficients. Having such granules $g=(\alpha,\beta)$, $g'=(\alpha',\beta')$, one can define the inclusion and closeness of g and g' by

$$\nu^{\mathcal{A}}_{t,t'}\left(g,g'\right) \text{ if and only if } \nu^{\mathcal{A}}_{t,t'}\left(\alpha,\alpha'\right) \text{ and } \nu^{\mathcal{A}}_{t,t'}\left(\beta,\beta'\right). \tag{9}$$

Closeness can be defined by

$$cl^{\mathcal{A}}_{t,t'}\left(g,g'\right) \text{ if and only if } \nu^{\mathcal{A}}_{t,t'}\left(g,g'\right) \text{ and } \nu^{\mathcal{A}}_{t,t'}\left(g',g\right). \tag{10}$$

Another way of defining the inclusion of granules corresponding to decision rules is as

$$\begin{aligned} &\nu^{\mathcal{A}}_{t}\left((\alpha,\beta),\left(\alpha',\beta'\right)\right) \text{ if and only if} \\ &\quad \nu^{\mathcal{A}}_{t_1,t_2}\left(\alpha,\alpha'\right) \text{ and } \nu^{\mathcal{A}}_{t_1,t_2}\left(\beta,\beta'\right) \text{ and } t=w_1\cdot t_1+w_2\cdot t_2, \end{aligned} \tag{11}$$

where w_1,w_2 are some given weights satisfying $w_1+w_2=1$ and $w_1,w_2\geq 0$.

Extensions of elementary granules by tolerance relation. For extensions of elementary granules defined by similarity (tolerance) relation, i.e., granules of the form (α,τ), (β,τ), one can consider the following inclusion measure:

$$\begin{aligned} &\nu^{\mathcal{A}}_{t,t'}((\alpha,\tau),(\beta,\tau)) \text{ if and only if} \\ &\quad \nu^{\mathcal{A}}_{t,t'}\left(\alpha',\beta'\right) \text{ for any } \alpha',\beta' \text{ such that } \left(\alpha,\alpha'\right)\in\tau \text{ and } \left(\beta,\beta'\right)\in\tau \end{aligned} \tag{12}$$

and the following closeness measure:

$$cl^{\mathcal{A}}_{t,t'}((\alpha,\tau),(\beta,\tau)) \text{ if and only if } \nu^{\mathcal{A}}_{t,t'}((\alpha,\tau),(\beta,\tau)) \text{ and } \nu^{\mathcal{A}}_{t,t'}((\beta,\tau),(\alpha,\tau)).$$

It can be important for some applications to define the closeness of an elementary granule α and the granule (α,τ). The definition reflecting an intuition that α should be a representation of (α,τ) sufficiently close to this granule is the following:

$$cl^{\mathcal{A}}_{t,t'}(\alpha,(\alpha,\tau)) \text{ if and only if } cl^{\mathcal{A}}_{t,t'}(\alpha,\beta) \text{ for any } (\alpha,\beta)\in\tau. \tag{13}$$

Sets of rules. An important problem related to association rules is that the number of such rules generated even from a simple data table can be large. Hence, one should search for methods of aggregating close association rules [5,38]. This can be defined as searching for some close information granules. Let us consider two finite sets *Rule_Set* and *Rule_Set'* of association rules defined by

$$Rule\_Set = \{(\alpha_i, \beta_i) : \ i = 1, \ldots, k\} \text{ and } Rule\_Set' = \left\{(\alpha'_i, \beta'_i) : \ i = 1, \ldots, k'\right\}. \tag{14}$$

One can treat them as higher order information granules. These new granules,

$$Rule\_Set, Rule\_Set', \tag{15}$$

can be treated as close to the degree at least t (in $\mathcal{A}$) if and only if there exists a relation *rel* between sets of rules *Rule_Set* and *Rule_Set'* such that

1. For any $Rule \in Rule\_Set$ there is $Rule' \in Rule\_Set'$ such that $(Rule, Rule') \in rel$ and *Rule* is close to *Rule'* (in $\mathcal{A}$) to the degree at least t.
2. For any $Rule' \in Rule\_Set'$ there is $Rule \in Rule\_Set$ such that $(Rule, Rule') \in rel$ and *Rule* is close to *Rule'* (in $\mathcal{A}$) to the degree at least t.

Another way of defining the closeness of two granules G_1, G_2 represented by sets of rules can be described as follows: Let us consider again two granules *Rule_Set* and *Rule_Set'* corresponding to two decision algorithms. We denote by $I(\beta'_i)$ the set $\{j : cl_p^{\mathcal{A}}\left(\beta'_j, \beta'_i\right)\}$ for any $i = 1, \ldots, k'$.

Now, we assume $\nu_p^{\mathcal{A}}(Rule\_Set, Rule\_Set')$ if and only if for any $i \in \{1, \ldots, k'\}$ there exists a set $J \subseteq \{1, \ldots, k\}$ such that

$$cl_p^{\mathcal{A}}\left(\bigvee_{j \in I(\beta'_i)} \beta'_j, \bigvee_{j \in J} \beta_j\right) \text{ and } cl_p^{\mathcal{A}}\left(\bigvee_{j \in I(\beta'_i)} \alpha'_j, \bigvee_{j \in J} \alpha_j\right), \tag{16}$$

and for closeness, we assume

$$cl_p^{\mathcal{A}}\left(Rule\_Set, Rule\_Set'\right) \text{ if and only if } \nu_p^{\mathcal{A}}\left(Rule\_Set, Rule\_Set'\right) \text{ and } \nu_p^{\mathcal{A}}\left(Rule\_Set', Rule\_Set\right). \tag{17}$$

For example, if the granule G_1 consists of rules: **if** α_1 **then** $d = 1$, **if** α_2 **then** $d = 1$, **if** α_3 **then** $d = 1$, **if** β_1 **then** $d = 0$, **if** β_2 **then** $d = 0$ and the granule G_2 consists of rules: **if** γ_1 **then** $d = 1$, **if** γ_2 **then** $d = 0$, then $cl_p(G_1, G_2)$ if and only if

$$cl_p(\alpha_1 \vee \alpha_2 \vee \alpha_3, \gamma_1) \quad \text{and} \quad cl_p(\beta_1 \vee \beta_2, \gamma_2).$$

One can consider a searching problem for a granule *Rule_Set'* of minimal size such that *Rule_Set* and *Rule_Set'* are close.

Certainly, one would like to provide closeness of rules on the extension on the universe U of objects used for inducing rules. Hence, rule grouping is not so simple as in the above example. Rule grouping should be tuned to preserve such constraints.

Granules defined by sets of granules. The previously discussed methods of inclusion and closeness definition can be easily adopted for granules defined by sets of already defined granules. Let G,H be sets of granules.
The inclusion of G in H can be defined by

$$\nu_{t,t'}^{\mathcal{A}}(G,H) \text{ if and only if for any } g \in G \text{ there is } h \in H \text{ for which } \nu_{t,t'}^{\mathcal{A}}(g,h) \tag{18}$$

and the closeness by

$$cl_{t,t'}^{\mathcal{A}}(G,H) \text{ if and only if } \nu_{t,t'}^{\mathcal{A}}(G,H) \text{ and } \nu_{t,t'}^{\mathcal{A}}(H,G). \tag{19}$$

Let G be a set of granules, and let φ be a property of sets of granules from G [e.g., $\varphi(X)$ if and only if X is a tolerance class of a given tolerance $\tau \subseteq G \times G$.] Then, $P_\varphi(G) = \{X \subseteq G : \varphi(X) \text{ holds}\}$. Closeness of granules $X,Y \in P_\varphi(G)$ can be defined by

$$cl_t(X,Y) \text{ if and only if } cl_t\left(g,g'\right) \text{ for any } g \in G \text{ and } g' \in H. \tag{20}$$

We have the following examples of inclusion and closeness propagation rules:

$$\frac{\text{for any } \alpha \in G \text{ there is } \alpha' \in H \text{ such that } \nu_p(\alpha,\alpha')}{\nu_p(G,H)}, \tag{21}$$

$$\frac{cl_p(\alpha,\alpha'), cl_p(\beta,\beta')}{cl_p\left[(\alpha,\beta),(\alpha',\beta')\right]}, \tag{22}$$

$$\frac{\text{for any } \alpha' \in \tau(\alpha) \text{ there is } \beta' \in \tau(\beta) \text{ such that } \nu_p(\alpha',\beta')}{\nu_p\left[(\alpha,\tau),(\beta,\tau)\right]}, \tag{23}$$

$$\frac{cl_p(g,g'), cl_p(h,h')}{cl_p\left[(g,h),(g',h')\right]}, \tag{24}$$

where $\alpha,\alpha',\beta,\beta'$ are elementary granules and g,h,g',h' are finite sets of elementary granules.

One can also present other cases for measuring the inclusion and closeness of granules in the form of inference rules. The exemplary rules have a general form, i.e., they are true in any $\mathcal{A}$ (under the chosen definition of inclusion and closeness). Some of them are derivable from others. We will see in the next part of the chapter that there are also some operations of new granule construction specific for a given information granule system. In this case, one should extract the specific inference rules from existing data.

Information granules defined by inclusion and closeness measures. Let us observe that inclusion (closeness) measures can be used to define new granules that are approximations or generalizations of existing ones. Assume that g,h are given information granules and ν_p is the inclusion measure (where $p \in [0,1]$). A (h,p)-approximation of g is an information granule $Apr(\nu,h,p)$ represented by a set $\{h' : \nu_1(h',h) \wedge \nu_p(h',g)\}$. Now, the lower and upper approximations of given information granules can be easily defined [33] (see Sect. 11.3).

4.4 Target Information Granule Systems: Approximation Spaces

One of an interesting class of information granules consists of *classifiers* (see also Sect. 6.1 in Chap. 25). One can observe that sets of decision rules generated from a given decision table $DT = (U, A, d)$ can be interpreted as information granules. First, one can construct granules G_j corresponding to each particular decision $j = 1, \ldots, r$ by taking a collection $\{g_{ij} : i = 1, \ldots, k_j\}$ of left-hand sides of decision rules for a given decision, where k_j is the number of decision rules for decision j. Next, one can construct from them a collection $G = \{G_1, \ldots, G_r\}$ represented by one granule. Let E be a set of elementary granules over $\mathcal{A} = (U, A)$. We can now consider a granule denoted by $H(e, G)$ for any $e \in E$ that is a collection of coefficients ε_{ij}, where $\varepsilon_{ij} = 1$ if $Sem_{\mathcal{A}}(e) \subseteq Sem_{\mathcal{A}}(g_{ij})$ and 0, otherwise. Hence, the coefficient ε_{ij} is equal to 1 if and only if granule e matches granule g_{ij} in $\mathcal{A}$. Denote now by $Conflict\_res$ a function (resolving conflict between decision rules recognizing elementary granules) defined on granules of the form $H(e, G)$ with values in the set of possible decisions $1, \ldots, r$. Then, $Conflict\_res[H(e, G)]$ is equal to the decision predicted by the classifier $Conflict\_res[H(\bullet, G)]$ on the input granule e.

