GRANULAR COMPUTING
A Rough Set Approach

Son H. Nguyen[1] & Andrzej Skowron[1] & Jarosław Stepaniuk[2]
[1]Institute of Mathematics, University of Warsaw,
Banacha 2, 02-097 Warsaw, Poland,
E-mail: {skowron, son}@mimuw.edu.pl
[2]Institute of Computer Science
Bialystok University of Technology
Wiejska 45A, 15-351 Bialystok, Poland,
E-mail: jstepan@ii.pb.bialystok.pl

Abstract

We discuss information granule calculi as a basis of granular computing. They are defined by constructs like information granules, basic relations of inclusion and closeness between information granules as well as operations on them. The exact interpretation between granule languages of different information sources (agents) often does not exist. Hence (rough) inclusion and closeness of granules are considered instead of their equality. Examples of all the basic constructs of information granule calculi are presented. The construction of more complex information granules is described by expressions called terms. We discuss the synthesis problem of robust terms, i.e., descriptions of information granules, satisfying a given specification in a satisfactory degree. We also present a method for synthesis of information granules represented by robust terms (approximate schemes of reasoning) by means of decomposition of specifications for such granules. The discussed problems of granular computing are of special importance for many applications, in particular related to spatial reasoning as well as to knowledge discovery and data mining.

1 Introduction

In many application areas there is a need for special tools helping to construct approximate description of complex concepts. Let us consider some examples:

- In the area of pattern recognition and computer vision efficient methods of approximate reasoning from sensor measurements to conclusion are investigated, e.g., for evaluation of identified situation on the road as safe or not by unmanned aerial vehicle receiving sensor measurements from cameras, information geographical systems, etc. (see e.g. [32]).
- In the area of multi-agent systems [9], [31] researchers are developing methods of approximate reasoning in distributed environment of cooperating and competing agents performing some tasks, like soccer playing [31].
- Information granulation [36], [37], [17] is very relevant for data mining and knowledge discovery in spatio-temporal databases (see e.g. [8], [21], [4], [7], [29], [20]) and for flexible query answering (see e.g., [39]).

The reasoning used to solve problems in the above mentioned areas should be performed on the level allowing to deal with complex objects called information granules used to describe, e.g., concept approximations or complex classifiers. Roughly speaking, information granules can be treated as linked collections (clumps) of objects drawn together by the criteria of indiscernibility, similarity or functionality [36], [37]. Descriptions of some information granules can be induced from sensor measurements. However, they can be not relevant for inducing target information granules (approximations of target concepts). Hence it is necessary to develop calculi on information granules consisting of operations on information granules for generating new information granules. These operations can be of different nature. Some of them can be defined as simple set theoretical operations on information granules, in case of crisp granules. However, when information granules are represented by rough or fuzzy sets these operations should be accordingly modified. Operations can also be defined by experimental data tables. Then it is possible to induce approximate descriptions of these operations from data tables. Yet other operations are related to the necessity of information granule approximation when the granules are exchanged between different sources of information (agents) describing these granules in different languages. Classifiers investigated by machine learning, pattern recognition and data mining communities (see e.g., [13]) can be treated as examples of complex information granules [19]). Operations of fusion of classifiers (or fusion of ensambles of classifiers) are currently intensively studied operations on information granules representing classifiers (see e.g., [6]). It has been recognized that for many real-life problems fusion operation performed on partial classifiers for obtaining the global classifer (solving the target task) can be only constructed via a multi-layered construction (see e.g., [31]). Hence, composition of operations transforming information granules should be taken into account in searching for synthesis methods of complex granules satisfying constrains described by specifications. Lotfi Zadeh [36], [37] pointed out another class of complex information granules for Computing with Words. The aim of this new computing paradigm is to find foundations for computing rather on words representing soft concepts, corresponding to linguistic variables, than on numbers.

We would like to discuss briefly an example showing a motivation for our work [29]. Let us consider a team of agents recognizing the situation on the road. The aim is to classify a given situation as, e.g., *dangerous or not.* This *soft specification granule* is represented by a family of information granules called *case soft patterns* representing soft cases, like *cars are too close.* The whole scene (actual situation on the road) is decomposed into regions perceived by local agents. Higher level agents can reason about regions observed by team of their children agents. They can express in their own languages features used by their children. Moreover, they can use new features like attributes describing relations between regions perceived by children agents. The problem is how to organize agents into a team with the following property:

The information granule representing situations on the road identified by team of agents as dangerous is close in a sufficiently high degree to the soft specification *situation dangerous on the road.*

The aim of our project is to make step toward foundations for this kind of reasoning. In particular it is necessary to define: information granules, soft information granules, closeness of information granules in satisfactory degree, information granules derived by team of agents etc.