Hence, one can see that classifiers can be treated as special cases of granules. The parameters to be tuned are voting strategies, matching strategies of objects against rules, as well as other parameters discussed above, such as the closeness of classifier granule in a target granule.

Note that classifiers are information granules in relevant information granule systems. These information granule systems can be treated as approximation spaces. They are targets in computations on information granule systems. The relevant approximation spaces for a given task are information granule systems in which relevant information granules for concept approximation can be selected.

Let us look more deeply into the structure of approximation spaces in the framework of information granule systems. Such information granule systems satisfy the following conditions related to their information granules, relational structure, and semantics:

1. Semantics consists of two parts, namely, relational structure R and its extension R^*.
2. Different types of information granules can be identified: (a) object granules (denoted by x), (b) neighborhood granules (denoted by n with subscripts), (c) pattern granules (denoted by *pat*), and (d) decision class granules (denoted by c).
3. There are decision class granules $c_1, \ldots, c_r$ with semantics in R^* defined by a partition of object granules into r decision classes. However, only the restrictions of these collections on the object granules from R are given.
4. For any object granule x, there is a uniquely defined neighborhood granule n_x.

5. For any class granule c, there is constructed a collection granule $\{(pat,p) : \nu_p^R(pat,c)\}$ of pattern granules labeled by maximal degrees to which *pat* is included in c (in R).
6. For any neighborhood granule n_x, there is distinguished a collection granule $\{(pat,p) : \nu_p^R(n_x,pat)\}$ of pattern granules labeled by maximal degrees to which n_x is at least included in *pat* (in R).
7. There is a class of $Classifier$ functions transforming collection granules (corresponding to a given object x) described in the two previous steps into the power set of $\{1,\ldots,r\}$. One can assume that object granules are the only arguments of $Classifier$ functions if other arguments are fixed.

The classification problem is to find a $Classifier$ function defining a partition of object granules in R^* as close as possible to the partition defined by decision classes. Any such $Classifier$ defines the lower and the upper approximations of any family of decision classes $\{c_i\}_{i\in I}$, where I is a nonempty subset of $\{1,\ldots,r\}$, by

$$\underline{Classifier}(\{c_i\}_{i\in I}) = \{x \in \bigcup_{i\in I} c_i : \emptyset \neq Classifier(x) \subseteq I\}, \tag{25}$$

$$\overline{Classifier}(\{c_i\}_{i\in I}) = \{x \in U^* : Classifier(x) \cap I \neq \emptyset\}. \tag{26}$$

The positive region of the $Classifier$ is defined by

$$POS\,(Classifier) = \underline{Classifier}(\{c_1\}) \cup \ldots \cup \underline{Classifier}(\{c_r\}). \tag{27}$$

The closeness of the partition defined by the constructed $Classifier$ and the partition in R^* defined by decision classes can be measured, e.g., by using the ratio of the positive region size of the $Classifier$ to the size of the object universe. The *quality of* the $Classifier$ can be defined by taking into account, as usual, only objects from $U^* - U$:

$$Quality(Classifier) = \frac{|POS\,(Classifier) \cap (U^* - U)|}{|(U^* - U)|}. \tag{28}$$

One can observe that approximation spaces have many parameters to be tuned to construct the approximation of high-quality class granules .

5 Examples of Operations on Information Granules

New granules can be generated by operations. One can distinguish several classes of operations on information granules [35]. In this section, we discuss some important examples of such operations.

5.1 Generalization

Generalization operations are very important in inducing concept description in machine learning, pattern recognition, knowledge discovery, and data mining applications [15]. A general form of such operations can be defined using similarity, inclusion, or closeness relations. One can define the generalization of an information granule g relative to a given family of information granules G and measure ν by

$$Gen(g,G,\nu) = Make\_granule(\{h \in G : \nu(g,h)\}) \tag{29}$$

Some special cases can be obtained assuming $\nu = \nu_p$ or $\nu = cl_p$, i.e., assuming that ν is an inclusion degree relation to the degree at least p or a closeness relation to the degree at least p. In the former case, one can obtain a cluster of information granules sufficiently covering a given information granule g. In the latter case, we obtain a cluster of information granules close to a given information granule g as a generalization of g. The operation *Make_granule* constructs a new granule from a collection of granules. In the simplest case, when granules in the collections are object sets, this operation can be defined as the set theoretical union.

Let us observe that this general definition can be applied, e.g., to an information granule representing a classifier over a set of elementary granules E and to its generalization to a superset of E. One can treat the classifier $Classifier$ across a set E of elementary granules as a collection of pairs $[e, Classifier(e)]$. Such a classifier can be extended to a new classifier, $Classifier^*$, over a superset E^* of E. The inclusion degree of $Classifier^*$ in the concept being approximated can be expressed as a ratio of the number of the same pairs in both classifiers for elementary granules from $E^* - E$ to the number of all elementary granules in $E^* - E$. The operation *Make_granule* can be interpreted as a fusion operation of different extensions of the classifier, $Classifier$, to some supersets of the initial set of elementary granules E.

5.2 Operations Defined by Decision Tables

Operations on information granules are often (partially) specified by means of data tables. Let us discuss such a specification in more detail. We assume that any (partial) operation $f : G_1 \times \ldots \times G_k \to H$ with arguments from the sets $G_1, \ldots, G_k$ of information granules and values in the set H of information granules is partially specified by a data table (information system [19]). Any row of the data table corresponds to an object that is a tuple $[g_1, \ldots, g_k, f(g_1, \ldots, g_k)]$, where $(g_1, \ldots, g_k)$ belongs to the domain of f. The attribute values for a given object consist of

1. Values of attributes from sets $A_{G_1}, \ldots, A_{G_k}$ on information granules $g_1, \ldots, g_k$ (attributes are extracted from some preassumed feature languages $L_1, \ldots, L_k$).
2. Values of attributes characterizing relations among information granules $g_1, \ldots,$ g_k specifying constraints under which the tuple $(g_1, \ldots, g_k)$ belongs to (a relevant part of) the domain of f.
3. Values of attributes selected for the information granule $f(g_1, \ldots, g_k)$ description.

In this way, partial information about the function f is given. In our considerations, we assume that objects indiscernible by condition attributes are indiscernible by decision attribute, i.e., the decision table $DT = (U, A, d)$ considered is consistent [19]. We also assume, that the representation is consistent with a given function

on information granules, i.e., any image obtained by f of the Cartesian product of indiscernibility classes defined by condition attributes is included in a decision indiscernibility class.
Now, we explain in what sense the decision table $DT = (U,A,d)$ can be treated as partial information about the function $f : G_1 \times \ldots \times G_k \to H$. For $i = 1,\ldots,k$, let

$$G_i^{DT} = \{g_i \in G_i : \text{ there exists in } DT \text{ an object } (g_1,\ldots,g_i,\ldots,g_k,h)\}. \tag{30}$$

One can define H^{DT} in an analogous way. The decision table DT defines a function

$$f_{DT} : G_1/IND(A_{G_1}) \times \ldots \times G_k/IND(A_{G_k}) \to H^{DT}/IND(d)$$

by

$$f_{DT}([g_1]_{IND(A_{G_1})},\ldots,[g_k]_{IND(A_{G_k})}) = [h]_{IND(d)} \text{ if and only if } (g_1,\ldots,g_k,h) \text{ is an object of } DT, \tag{31}$$

where $[g_i]_{IND(A_{G_i})}$ denotes the A_{G_i}-indiscernibility class defined by g_i for $i = 1,\ldots,k$. We assume that a consistency modeling condition for f is satisfied, namely,

$$f([g_1]_{IND(A_{G_1})} \times \ldots \times [g_k]_{IND(A_{G_k})}) = f_{DT}([g_1]_{IND(A_{G_1})},\ldots,[g_k]_{IND(A_{G_k})}) \tag{32}$$

for any $(g_1,\ldots,g_k) \in G_1^{DT} \times \ldots \times G_k^{DT}$ where the left-hand side in (32) denotes the image obtained by f of the set $[g_1]_{IND(A_{G_1})} \times \ldots \times [g_k]_{IND(A_{G_k})}$. The function description can be induced from such a data table by interpreting it as a decision table with the decision corresponding to the attributes specifying the values of the function f.

Certainly, the induced description should be experimentally verified and can be achieved by searching for relevant parameters of data tables, such as relevant features (attributes) and decomposition methods for information granules.

6 Granulation of Relational Structures

In this section, we discuss the information granulation process in a logical framework. Such a process returns information granules of different kinds.

Let us fix some basic notation [7]. Among the basic concepts that will be used are relational structure M of a given signature Sig with a domain Dom and a language L of signature Sig.

There is one more very important concept for information granulation, namely, *neighborhood function*, i.e., any function

$$\mathcal{N} : Dom \longrightarrow Pow^{\omega}(Dom), \tag{33}$$

where

- $Pow^{\omega}(Dom) = \bigcup_{k\in\omega} Pow^{k}(Dom)$.
- $Pow^{1}(Dom) = Pow(Dom)$ and $Pow^{k+1}(Dom) = Pow\left[Pow^{k}(Dom)\right]$ for any non-negative integer k.[1]

To explain this concept, let us consider an information system $\mathcal{A} = (U,A)$ as an example of a relational structure. A neighborhood function $\mathcal{N}_{\mathcal{A}}$ of $\mathcal{A} = (U,A)$ is defined by $\mathcal{N}_{\mathcal{A}}(x) = [x]_A$ for $x \in U = Dom$ where $[x]_A$ denotes the A-indiscernibility class of x (see Chaps. 1 and 8). Hence, the neighborhood function forms basic granules of knowledge about the universe corresponding to objects.

One can approximate sets by means of such neighborhoods (see Chap. 1). It is possible to consider a more general case by taking as the indiscernibility relation a similarity relation between objects instead of an equivalence relation. Such a similarity relation can be defined on objects by means of a similarity of vectors of attribute values on the objects. In this case, the neighborhood function value for a given object x is equal to the similarity class of x consisting of all objects similar to x. We show an example of a neighborhood function returning values more complex than elements of $Pow(Dom)$. Let us consider a case when the neighborhood function values are from $Pow^{2}(Dom)$. Assume that together with an information system, $\mathcal{A} = (U,A)$ is also given a similarity relation τ defined on vectors of attribute values. This relation can be extended to objects. An object $y \in U$ is similar to a given object $x \in U$ if the attribute value vector on x is τ-similar to the attribute value vector on y. Now consider a neighborhood function defined by $\mathcal{N}_{\mathcal{A},\tau}(x) = \{[y]_A : x\tau y\}$.

One might ask why we need such detailed information about the neighborhood function value as in the last case, i.e., a family of indiscernibility classes. One choice will be to take the union of these indiscernibility classes. However, more detailed information is often needed to define some basic relations on information granules, namely, inclusion and closeness of (generated and target) information granules, or, using such a family of elementary information granules, to define some relevant patterns. It can be important for some applications to know about information granules created by a family of indiscernibility classes not only a degree to which its union is included in the target concept but also degrees into which fusions of some subfamilies[2] of such classes are included in a given target concept.

Let us consider more examples. They are related to the granulation of relational structure M by neighborhood functions. We would like to show that, due to the relational structure granulation, we obtain new information granules of more complex structure and in consequence, more general neighborhood functions than those discussed above. Hence, basic granules of knowledge about the universe corresponding to objects can have more complex structures.

---

[1] The power set of X is denoted by $Pow(X)$.

[2] For example, a family of neighborhoods included in the target information granules (concept) to a satisfactory degree.

Assume that a relational structure M and a neighborhood function $\mathcal{N}$ are given. Let us assume at the beginning that the domain of $f_{\mathcal{N}}$ is equal to $Pow(Dom)$. The aim is to define a new relational structure $M_{\mathcal{N}}$ called the *$\mathcal{N}$-granulation* of M. This is done by granulating all components of M by means of $\mathcal{N}$.

We restrict our considerations to two examples. Let us consider first a binary relation $r \subseteq Dom \times Dom$. There are numerous possibilities to define a relation $r_{\mathcal{N}}$ from r. The choice depends on applications. Let us list some possible definitions of $r_{\mathcal{N}}$:

$$
\begin{aligned}
&r_{\mathcal{N}}[\mathcal{N}(x),\mathcal{N}(y)] \text{ iff } \mathcal{N}(x)\times\mathcal{N}(y)\subseteq r, \qquad (34)\\
&r_{\mathcal{N}}[\mathcal{N}(x),\mathcal{N}(y)] \text{ iff } [\mathcal{N}(x)\times\mathcal{N}(y)]\cap r\neq\emptyset,\\
&r_{\mathcal{N}}[\mathcal{N}(x),\mathcal{N}(y)] \text{ iff } card\left\{[\mathcal{N}(x)\times\mathcal{N}(y)]\cap r\right\}\geq s\cdot card\,[\mathcal{N}(x)],\\
&\text{where } s\in[0,1] \text{ is a threshold.}
\end{aligned}
$$

In this way some patterns for pairs of objects are created. Such patterns can be used to approximate a target concept (or concept on an intermediate level) over objects composed of pairs (x,y). Certainly, to induce high-quality approximations it is necessary to search for relevant patterns for concept approximation. This problem is discussed in Sect. 7.

Now, let us consider a function f from M and some possible $\mathcal{N}$-granulations $f_{\mathcal{N}}$ of $\mathcal{N}$. [3]

$$
\begin{aligned}
&f_{\mathcal{N}}[\mathcal{N}(x),\mathcal{N}(y)] = \{\mathcal{N}(z) : z = f(x',y') \text{ for some } x'\in\mathcal{N}(x), y'\in\mathcal{N}(y)\}, \qquad (35)\\
&f_{\mathcal{N}}[\mathcal{N}(x),\mathcal{N}(y)] = \bigcup_{x'\in\mathcal{N}(x),y'\in\mathcal{N}(y)} \{\mathcal{N}(z) : z = f(x',y')\},\\
&f_{\mathcal{N}}[\mathcal{N}(x),\mathcal{N}(y)] = \bigcup_{x'\in\mathcal{N}(x),y'\in\mathcal{N}(y)} \{\mathcal{N}(z) : card\,(\mathcal{N}(z))\geq s \text{ and } z = f(x',y')\},
\end{aligned}
$$

where $s\in[0,1]$ is a threshold.

One can consider the values of $f_{\mathcal{N}}$ as generators of patterns used for the target concept approximation. An example of pattern language can be obtained by considering the results of set theoretical operations on neighborhoods.

Observe that in the first example the values of function $f_{\mathcal{N}}$ are in $Pow^2(Dom)$. Hence, one could extend the neighborhood function and the relation granulation on this more complex domain. Certainly, this process can be continued, and more complex patterns can be generated. On the other hand, it is also necessary to bound the depth of exploring $Pow^{\omega}(Dom)$. This can be done by using the rough set approach. For example, after generating patterns from $Pow^2(Dom)$, one should, in a sense, reduce them to $Pow(Dom)$ by considering some operations from $Pow^2(Dom)$

[3] We assume that f has two arguments.

into $Pow(Dom)$ returning the relevant patterns for the target concept approximation. Such a reduction is necessary, especially if the target concepts are elements of the family $Pow(Dom)$.

We have discussed relations of inclusion and closeness in this chapter. They can also be used for granulation of relational structures. Here, it is worth mentioning that such relations between information granules must be considered not only due to the incomplete information about objects. Many real-life problems can often be expressed as searching for construction of information granules (from some elementary ones) satisfying a given specification (which can also be treated as information granule) to a satisfactory degree. This allows making the construction (reasoning) process more efficient and robust with respect to deviation of parameters, e.g., related to input information granules.

7 Rough Set Approach to Inductive Reasoning

In this section, we would like to explain in more detail computations on information granule systems aiming at constructing relevant approximation spaces (see Sect. 4.4).

One can consider the rough set approximations of decision classes in decision systems (see Chap. 1). Such approximations can be used to obtain models of decision classes. However, in inductive reasoning, we would like to approximate concepts over universe of objects, say U^∞, wider than the universe U of objects in a given decision system. In other words, assuming $U \subset U^\infty$, we would like to approximate concepts over U^∞ that are extensions of decision classes in a given decision system. In this section, we present a general searching scheme for approximation spaces relevant to approximation of such concepts and show how to induce in them *classifiers* approximating those concepts. This is a basic scheme for machine learning, pattern recognition, data mining, and knowledge discovery [11,15].

The main observation is that, in the case considered, it is also necessary to induce a relevant approximation space. Such a space is usually different from the partition defined by the conditional attributes of a given decision system. It consists of some subsets of U^∞, called neighborhoods of objects. It should be emphasized that neighborhoods usually create a covering of U^∞, not necessarily a partition. They are defined by *patterns* chosen from some relevant *pattern languages*. In practical applications, it is often necessary to specify a given model using its particular description in a pattern language. Different descriptions may have substantially different properties outside of U, e.g., patterns for disjoint decision classes can have different nonempty intersections in $U^* - U$. To indicate that a given model is specified by a particular description, we use the term *description model*.

The structure of the pattern languages and the patterns themselves should be discovered. The whole process is quite complex and is illustrated in Fig. 2, where

- $\mathcal{A} = (U, A, d)$ denotes a decision system.[4]
- $\mathcal{A}_{\text{train}}$ and $\mathcal{A}_{\text{test}}$ are training and testing subsystems of $\mathcal{A}$, respectively.
- $\mathcal{L} = \{L_i\}_{i \in I}$ is a family of pattern languages.
- $\mathcal{Q} = \{Q_j\}_{j \in J}$ is a family of quality measures for description models.
- M is a description model covering objects in U.
- C is a classifier obtained from M covering (almost) the whole universe U^∞.

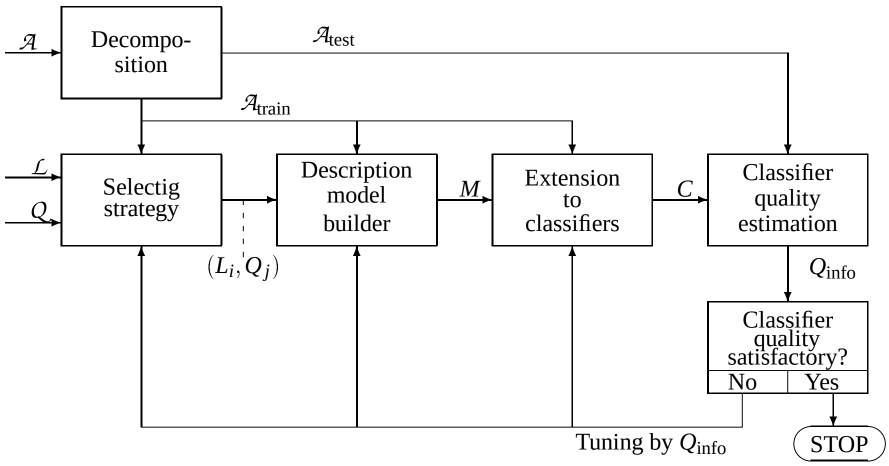

Fig. 2. Approximation space and classifier construction using rough sets.

Elements of L_i are formulas called *patterns*. Patterns, in a given decision system, define sets of objects in which they are satisfied. Description models describe the decision classes of $\mathcal{A}$ by using patterns from L_i and some inclusion measures of those patterns in decision classes. Description models can be built by means of, e.g., decision rules over descriptors from L_i. As a typical example, one can consider the language of patterns consisting of conjunctions of descriptors over a selected set of attributes. More complex pattern language can include conjunctions of formulas that are disjunctions of descriptor conjunctions.

Quality measures can be used as criteria for tuning the model. For given L_i and Q_j, one can search for a description model using patterns from L_i which is (sub-) optimal w.r.t. the measure Q_j. However, the goal is to induce the relevant description model for the induced classifier, covering the whole universe of objects.

[4] For simplicity in reasoning, we assume that $\mathcal{A}$ does not change in time.

This, in particular, requires tuning parameters of the description quality measure. There are many ways to specify quality measures. For example, a measure Q_j, can be specified using the *minimum description length principle* (see Chap. 25), where one estimates the quality of approximation as well as the size of the description model defined. The minimum description length principle requires choosing a description of the smallest size from those of the same approximation quality. In this case, the quality measure depends on two arguments. The first, represents the quality of approximation (e.g., using the positive region of decision classes or entropy measure). The second represents the values of some measures based on the model size. A proper balance between these two arguments is generally obtained by using training data. Tuning may involve thresholds for degrees of inclusion of patterns from L_i in decision classes or for the positive region size. The use of the notion of inclusion to a satisfactory degree allows one to reduce the size of the positive region description, compared to descriptions based on crisp inclusion.

The whole process presented in Fig. 2 can be viewed as a searching process for a relevant approximation space. As we have mentioned before, such an approximation space consists of neighborhoods of objects from U as well as inclusion relations making it possible to measure degrees of inclusion (or closeness) of such neighborhoods in other information granules.

The induced description model should be extended to a classifier of all objects from the whole universe of objects U^{∞}, not only from U. [5] Recall that, for any object to be classified, it is necessary to compute its degree of inclusion in any pattern from the description model. For new objects (outside of U), these degrees can suggest conflicting decisions and, together with the degrees of pattern inclusion in decision classes, create input for conflict resolution strategy necessary to compute the classifier output.