In Figure 1 the following entities are depicted:

- a specification soft granule represented by family of case soft patterns g_1, g_2, g_3, g_4;
- input granules ig_1, ig_2 representing actual local situations for ag_1, ag_2;
- higher level granules describing situation received by fusion of granules perceived by ag_1, ag_2 taking into account the relationships between granules and a context in which they appear;
- og, og_1, og_2 granules returned by ag, ag_1, ag_2, respectively; og is received by performing an operation at ag on og_1, og_2 being clumps of local situations similar to ig_1, ig_2, respectively.

Let us observe that the whole construction scheme for identification situation on the road should consists of separate construction (like this one presented in Figure 1) for any case soft pattern.

From the above discussion it follows that building any calculus of information granules one should answer the following basic questions:

- How one can define relevant information granules for a given problem?
- What are the relevant operations on information granules for a given problem?

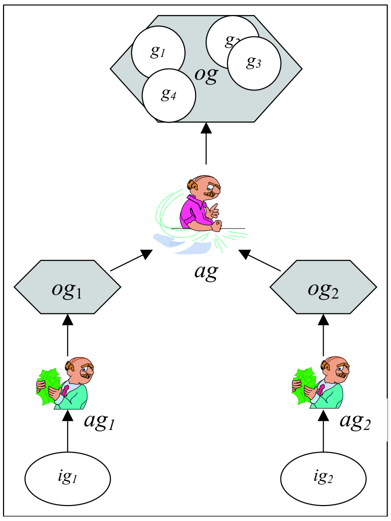

Figure 1: Illustrative Example

- How to construct from given granules information granules satisfying given soft constraints?
- What properties should have the information granule constructions?

Several attempts have been made to develop foundations for computing with words (see, e.g., [36], [37], [38], [39]). Among them there is a rapidly growing area of granular computing aiming to develop computing model based on information granules (see, e.g., in [12], [25], [26], [27], [33], [34], [35]). Calculi of information granules have been investigated in a framework of rough mereology (see e.g. [16], [17], [18], [11]).

In the paper we present partial answers for the above questions. Following [27] we present illustrative examples showing how higher level information granules and related to them constructs can be defined using already constructed lower level information granules and constructs. We emphasize in the paper that the information granules are parameterized what allows to tune them in the process of new granule construction. There are two main kinds of parameters which can be tuned in searching for relevant information granules. The first ones allow to adjust the proper size of granules (a proper degree of generalization) in the process of inductive reasoning, the second ones help to tune degrees of inclusion between information granules in the process of construction of complex information granules.

We introduce parameterized approximation spaces for each unit (agent) involved in granule construction. These approximation spaces are more general than the approximation spaces used for rough set approach in [15]. However, the basic idea of the lower and upper approximation operations can be easily extended to this more general case. The generalized approximation spaces and in particular the lower and upper approximation operations are basic tools in our approach. They can be used to specify how given granules can be approximately described in terms of other granules (or granules sets). In the paper we show an application of rough sets, namely for approximation of concepts exchanged between different units (agents).

Information granules should allow for compressed representation of complex nested clumps of objects (like, e.g., soft patterns) still relevant for construction of granule satisfying a given specification (in a satisfactory degree). In our approach it is realized by means of information granules and constructions allowing to build more complex information granules from simpler ones together with the relevant extensions of inclusion and closeness relations [18].

One more important aspect of granule construction is their robustness with respect to deviations of granule components. We consider here a notion of satisfiability in a degree developed in rough mereological approach [16] to deal with the problem. This enable us to define robust granule constructions. Informally speaking, the robustness of a given construction means that the closeness (inclusion) of constructed granules is preserved in a satisfactory degree under small deviations of input granules (or operation parameters used for the granule construction). The robustness of the target construction can be deduced from robustness of their sub-constructions when some constraints for composition are satisfied (see e.g., [18]).

Finally, we discuss a method for synthesis of complex granules satisfying a given specification by decomposition of specification.

The paper is an extension of results presented in [25], [26], [27]. We adopt some basic ideas developed in rough mereological approach (see e.g., [16], [17], [18], [11]). In our approach a crucial role play the rough set approach [15], [11] especially the approximation operations in generalized approximation spaces [23].