Next, the induced classifier is tested on objects from $\mathcal{A}_{\text{test}}$. Information Q_{info} about the classifier behavior quality is returned from the classifier quality estimation module. If Q_{info} shows that the classifier quality is unsatisfactory, it is used to tune parameters in different modules presented in Fig. 2 and to reconstruct the classifier to a new one with better quality. In addition, matching strategies for objects and patterns, as well as parameters for conflict resolution strategy, can also be tuned. The parameters involved in the tuning process can, for instance, be inclusion degree thresholds, parameters characterizing approximation quality, or parameters measuring the description model size.

The approximation spaces discussed are examples of information granules. They can be obtained as results of complex computations on information granule systems.

[5] In Sect. 4.4 and Chap. 25, we discuss a classifier structure.

8 AR schemes and Rough Neural Networks

AR schemes are the basic constructs used in RNC. Such schemes can be derived from parameterized productions representing robust dependencies on data. Algorithmic methods for extracting such productions from data are discussed in [24,31], [34]. The left-hand side of each *production* (see Fig. 3) is (in the simplest case) of the form

$$\left(st_1(ag),(\varepsilon_1^{(1)},\ldots,\varepsilon_r^{(1)})\right),\ldots,\left(st_k(ag),(\varepsilon_1^{(k)},\ldots,\varepsilon_r^{(k)})\right) \tag{36}$$

and the right-hand side is of the form $[st(ag),(\varepsilon_1,\ldots,\varepsilon_r)]$ for some positive integers k,r.

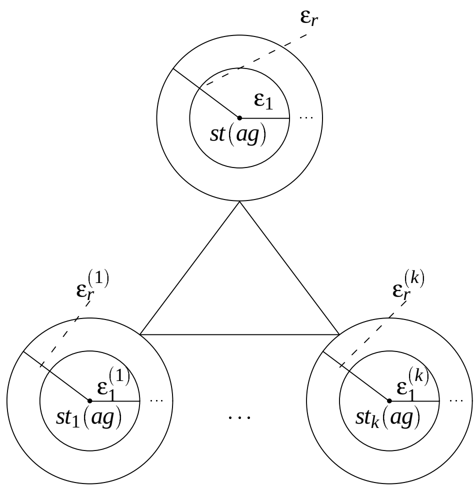

Fig. 3. Parameterized production

Such a production represents information about an operation o that can be performed by the agent ag. In a production, k denotes the arity of operation. The operation o represented by the production transforms standard (prototype) input information granules $st_1(ag),\ldots,st_k(ag)$ into the standard (prototype) information granule $st(ag)$. Moreover, if input information granules $g_1,\ldots,g_k$ are close to

$$st_1(ag),\ldots,st_k(ag)$$

to degrees at least $\varepsilon_j^{(1)},\ldots,\varepsilon_j^{(k)}$, then the result of operation o on information granules $g_1,\ldots,g_k$ is close to the standard $st(ag)$ to a degree at least ε_j where $1\le j\le k$. Standard (prototype) granules can be interpreted in different ways. In particular, they can correspond to concept names in natural language.

The productions described above are basic components of a reasoning system over an agent set *Ag*. An important property of such productions is that they are expected to be discovered from available experimental data and background knowledge. Let us also observe that the degree structure is not necessarily restricted to reals from the interval $[0,1]$. The inclusion degrees can have a structure of complex information granules used to represent the degree of inclusion. It is worthwhile mentioning that the productions can also be interpreted as constructive descriptions of some operations on fuzzy sets. The methods for such constructive description are based on rough sets and Boolean reasoning [12,19].

AR schemes can be treated as derivations obtained by using productions from different agents. The relevant derivations generating AR schemes satisfy a so-called *robustness* (or *stability*) condition. This means that at any node of derivation the inclusion (or closeness) degree of a constructed granule to the prototype (standard) granule is higher than that required by the production to which the result should be sent (see Fig. 4). This makes it possible to obtain a sufficient robustness condition for all derivations. For details, refer to [26,27].

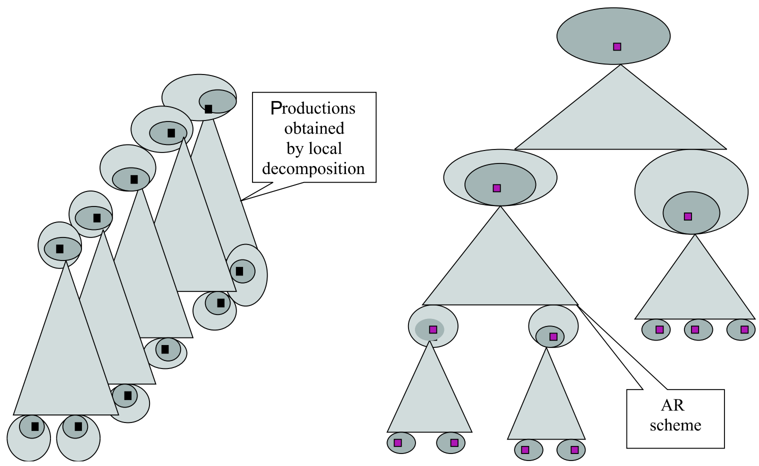

Fig. 4. Productions and AR schemes

AR schemes are discovered from data and background knowledge by using symbolic reasoning. After that they can be used as in connectionist approach. If sensor measurements are sufficiently close to inputs of AR schemes, then with high probability, one can predict that an analyzed object (situation) belongs to a target concept (the concept of a *dangerous* situation on a road considered earlier). Hence, symbolic and connectionist approaches can work as complementary not competitive approaches.

When standards are interpreted as concept names in natural language and a reasoning scheme in natural language over the standard concepts is given, the corresponding AR scheme represents a cluster of reasoning (constructions) approximately following (by means of other information granule systems) the reasoning in natural language. In the following section, we discuss different approaches to standard information granule definition.

8.1 Standards Represented by Rough Sets

In the simplest case, standards can be represented by lower approximations of concepts. The degree of inclusion of a pattern supported by objects from the set $X \subseteq U$ in the lower approximation supported by the objects from the set $Y \subseteq U$ can be measured by the ratio $|X \cap Y|/|X|$ where U is the set of objects in a given decision table (representing the training sample).

However, if the lower approximation is intended to describe the concept in an extension of the training sample U, then inductive reasoning should be used to find an approximation of this lower approximation of the concept. Such approximations can be represented, e.g., by decision rules describing the lower approximation and its complement together with a method making it possible to measure matching degrees of new objects and the decision rules as well as the method for conflict resolution between decision rules voting for the new objects. In such cases, the degree of inclusion of any pattern in the lower approximation has a more complex structure and can be represented by two vectors of inclusion degrees of this pattern in decision rules representing the lower approximation and its complement, respectively.

Using the rough set approach, one can measure not only the degree of inclusion of a concept in the lower approximation but also the degree of inclusion of a concept in other information granules defined using the rough set approach, such as upper approximations, boundary regions, or complements of upper approximations of concepts. In this case, instead of one degree, one should consider a vector of degrees. However, if the lower approximation is too small, then such a lower approximation cannot be treated as a standard of good quality and it may be necessary to consider other kinds of standards that can be constructed using, e.g., a rough-fuzzy approach or classifier construction methods.

8.2 Standards Corresponding to Rough-Fuzzy Sets

The approach presented in the previous section can be extended to concepts defined by fuzzy sets. We will show that the dependencies between linguistic variables can be modeled by productions. Using the rough-fuzzy approach, one can search for dependencies between lower approximations of differences between relevant cuts of fuzzy sets modeling linguistic variables. The productions built along such dependencies make it possible to model dependencies between linguistic variables.

Moreover, the approximate reasoning on linguistic variables can be modeled by approximate reasoning schemes derived from productions.

We are now going to describe rough-fuzzy granules. We assume that if X is an information granule, e.g., a set of objects, then its upper and lower approximations with respect to any subset of attributes in a given information system or decision table are information granules, too. Let us see now how such information granules can be used to define fuzzy concept [41] approximations in a constructive way.

Let $DT = (U, A, d)$ be a decision table where the decision d is the fuzzy membership function ν restriction to the objects from U. Consider reals $0 < c_1 < \ldots < c_k$ where $c_i \in (0,1]$ for $i = 1, \ldots, k$. Any c_i defines c_i-cut by $X_i = \{x \in U : \nu(x) \geq c_i\}$. Assume that $X_0 = U$ and $X_{k+1} = X_{k+2} = \emptyset$. A *rough-fuzzy granule* (*rf-granule*, for short) corresponding to $(DT, c_1, \ldots, c_k)$ is any granule $g = (g_0, \ldots, g_k)$ such that for some $B \subseteq A$,

$$\begin{aligned} &Sem_B(g_i) = \left[\underline{B}(X_i - X_{i+1}), \overline{B}(X_i - X_{i+1})\right], \text{ for } i = 0, \ldots, k, \text{ and} \\ &\overline{B}(X_i - X_{i+1}) \subseteq (X_{i-1} - X_{i+2}), \text{ for } i = 1, \ldots, k, \end{aligned} \tag{37}$$

where $\underline{B}$ and $\overline{B}$ denote the B-lower and B-upper approximation operators, respectively [19], and $Sem_B(g_i)$ denotes the semantics of g_i.

Any function $\nu^* : U \rightarrow [0,1]$ satisfying the conditions

$$\begin{aligned} &\nu^*(x) = 0, \text{ for } x \in U - \overline{B}X_1, \\ &\nu^*(x) = 1, \text{ for } x \in \underline{B}X_k, \\ &\nu^*(x) = c_{i-1}, \text{ for } x \in \underline{B}(X_{i-1} - X_i), \text{ and } i = 2, \ldots, k-1, \\ &c_{i-1} < \nu^*(x) < c_i, \text{ for } x \in (\overline{B}X_i - \underline{B}X_i), \text{ where } i = 1, \ldots, k, \text{ and } c_0 = 0, \end{aligned} \tag{38}$$

is called a B-approximation of ν.

Assume that a rule *if* α *and* β *then* γ is given, where α, β, γ are linguistic variables. The aim is to develop a searching method for rough-fuzzy granules g^1, g^2, g^3 approximating, to satisfactory degrees, α, β, γ, respectively, and at the same time, making it possible to discover association rules of the form *if* α' *and* β' *then* γ' with sufficiently large support and confidence coefficients, where α', β', γ' are some components (e.g., the lower approximations of differences between cuts of fuzzy concepts corresponding to linguistic variables) of granules g^1, g^2, g^3 (modeling linguistic variables), respectively. Searching for such patterns and rules is a complex process with many parameters to be tuned. For given linguistic rules, the relevant cuts for fuzzy concepts corresponding to them should be discovered. Next, the relevant features (attributes) should be chosen. They are used to construct approximations of differences between cuts. Moreover, relevant measures should be chosen to measure the degree of inclusion of object patterns in the lower approximations constructed. One can expect that these measures are parameterized and that the relevant parameters should be discovered in the process of searching for productions.

Certainly, in searching for relevant parameters in this complex optimization process, evolutionary techniques can be used. The quality of discovered rules can be measured as a degree to which discovered rule *if* α' *and* β' *then* γ' approximates the linguistic rule *if* α *and* β *then* γ. This can be expressed by such parameters as degrees of inclusion of patterns α', β', γ' in α, β, γ, their supports, etc.

Let us observe that for a given linguistic rule, it will be necessary to find a family of rules represented by discovered patterns which together create an information granule sufficiently close to a modeled linguistic rule. One can also search for more general information granules representing clusters of discovered rules

$$if\ \alpha'\ and\ \beta'\ then\ \gamma'$$

approximating the linguistic rule

$$if\ \alpha\ and\ \beta\ then\ \gamma.$$

These clustered rules can be of higher quality. Certainly, this makes it necessary to discover and tune many parameters relevant to measuring the similarity or closeness of rules.

The problem discussed is of great importance in classifying situations by autonomous systems on the basis of sensor measurements [40]. Moreover, this is one of the basic problems to be investigated for hybridization of rough and fuzzy approaches.

8.3 Standards Corresponding to Classifiers

For classifiers, we obtain another possibility. Let us consider information granules corresponding to values of terms $Match(e, \{G_1, \ldots, G_k\})$ for $e \in E$ (see also Sect. 6.1 in Chap. 25 and [31]), where E is a set of elementary granules and $G_1, \ldots,$ G_k are granules corresponding to patterns described by the left-hand sides of decision rules for k decision classes. Any such granule defines a probability distribution on a set of possible decisions (extended by the value corresponding to *no decision predicted*). The probability for each such value is obtained simply as a ratio of all votes for the decision value determined by this information granule and the number of objects. Some probability distributions can be chosen as standards. this means that instead of the lower approximations, one can use such probability distributions. Certainly, it can sometimes be useful to choose not one such standard but a collection of them. Now, one should decide how to measure the distances between probability distributions. Using a chosen distance measure, e.g., Euclidean or a more advanced one developed in statistics, it is possible to measure the degree of closeness of classified objects e, e' using the probability distributions corresponding to them. The next steps in constructing an approximate reasoning rule based on classifiers is analogous to the discussed before.

One of the most interesting cases occurs when standards are interpreted as concepts from natural language. In this case, measures of inclusion and closeness can be based on semantic similarity and closeness relations rather than on statistical properties. Constructing such measures is a challenge. This case is strongly related to the CW paradigm. The productions discovered can, to a satisfactory degree, be consistent with reasoning steps performed in natural language.

9 Rough Neural Networks

Rough neural networks are schemes for information granule construction in a distributed environment. They perform computations on information granules representing concepts rather than on numbers. These information granules can be constructed, e.g., using a rough set approach, rough-fuzzy approach, or an approach based on classifiers.

Rough neural networks have several types of parameters to be tuned. Let us list some of them assuming that we use a rough set approach for concept approximations:

- parameters of approximation spaces used to approximate concepts transformed by operations performed by agents.
- parameters of approximation spaces located in interfaces between agents used to approximate, by one agent, concepts communicated by some other agents.
- degrees of inclusion and closeness of patterns constructed along the network to concept approximations in the network (see Fig. 5).
- parameters of approximation spaces for pattern approximations.

Let us observe that for uncertainty rules illustrated in Fig. 5, the approximation spaces related to agents ag, ag_1, ag_2 are also involved in discovery from data of function f, called a mereological connective [24]. Rough mereological connectives make it possible to propagate uncertainty coefficients. If objects (or patterns) x_1, x_2 delivered by agents ag_1, ag_2 are close to standards $st(ag_1), st(ag_2)$ to degrees at least $\varepsilon_1, \varepsilon_2$, then the object (pattern) x constructed by operation O from x_1, x_2 is close to $st(ag)$ to degree at least $f(\varepsilon_1, \varepsilon_2)$. Hence, if $f(\varepsilon_1, \varepsilon_2) \geq \varepsilon$, where ε is a given admissible uncertainty threshold for *ag*, then an estimate given by a rough mereological connective is sufficient.

From the above considerations, it follows that a rough neural network is a complex structure with different sorts of parameters to tune. The goal of tuning these parameters is to learn structures that can compute relevant information granules with acceptable accuracy.

We would like to comment on approximation spaces creating interfaces between communicating agents. These parameterized approximation spaces can be treated

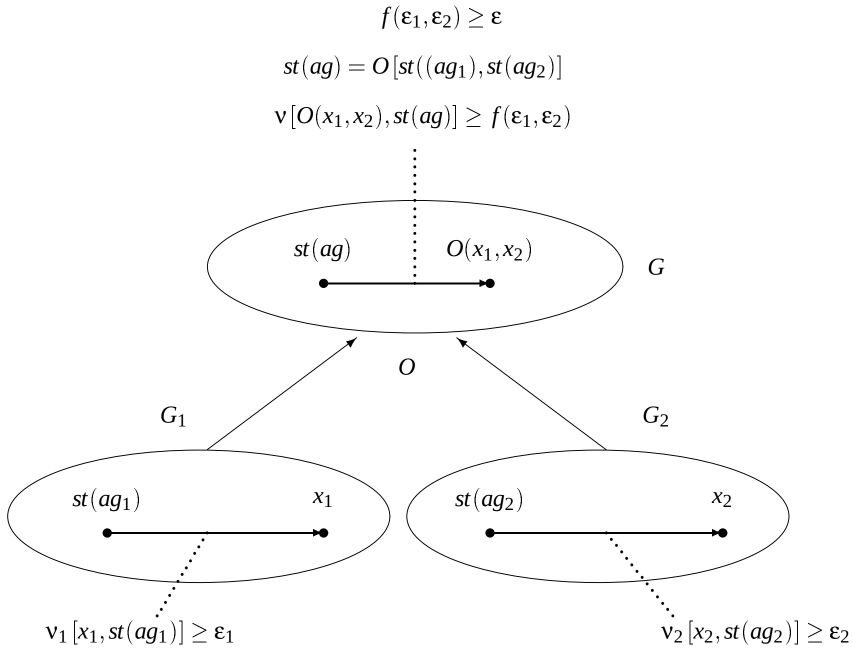

Fig. 5. Uncertainty rules

as analogous to neural network weights. These parameters should be learned to induce the relevant information granules.

Extending AR schemes for synthesizing information granules by adding interfaces represented by approximation spaces and used for approximating information granules exchanged between agents, we obtain granule construction schemes that can be treated as generalizations of neural network models. The main idea is that granules sent by one agent to another are not, in general, exactly understandable by the receiving agent because these agents are using different languages and usually there is no translation (from the sender language to the receiver language) preserving the exact semantic meaning of formulas. Hence, it is necessary to construct interfaces that will make it possible to approximately understand received granules. These interfaces can be, in the simplest case, constructed on the basis of information exchanged by agents and stored in the form of decision data tables. From such tables, the approximations of concepts can be constructed using a rough set approach. In general, it is a complex process because a high-quality approximation of concepts can often be obtained only in dialogue (involving negotiations, conflict resolutions, and cooperation) among agents. In this process, the approximation can be constructed gradually when the dialogue is progressing. In our model, we assume that for any n-ary operation $o(ag)$ of agent ag, there are approximation spaces $AS_1[o(ag), in], \ldots, AS_n[o(ag), in]$ that will filter (approximate) the granules received by the agent for performing operation $o(ag)$. In turn, the granule sent by an agent after performing the operation is filtered (approximated) by the approximation space $AS[o(ag), out]$. These approximation spaces are parameterized. The parameters are used to optimize the size of neighborhoods in these spaces as well as the inclusion relation. A granule approximation quality is taken as the optimization criterion. Approximation spaces attached to any operation of ag correspond to neuron weights in neural networks, whereas operation performed by an agent ag on information granules corresponds to an operation realized on vectors of real numbers by a neuron (see Fig. 1 in Chap. 2). A parameterized approximation space can be treated as an analogy to a neural network weight. In this figure, $w_1, \ldots, w_n, f$ denote weights, aggregation operator, and activation function of a classical neuron, respectively, whereas

$$AS_1(P), \ldots, AS_k(P)$$

denote parameterized approximations spaces where agents process input granules $G_1, \ldots, G_k$ and O denotes an operation (usually parameterized) that produces the output of a granular network. The parameters P of approximation spaces should be learned to induce the relevant information granules.

We call extended schemes for complex object construction *rough neural networks*. The problem of deriving such schemes is closely related to perception [43]. The stability of such networks corresponds to the resistance to noise of classical neural networks.

Let us observe that in our approach deductive systems are substituted by production systems of agents linked by approximation spaces, communication strategies, and mechanism for deriving AR schemes. This revision of classical logical notions seems to be important for solving complex problems in distributed environments.

10 Extracting AR Schemes from Data and Background Knowledge

In this section, we present some methods of information granule decomposition aimed at extracting decomposition rules from data. We restrict our considerations to methods based only on experimental data. This approach can be extended to information granule decomposition methods using background knowledge [36].

The search methods discussed in this section return local granule decomposition schemes. These local schemes can be composed by using techniques discussed in the previous section. The schemes of granule construction received (which can also be treated as approximate reasoning schemes) also have the following property: if input granules are sufficiently close to input concepts, then the output granule is sufficiently included in the target concept, provided this property is preserved locally [27].

The above may be formulated in terms of a synthesis grammar [28] with productions corresponding to local decomposition rules. The relevant derivations over a given synthesis grammar represent approximate reasoning schemes. Note that synthesis grammars reflect processes in multiagent systems in which agents are involved in cooperation, negotiation, and conflict-resolving actions when attempting to provide a solution to the specification of a problem. Complexities of membership problems for languages generated by synthesis grammars may be taken *ex definitione* as complexities of the underlying synthesis processes.

We show that in some cases decomposition can be performed using methods for specific rule generation based on Boolean reasoning [12]. Moreover, we present the way the decomposition, stable with respect to information granule deviations, can be obtained.

First, let us start from some general remarks. Information granule decomposition methods are important components of methods for inducing AR schemes from data and background knowledge. Such methods are used to extract local decomposition schemes, called productions, from data [26]. The AR schemes are constructed by means of productions.

Decomposition methods are based on searching for the parts of information granules that can be used to construct relevant, higher level patterns that match, to a satisfactory degree, the target granule.

One can distinguish two kinds of parts (represented, e.g., by subformulas or subterms) of AR schemes. Parts of the first type are represented by expressions from a language, called the *domestic* language L_{d}, that has known semantics (consider, for example, semantics defined in a given information system [19]). Parts of the second type of AR scheme are from a language, called *foreign* language L_{f} (e.g., natural language), that has semantics definable only in an approximate way (e.g., by patterns extracted using rough, fuzzy, rough-fuzzy or other approaches). For example, the parts of the second kind of scheme can be interpreted as soft properties of sensor measurements [4].

For a given expression e, representing a given scheme that consists of subexpressions from L_{f} first it is necessary to search for relevant approximations in L_{d} of the foreign parts from L_{f} and next to derive global patterns from the whole expression after replacing the foreign parts by their approximations. This can be a multilevel process, i.e., we face problems of pattern propagation discovered through several domestic-foreign layers.