The paper is organized as follows. In Section 2 we recall the definition of generalized approximation space. Examples of constructions of information granules are presented in Section 3. In Section 4 their inclusion and closeness are discussed. Next, in Section 5 the problem of information granule construction in distributed environment is investigated. In Section 6 we discuss the problem of robustness of expressions (terms) specifying granule construction. In Section 7 we investigate the decomposition problems of specifications as a method for synthesis of information granules satisfying a given specification in satisfactory degree.

2 Rough Sets and Approximation Spaces

We recall general definition of approximation space [23], [30].

Definition 1 *A parameterized approximation space is a system* $AS_{\#,\$} = (U, I_{\#}, \nu_{\$})$, *where*

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

We will also write I, ν instead of $I_{\#}, \nu_{\$}$ to simplify notation. If $X = \{x\}$ and $Y = \{y\}$ we also write $I(x)$ and $\nu(x,y)$ instead of $I(X)$ and $\nu(X,Y)$, respectively. If $p \in [0,1]$ then $\nu_p(x,y)$ denotes the following condition $\nu(x,y) \geq p$ holds.

The uncertainty function defines for every object x a set of similarly described objects. A constructive definition of uncertainty function can be based on the assumption that some metrics (distances) are given on attribute values. For example, if for some attribute $a \in A$ a metric $\delta_a : V_a \times V_a \longrightarrow [0, \infty)$ is given, where V_a is the set of all values of attribute a, then one can define the following uncertainty function:

$$y \in I_a^{f_a}(x) \text{ if and only if } \delta_a(a(x), a(y)) \leq f_a(a(x), a(y)),$$

where $f_a : V_a \times V_a \to [0, \infty)$ is a given threshold function.

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 [23].

The lower and the upper approximations of subsets of U are defined as follows.

Definition 2 *For a parameterized approximation space* $AS_{\#,\$} = (U, I_{\#}, \nu_{\$})$ *and any subset* $X \subseteq U$ *the lower and the upper approximations are defined by*

$LOW(AS_{\#,\$}, X) = \{x \in U : \nu_{\$}(I_{\#}(x), X) = 1\}$,

$UPP(AS_{\#,\$}, X) = \{x \in U : \nu_{\$}(I_{\#}(x), X) > 0\}$, *respectively.*

Approximations of concepts (sets) are constructed on the basis of background knowledge. Obviously, concepts are also related to new (unseen) objects. Hence it is very useful to define parameterized approximations with parameters tuned in the searching process for approximations of concepts. This idea is crucial for methods of construction of concept approximations in particular for rough set methods. In our notation $\#, \$$ denote vectors of parameters which can be tuned in the process of concept approximation.

We assume information granules are also constructed by applying some operations to already constructed information granules by some units (agents). Any such unit (agent) has its own approximation space with objects being information granules. An approximation space of a given agent ag can be related to approximation spaces of children of ag by means of some operation on approximation spaces. The operations on approximation spaces and their properties will be discussed in our next paper.

Let us now recall some basic definitions [11]. If $IS = (U, A)$ is an information system then by $Inf_A^{IS}(u)$ (or by $Inf_A(u)$) we denote the signature of x in IS, i.e., the set $\{(a, a(x)) : a \in A\}$ and by $INF^{IS}(A)$ the set $\{Inf_A(u) : u \in U\}$. If $DT = (U, A, d)$ is a decision table then we assume the set V_d of values of the decision d to be equal to $\{1, \ldots, r(d)\}$ for some positive integer called *the range of* d. The decision d determines a partition $\{C_1, ..., C_{r(d)}\}$ of the universe U, where $C_k = \{x \in U : d(x) = k\}$ for $1 \leq k \leq r(d)$. The set C_k is called the *k-th decision class* of DT.

Now, we are going to define some relations and operations on information systems which will be used in Section 7:

1. *Restriction:* Given an information system $IS = (U, A)$ and a set of attributes $B \subseteq A$, the information system $IS|_B = (U, B)$ is called *the B-restriction of IS.*

2. *Composition:* For given information systems $IS_1 = (U_1, A_1)$ and $IS_2 = (U_2, A_2)$, we define a new information system $IS = (U, A)$, called *composition of IS_1 and IS_2*, by

$$\begin{aligned} U &= \{(x_1, x_2) \in U_1 \times U_2 : Inf_{A_1 \cap A_2}^{IS_1}(x_1) = Inf_{A_1 \cap A_2}^{IS_2}(x_2)\} \\ A &= A_1 \cup A_2 \end{aligned}$$

   for any $x = (x_1, x_2) \in U$ and $a \in A$ the value of a on x is defined by:

$$a(x) = \begin{cases} a(x_1) & \text{if } a \in A_1 \\ a(x_2) & \text{otherwise.} \end{cases}$$

   We denote the composition operation of information systems by $\otimes$, i.e., $IS = IS_1 \otimes IS_2$. If U_1, U_2 are sets of objects of IS_1, IS_2 then we also write $U_1 \otimes U_2$ to denote the composition of the restrictions of information systems IS_1, IS_2 to U_1, U_2, respectively.