Productions from which AR schemes are built can be induced from data and background knowledge by pattern extraction strategies. Let us consider some such strategies. The first makes it possible to search for relevant approximations of parts by using the rough set approach. This means that each part from L_{f} can be replaced by its lower or upper approximation with respect to a set B of attributes. The approximation is constructed on the basis of a relevant data table [12,19]. With the second strategy, parts from L_{f} are partitioned into a number of subparts corresponding to cuts (or the set theoretical differences between cuts) of fuzzy sets representing vague concepts, and each subpart is approximated by rough set methods. The third strategy is based on searching for patterns sufficiently included in foreign parts. In all cases, the extracted approximations replace foreign parts in the scheme, and candidates for global patterns are derived from the scheme obtained after the replacement. Searching for relevant global patterns is a complex task because many parameters should be tuned, e.g., the set of relevant features used in approximation, relevant approximation operators, the number and distribution of objects from the universe of objects among different cuts, and so on. One can use evolutionary techniques [13] in searching for semioptimal patterns in the decomposition.

It has been shown that decomposition strategies can be based on rough set methods developed for decision rule generation in combination with the Boolean approach [16,23,24,35]. In particular, methods for decomposition based on background knowledge can be developed [31,34,36].

Now we can turn to some details of granule decomposition methods. We assume that a family of inclusion relations $\nu_p^i \subseteq G_i \times G_i$, $\nu_p^H \subseteq H \times H$ and a family of closeness relations $cl_p^1, \ldots, cl_p^k, cl_p^H$ for every $p \in [0,1]$ and $i = 1, \ldots, k$ are given [27]. Let us assume that two thresholds t, p are given. We define a relation

$$Q_{t,p}^{DT}(Pattern_1, \ldots, Pattern_k, \overline{v})$$

between granules called patterns $Pattern_1, \ldots, Pattern_k$ for arguments of f and the target pattern $\overline{v}$ representing the decision value vector in the following way:

$$\begin{aligned} &Q_{t,p}^{DT}(Pattern_1, \ldots, Pattern_k, \overline{v}) \text{ if and only if} \qquad (39)\\ &\quad \nu_p^H \left\{ f\left[Sem_{DT}(Pattern_1) \times \ldots \times Sem_{DT}(Pattern_k)\right], [\overline{v}]_{IND(d)} \right\} \text{ and}\\ &\quad card\left[Sem_{DT}(Pattern_1) \times \ldots \times Sem_{DT}(Pattern_k)\right] \geq t. \end{aligned}$$

Let us now consider the following decomposition problem:

Granule decomposition problem
Input:

- Two thresholds t, p.
- A decision table $DT = (U, A, d)$ representing an operation $f : G_1 \times \ldots \times G_k \rightarrow H$ where $G_1, \ldots, G_k$ and H are given finite sets of information granules.

- A fixed decision value vector $\overline{v}$ represented by a value vector of decision attributes.

Output:

- A tuple $(Pattern_1, \ldots, Pattern_k)$ of patterns such that

$$Q_{t,p}^{DT}(Pattern_1, \ldots, Pattern_k, \overline{v}).$$

We consider a description given by decision rules extracted from the data table specifying the function f. Any left-hand side of a decision rule can be divided into parts corresponding to different arguments of the function f. The ith part, denoted by $Pattern_i$, specifies a condition that should be satisfied by the ith argument of f to obtain the function value specified by the decision attributes. For simplicity, we do not consider conditions specifying the relations between arguments. In this way, the left-hand sides of decision rules describe patterns, $Pattern_i$. The semantics of extracted patterns relevant for the target can be defined as the image with respect to f of the Cartesian product of sets $Sem_{DT}(Pattern_i)$, i.e., by $f[Sem_{DT}(Pattern_1) \times \ldots \times Sem_{DT}(Pattern_k)]$ (see Fig. 6). One can use one of the methods for decision rule generation, e.g., for generating of minimal rules or their approximations (e.g., in the form of association rules) [12] to obtain such decision rules.

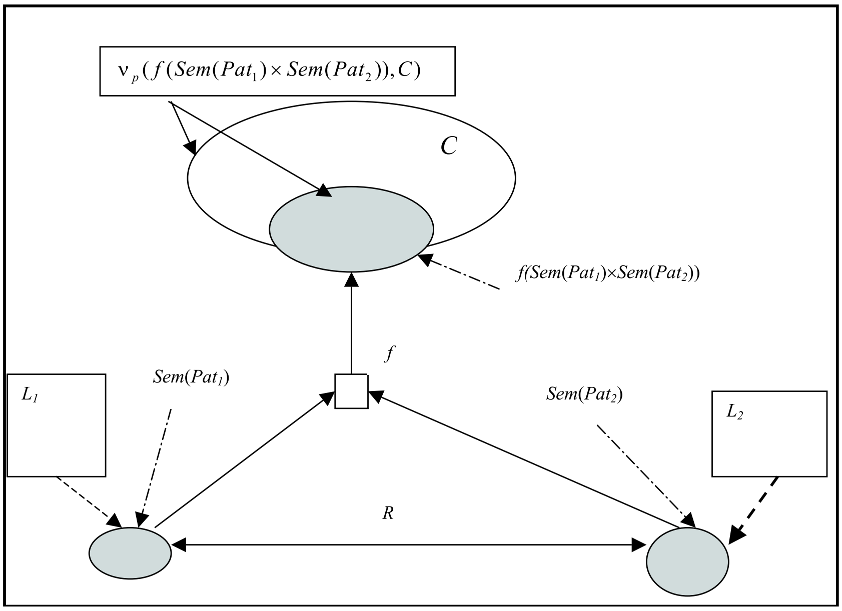

Fig. 6. Decomposition of information granule

In the former case, we receive the most general patterns for function arguments consistent with a given decision table, i.e., the information granules constructed by means of the function f from patterns extracted for arguments are included exactly in the information granule represented by a given decision value vector in the data table. In the latter case, we obtain more general patterns for function arguments having the following property: information granules constructed by means of f from such patterns will be included to a satisfactory degree in the information granule represented by a given decision value vector in the data table.

One of the very important properties of the operations on information granules discussed above is their robustness with respect to the deviations of arguments (see, e.g., [28]). This property can be formulated as follows: if an information granule constructed by means of f from the extracted patterns, $Pattern_1, \ldots, Pattern_k$, satisfies the target condition, then the information granule constructed from patterns, $Pattern'_1, \ldots, Pattern'_k$, sufficiently close to $Pattern_1, \ldots, Pattern_k$, respectively, satisfies the target condition, too. In this way, we obtain the following problem:

Robust decomposition problem (RD problem)
Input:

- Thresholds t, p.
- A decision table $DT = (U, A, d)$ representing an operation $f : G_1 \times \ldots \times G_k \to H$ where $G_1, \ldots, G_k$ and H are given finite sets of information granules.
- A fixed decision value vector $\overline{v}$ represented by a value vector of decision attributes.

Output:

- A tuple $(p_1, \ldots, p_k)$ of parameters.
- A tuple $(Pattern_1, \ldots, Pattern_k)$ of patterns such that

$$Q^{DT}_{t,p}(Pattern'_1, \ldots, Pattern'_k, \overline{v}),$$

if $cl^i_{p_i}(Sem_{DT}(Pattern_i), Sem_{DT}(Pattern'_i))$ for $i = 1, \ldots, k$.

It is possible to search for the solution of the RD problem by modifying the previous approach of decision rule generation. In the process of rule generation, one can impose a stronger discernibility condition by assuming that objects are discernible if their tolerance classes are disjoint. Certainly, one can tune parameters of tolerance relations to obtain rules of satisfactory quality. We would like to stress that efficient heuristics for solving these problems can be based on Boolean reasoning [12].

Searching for relevant patterns for information granule decomposition can be based on methods for tuning parameters of rough set approximations of fuzzy cuts or concepts defined by differences between cuts (see Sect. 8). In this case, pattern languages consist of parameterized expressions describing the rough set approximations of *parts* of fuzzy concepts as fuzzy cuts or differences between cuts. Hence, an interesting research direction related to the development of new hybrid rough-fuzzy methods arises aiming at developing algorithmic methods for rough set approximations of such parts of fuzzy sets relevant to information granule decomposition. An approach presented in this section can be extended to local granule decomposition based on background knowledge [36].

11 Basic Concepts Definable by Means of Rough Inclusions

In this section, we present several examples of basic notions for rough sets and granular computing definable by means of rough inclusion relation. The examples illustrate an important role of inclusion relations in rough set theory and granular computing.

11.1 Indiscernibility and Discernibility

Let us start from the fundamental notion of rough set theory, namely, indiscernibility of objects in a given information system. This notion, in granular computing, should be generalized to arbitrary information granules. Let us observe that indiscernibility is defined relative to a given information system that is, as shown, a special kind of information granule. We generalize the indiscernibility relation to arbitrary information granules and define it relative to given information granule and degree inclusion. Any information granules g_1, g_2 are indiscernible relative to a given information granule h and a degree p if for any exact part h' of h, granule g_1 is included to a degree at least p in h' if and only if g_2 is included to a degree at least p in h'.

More formally, granules g_1, g_2, h, p are given information granules. Granules g_1, g_2 are *h-indiscernible* (see Fig. 7) to a degree at least p, in symbols $g_1 IND_h^p g_2$, if and only if

$$\forall h' \left\{ \nu_1(h', h) \Rightarrow \left[\nu_p(g_1, h') \Leftrightarrow \nu_p(g_2, h') \right] \right\} . \tag{40}$$

Certainly, there are some other possibilities for introducing the indiscernibility of granules. For example, one could consider, instead of the rough inclusion relation, the closeness of granules to a degree (see Sect. 4.3) or only some parts of h, e.g., maximal proper parts of h.

When inclusion relations are set inclusions, $p = 1$, h is an information system, g_1, g_2 are objects in the information system and as exact parts of h are considered indiscernibility classes [19] of the information system, we obtain the definition of the indiscernibility considered in rough set theory.

Let us consider one more example of indiscernibility for decision rules. Intuitively, two such rules are indiscernible if they are matched by the same objects and they

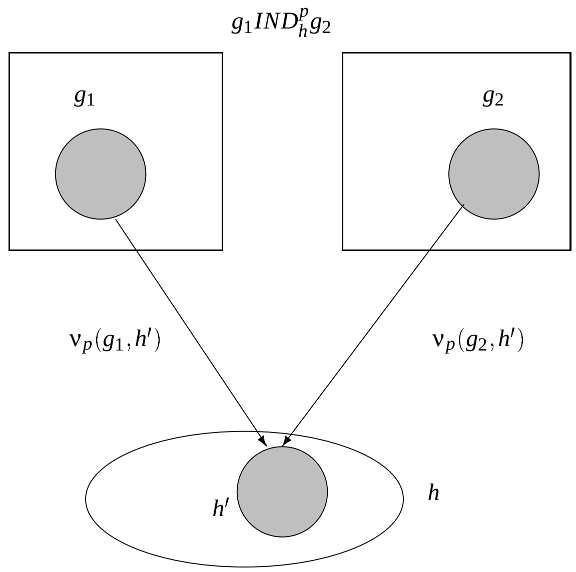

Fig. 7. Indiscernibility of information granules

predict the same decision. The indiscernibility of such information granules can be defined relative to a given information granule as a set of attribute value vectors corresponding to objects. Two decision rules r_1, r_2 are indiscernible relative to such a granule to a degree at least p if and only if any attribute value vector matches (i.e., is included to degree p) the left-hand side of rule r_1 if and only if it matches the left-hand side of rule r_2.

One direct method defining *discernibility* of information granules relative to a given information granule is to define such a relation as a complement of the indiscernibility relation, i.e.,

$$g_1 DIS_h^p g_2 \text{ if and only if } g_1 IND_h^p g_2 \text{ does not hold.} \tag{41}$$

Hence, two objects g_1, g_2 are discernible relative to h to a degree at least p if there exists a part h' of h discerning them, i.e., such that conditions $\nu_p(g_1, h')$ and $\nu_p(g_2, h')$ are not equivalent. Certainly, this is only the simplest case of discernibility. Discernibility can also be defined by not taking the complement of indiscernibility.

11.2 Semantics of Information Granules

Let us observe that *semantics of information granules* can be defined relative to preassumed inclusion relations. Let us consider an example illustrated in Fig. 8. For given information granules g,h, we define h-semantics of g, as a granule equal to the result of *Make_granule* operation on the collection of granules h' included to degree at least p in g.

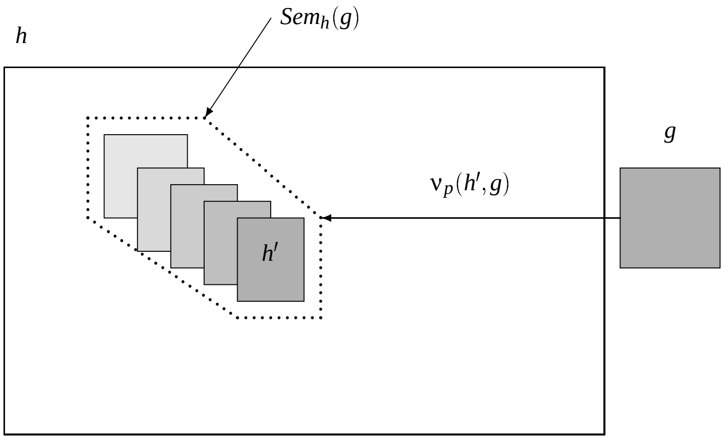

Fig. 8. Semantics of granule g defined relative to h

11.3 Approximation Spaces

Using rough inclusions, one can generalize the approximation operations for sets of objects, known in rough set theory, to arbitrary information granules. The idea is to consider a family $G = \{g_t\}_t$ of granules by means of which a given granule g should be approximated. We assume that for a given set $\{g_1, \ldots, g_k\}$ of information granules included to a degree at least p in g, there is a granule $Make\_granule(\{g_1, \ldots, g_k\})$ included to a degree at least $f(p)$ in g, representing in a sense a collection $\{g_1, \ldots, g_k\}$ where f is a function transforming inclusion degrees into inclusion degrees. A typical example of *Make_granule* is the set theoretical union used in rough set theory. Let us recall that inclusion degrees are partially ordered by a relation $\leq$.

Assume that p is an inclusion degree, $G = \{g_t\}_t$ is a given family of information granules, and g is a granule from a given information granule system S. The (G, p)-*approximation* of g, in symbols $APP_{G,p}(g)$, is an information granule defined by

$$Make\_granule(\{g_t : \nu_p(g_t, g)\}). \quad (42)$$

Now, assuming that, $p < q$, one can consider two approximations for a given information granule g by G. The (G, q)-*lower approximation* of g is defined by

$$LOW_{G,p,q}(g) = APP_{G,q}(g) \text{ (see Fig. 9).} \quad (43)$$

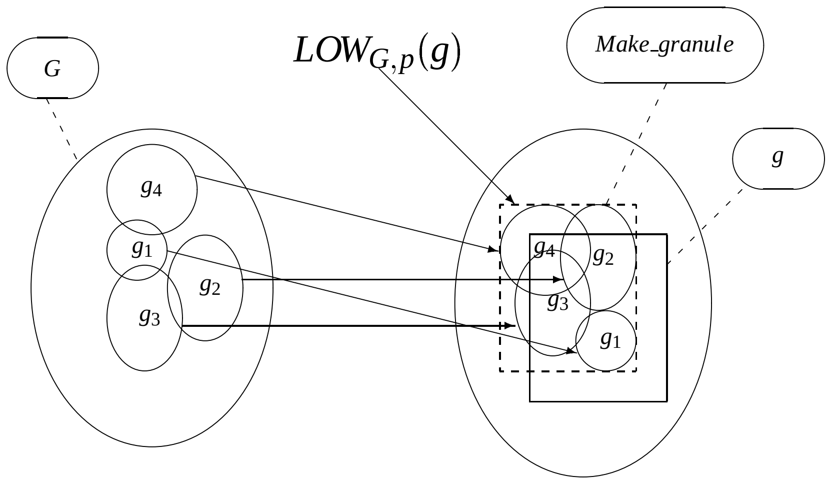

Fig. 9. (G,p)-lower approximation of granule g

The (G,q)-*upper approximation* of g is defined by

$$UPP_{G,p,q}(g) = Make\_granule(\{APP_{G,p'}(g) : p' > p\}). \quad (44)$$

Let us recall the general definition of an approximation space [33]. A *parameterized approximation space* is a system $AS_{\#,\$} = (U, I_{\#}, \nu_{\$})$, where

- U is a nonempty set of objects,
- $I_{\#} : U \to P(U)$, where $P(U)$ denotes the power set of U, is an uncertainty function, and
- $\nu_{\$} : P(U) \times P(U) \to [0,1]$ is a rough inclusion function.

If $p \in [0,1]$, then $\nu_p(X,Y)$ denotes that the condition, $\nu(X,Y) \geq p$, holds. The uncertainty function defines for every object x a set of similarly described objects, i.e., the neighborhood $I_{\#}(x)$ of x. A constructive definition of an uncertainty function can be based on the assumption that some metrics (distances) are given for attribute values.

A set $X \subseteq U$ is *definable in* $AS_{\#,\$}$, if it is a union of some values of the uncertainty function. The rough inclusion function defines the degree of inclusion between two subsets of U [33]. For example, if X is nonempty, then $\nu_p(X,Y)$ if and only if $p \leq \frac{card(X \cap Y)}{card(X)}$. If X is the empty set, we assume that $\nu_1(X,Y)$. For a parameterized approximation space $AS_{\#,\$} = (U, I_{\#}, \nu_{\$})$ and any subset $X \subseteq U$, the lower and the upper approximations are defined by

$$\begin{aligned} &LOW\left(AS_{\#,\$}, X\right) = \{x \in U : \nu_{\$}\left(I_{\#}(x), X\right) = 1\} \text{ and} \\ &UPP\left(AS_{\#,\$}, X\right) = \{x \in U : \nu_{\$}\left(I_{\#}(x), X\right) > 0\}, \text{ respectively.} \end{aligned} \quad (45)$$

One can observe that the above definition of a parameterized approximation space is an example of the introduced notion of information granule approximation. It is enough to assume that G is the set of all neighborhoods $I_{\#}(x)$ for $x \in U$, $g \subseteq U$, and $Make\_granule$ is the set theoretical union, $p = 0$, $q = 1$.

It is useful to define parameterized approximations with parameters tuned in the searching process for approximations of concepts. This idea is crucial for methods of constructing concept approximations.

11.4 Granule Structures

Using rough inclusions, one can define the structures of information granules. The structure of a given information granule can be defined by means of a graph in which (directed) edges represent the relation as a part to a given degree between parts of an information granule. In searching for relevant patterns in data mining problems, one can look for clusters of information granule structures representing collections of granules with similar structures and having a required property, e.g., the majority of objects with such structure are in a given decision class. In the simplest case, two information structures are similar if they have the same graph structure and parts labeling nodes in the structure are sufficiently close. Such clusters of information granules are called *perception structures* (*terms*) (see Fig. 10).

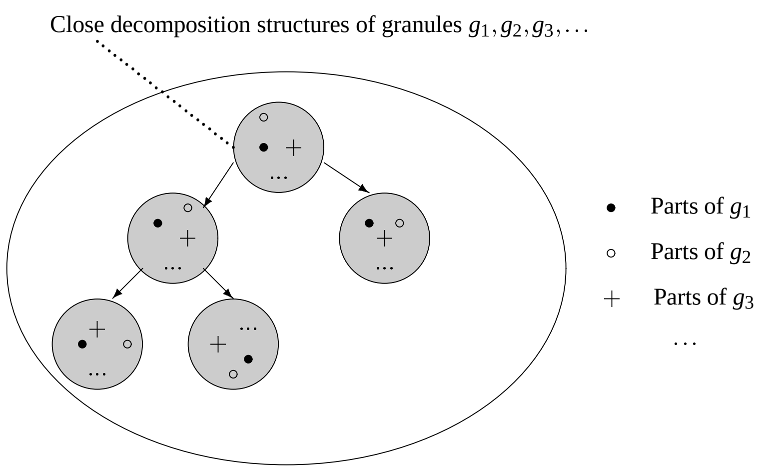

Fig. 10. Perception structure of information granules

12 Conclusions

We have outlined a methodology for approximate reasoning by means of RNC. Several research directions are related to AR schemes and rough neural networks discussed. We enclose a list of such directions together with examples of problems.

1. *Developing foundations for information granule systems.* Certainly, still more work is needed to develop solid foundations for synthesizing and analyzing information granule systems. In particular, methods for constructing hierarchical information granule systems and methods for representing such systems should be developed.
2. *Algorithmic methods for inducing parameterized productions.* Some methods have already been reported, such as the discovery of rough mereological connectives from data [24] and methods based on decomposition [23,31,32,36]. However, these are only initial steps toward algorithmic methods for inducing parameterized productions from data. One interesting problem is to determine how such productions can be extracted from data and background knowledge. A method in this direction has been proposed in [4].
3. *Algorithmic methods for synthesizing of AR schemes.* It was observed [29,32] that problems of negotiation and conflict resolution are of great importance in synthesizing of AR schemes. The problem arises, e.g., when we are searching in a given set of agents for a granule sufficiently included or close to a given one. These agents, often working with different systems of information granules, can derive different granules, and their fusion will be necessary to obtain the relevant output granule. In the fusion process, negotiations and conflict resolutions are necessary. Much more work should be done in this direction by using the existing results on negotiations and conflict resolution. In particular, Boolean reasoning methods seem to be promising [32]. Another problem is related to the size of production sets. These sets can be large, and it is important to develop learning methods for extracting *small* candidate production sets in the process of extending of temporary derivations out of huge production sets. For solving this kind of problem, methods for clustering productions should be developed to reduce the size of production sets. Moreover, dialogue and cooperative strategies between agents can help to reduce the search space for necessary extension of the temporary derivations.
4. *Algorithmic methods for learning in rough neural networks.* A basic problem in rough neural networks is related to selecting relevant approximation spaces and to parameter tuning. One can also look up to what extent existing methods for classical neural methods can be used for learning in rough neural networks. However, it seems that new approaches and methods for learning in rough neural networks should be developed to deal with real-life applications. In particular, it is due to the fact that high-quality approximations of concepts can often be obtained only through dialogue and negotiation processes among agents in which the concept approximation is gradually constructed. Hence, for rough neural networks, learning methods based on dialogue, negotiation and

conflict resolution should be developed. In some cases, one can use rough set and Boolean reasoning methods directly [35]. However, more advanced cases need new methods. In particular, hybrid methods based on rough and fuzzy approaches can bring new results [20].

5. *Fusion methods in rough neural neurons.* A basic problem in rough neurons is fusion of the inputs (information) derived from information granules. This fusion makes it possible to contribute to the construction of new granules. When a granule constructed by a rough neuron consists of characteristic signal values made by relevant sensors, a step in the direction of solving the fusion problem can be found in [21].
6. *Adaptive methods.* Certainly, adaptive methods for discovering productions and for inducing AR schemes and rough neural networks should be developed [13].
7. *Discovery of multiagent systems relevant to given problems.* Quite often, agents and communication methods among them are not given a priori with the problem specification, and the challenge is to develop methods for discovering multiagent system structures relevant to given problems, in particular, methods for discovering relevant communication protocols.
8. *Construction of multiagent systems for complex real-life problems.* Challenging problems are related to applying the methodology presented to real-life problems such as control of autonomous systems (see, e.g., www page of WITAS project [40]), web mining problems [10,31], sensor fusion [3,21,22], and spatial reasoning [6,8].
9. *Evolutionary methods.* For all of the above methods, it is necessary to develop evolutionary searching methods for semioptimal solutions [13].
10. *Parallel algorithms.* The problems discussed are of high computational complexity. Parallel algorithms searching for AR schemes and methods for their hardware implementation are one important research direction.

Acknowledgments

The research of Andrzej Skowron has been supported by the State Committee for Scientific Research of the Republic of Poland (KBN), research grant 8 T11C 025 19 and by the Wallenberg Foundation grant. The research of Jaroslaw Stepaniuk has been supported by the State Committee for Scientific Research of the Republic of Poland (KBN), research grant 8 T11C 025 19.

References

1. R. Agrawal, H. Mannila, R. Srikant, H. Toivonen, A. Verkano. Fast discovery of association rules. In U.M. Fayyad, G. Piatetsky-Shapiro, P. Smyth, R. Uthurusamy, editors. *Advances in Knowledge Discovery and Data Mining*, 307–328, AAAI/MIT Press, Cambridge, MA, 1996.
2. E. Bonabeau, M. Dorigo, G. Theraulaz, editors. *Swarm Intelligence: From Natural to Artificial Systems*. Oxford University Press, Oxford, 1999.

3. R.R. Brooks, S.S. Iyengar, editors. *Multi-Sensor Fusion*. Prentice-Hall, Upper Saddle River, NJ, 1998.
4. P. Doherty, W. Łukaszewicz, A. Skowron, A. Szałas. *Combining Rough and Crisp Knowledge in Deductive Databases* (this book).
5. G. Dong, J. Li. Interestingness of discovered association rules in terms of neighborhood-based unexpectedness. In X. Wu, K. Ramamohanarao, K.B. Korb, editors. *Proceedings of the 2nd Pacific-Asia Conference on Knowledge Discovery in Databases (PAKDD-98)*, LNAI 1384, 72–86, Springer, Berlin, 1998.
6. I. Düntsch, editor. Spatial reasoning (special issue). Vol. 45(1/2) of the *Fundamenta Informaticae*, 2001.
7. H-D. Ebbinghaus, H-D., J. Flum, W. Thomas. *Mathematical Logic*. Springer, Heidelberg, 1994.
8. M.T. Escrig, editor. *Qualitative Spatial Reasoning: Theory and Practice*. IOS Press, Amsterdam, 1998.
9. M.N. Huhns and M.P. Singh, editors. *Readings in Agents*. Morgan Kaufmann, San Mateo, CA, 1998.
10. H. Kargupta and P. Chan, editors. *Advances in Distributed and Parallel Knowledge Discovery*. MIT/AAAI Press, Cambridge, MA, 2001.
11. W. Kloesgen, J, Żytkow, editors. *Handbook of Knowledge Discovery and Data Mining*. Oxford University Press, Oxford, 2002.
12. J. Komorowski, Z. Pawlak, L. Polkowski, A. Skowron. Rough sets: A tutorial. In *[18]*, 3–98, 1999.
13. J.R. Koza. *Genetic Programming II: Automatic Discovery of Reusable Programs*. MIT Press, Cambridge, MA, 1994.
14. Y.Y. Lin, Y.Y. Yao, and L.A. Zadeh, editors. *Data Mining, Rough Sets and Granular Computing*. Physica, Heidelberg, 2002.
15. T.M. Mitchell. *Machine Learning*. McGraw-Hill, Portland, OR, 1997.
16. H.S. Nguyen, S.H. Nguyen, A. Skowron. Decomposition of task specification. LNCS 1609, 310–318, Springer, Berlin, 1999.
17. S.K. Pal, W. Pedrycz, A. Skowron, R. Swiniarski, editors. Rough-neuro computing (special issue). Vol. 36 of *Neurocomputing: An International Journal*, 2001.
18. S.K. Pal, A. Skowron, editors. *Rough-Fuzzy Hybridization: A New Trend in Decision Making*. Springer, Singapore, 1999.
19. Z. Pawlak. *Rough Sets: Theoretical Aspects of Reasoning about Data*. Kluwer, Dordrecht, 1991.
20. Z. Pawlak, A. Skowron. Rough set rudiments. *Bulletin of the International Rough Set Society*, 3/4: 181–185, 1999.
21. J. Peters, A. Skowron, S. Ramanna, M. Borkowski, Z. Suraj. Sensor, filter, and fusion models with rough Petri nets. *Fundamenta Informaticae*, 47: 307–323, 2001.
22. J.F. Peters, S. Ramanna, A. Skowron, J. Stepaniuk, Z. Suraj, M. Borkowski. Sensor fusion: A rough granular approach. In *Proceedings of the Joint 9th International Fuzzy Systems Association (IFSA) World Congress and 20th North American Fuzzy Information Processing Society (NAFIPS) International Conference*, 1367–1371, IEEE Press, Piscataway, NJ, 2001.
23. J.F. Peters, A. Skowron, J. Stepaniuk. Rough granules in spatial reasoning. In *Proceedings of the Joint 9th International Fuzzy Systems Association (IFSA) World Congress and 20th North American Fuzzy Information Processing Society (NAFIPS) International Conference, July 2001*, 1355–1361, IEEE Press, Piscataway, NJ, 2001.

24. L. Polkowski, A. Skowron. Rough mereological approach to knowledge-based distributed AI. In J.K. Lee, J. Liebowitz, and J.M. Chae, editors. *Critical Technology. Proceedings of the Third World Congress on Expert Systems*, 774–781, Cognizant Communications, New York, 1996.
25. L. Polkowski, A. Skowron. Rough mereology: A new paradigm for approximate reasoning. *International Journal of Approximate Reasoning*, 15(4): 333–365, 1996.
26. L. Polkowski, A. Skowron. Grammar systems for distributed synthesis of approximate solutions extracted from experience. In G. Paun, A. Salomaa, editors. *Grammar Models for Multiagent Systems*, 316–333, Gordon and Breach, Amsterdam, 1999.
27. L. Polkowski, A. Skowron. Towards adaptive calculus of granules. In *[44]*, 201–227, 1999.
28. L. Polkowski, A. Skowron. Rough mereological calculi of granules: A rough set approach to computation. *Computational Intelligence*, 17(3): 472–492, 2001.
29. L. Polkowski and A. Skowron. Rough-neuro computing. In W. Ziarko, Y.Y. Yao, editors. *Proceedings of the 2nd International Conference on Rough Sets and Current Trends in Computing (RSCTC 2000)*, LNAI 2005, 57–64, Springer, Berlin, 2001.
30. B.D. Ripley. *Pattern Recognition and Neural Networks*. Cambridge University Press, Cambridge, 1996.
31. A. Skowron. Toward intelligent systems: Calculi of information granules. *Bulletin International Rough Set Society*, 5(1/2): 9–30, 2001.
32. A. Skowron, L. Polkowski. Rough mereological foundations for design, analysis, synthesis, and control in distributive systems. *Information Sciences*, 104(1/2): 129–156, 1998.
33. A. Skowron, J. Stepaniuk. Tolerance approximation spaces. *Fundamenta Informaticae*, 27: 245–253, 1996.
34. A. Skowron, J. Stepaniuk. Information granule decomposition. *Fundamenta Informaticae*, 47(3/4): 337–350, 2001.
35. A. Skowron, J. Stepaniuk. Information granules: Towards foundations of granular computing. *International Journal of Intelligent Systems*, 16(1): 57–86, 2001.
36. A. Skowron, J. Stepaniuk, J.F. Peters. Extracting patterns using information granules. In *Proceedings of the JSAI International Workshop on Rough Set Theory and Granular Computing (RSTGC 2001)*, 135–142, Matsue, Japan, 2001.
37. P. Stone. *Layered Learning in Multi-Agent Systems: A Winning Approach to Robotic Soccer*. MIT Press, Cambridge, MA, 2000.
38. H. Toivonen, M. Klemettinen, P. Ronkainen, K. Hätönen, H. Mannila. Pruning and grouping discovered association rules. In *Proceedings of the MLnet Familiarization Workshop on Statistics, Machine Learning and Knowledge Discovery in Databases*, 47–52, Heraklion, Crete, Greece, 1995.
39. P.P Wang, editor. *Granular Computing*. Wiley, New York, 2001.
40. WITAS. available at http://www.ida.liu.se/ext/witas/eng.html. *Project web page*, 2001.
41. L.A. Zadeh. Fuzzy sets. *Information and Control*, 8: 333–353, 1965.
42. L.A. Zadeh. Fuzzy logic = computing with words. *IEEE Transactions on Fuzzy Systems*, 4: 103–111, 1996.
43. L.A. Zadeh. A new direction in AI: Toward a computational theory of perceptions. *AI Magazine*, 22(1): 73–84, 2001.
44. L.A. Zadeh, J. Kacprzyk, editors. *Computing with Words in Information/Intelligent Systems*, volumes 1, 2. Physica, Heidelberg, 1999.