3. *Set theoretical operations and relations*: Given two information systems $IS_1 = (U_1, A)$ and $IS_2 = (U_2, A)$, one can define set theoretical operations and relations like union $(IS_1 \cup IS_2)$, intersection $(IS_1 \cap IS_2)$, difference $(IS_1 - IS_2)$, inclusion $(IS_1 \subseteq IS_2)$ and equality $(IS_1 = IS_2)$ in term of information sets $INF^{IS_1}(A)$ and $INF^{IS_2}(A)$. For example

$$IS_1 \subseteq IS_2 \Leftrightarrow INF^{IS_1}(A) \subseteq INF^{IS_2}(A).$$

   Let A be an arbitrary set of attributes. We denote by $UNIVERSE(A)$ the maximal among information systems having A as a set of attributes (with respect to inclusion).

3 Syntax and Semantics of Information Granules

Usually, together with an approximation space, there is also specified a set of formulas Φ expressing properties of objects. Hence, we assume that together with the approximation space $AS_{\#,\$}$ there are given

- a set of formulas Φ over some language,
- semantics $||\bullet||$ of formulas from Φ, i.e., a function from Φ into the power set $P(U)$.

Let us consider an example [15]. We define a language L_{IS} used for elementary granule description, where $IS = (U, A)$ is an information system. The syntax of L_{IS} is defined recursively by

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

The semantics of formulas from L_{IS} with respect to an information system IS is defined recursively by

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

A typical method used by the classical rough set approach [15] for constructive definition of the uncertainty function is the following: for any object $x \in U$, there is given information $Inf_A(x)$ (information signature of x relatively to A) which can be interpreted as a conjunction $EF_B(x)$ of selectors $a = a(x)$ for $a \in A$ and the set $I_{\#}(x)$ is equal to $\|EF_B(x)\|_{IS} = \left\|\bigwedge_{a \in A} a = a(x)\right\|_{IS}$. One can consider a more general case taking as possible values of $I_{\#}(x)$ any set $\|\alpha\|_{IS}$ containing x. Next from the family of such sets the resulting neighborhood $I_{\#}(x)$ can be selected. One can also use another approach by considering more general approximation spaces in which $I_{\#}(x)$ is a family of subsets of U [5], [12].

We present now the syntax and semantics of examples of information granules. These granules are constructed by taking collections of already specified granules. They are parameterized by parameters which can be tuned in applications. In the following sections we discuss some other kinds of operations on granules as well as the inclusion and closeness relations for such granules.

Let us note that any information granule formally can be defined by a pair consisting of the granules syntax and semantics. However, for simplicity of notation we often use only one component of the information granules to denote it.

Elementary granules.

In an information system $IS = (U, A)$, 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

$$\|a = 1 \wedge b = 1\|_{IS} = \{x \in U : a(x) = 1 \;\&\; b(x) = 1\}.$$

The number of conjuncts in the granule can be taken as one of parameters to be tuned what is well known as the drooping condition technique in machine learning [13].

Sequences of granules.

Let us assume that S is a sequence of granules and the semantics $\|\bullet\|_{IS}$ in IS of its elements have been defined. We extend $\|\bullet\|_{IS}$ on S by $\|S\|_{IS} = \{\|g\|_{IS}\}_{g \in S}$.

Example 3 *Granules defined by rules in information systems are examples of sequences of granules. Let IS be an information system and let (α, β) be a new information granule received from the rule* ***if*** *α* ***then*** *β where α, β are elementary granules of IS. The semantics $\|(\alpha, \beta)\|_{IS}$ of (α, β) is the pair of sets $(\|\alpha\|_{IS}, \|\beta\|_{IS})$. If the right hand sides of rules represent decision classes than among parameters to be tuned in classification is the number of conjuncts on the left hand sides of rules. Typical goal is to search for minimal number of such conjuncts (corresponding to the largest generalization) which still guarantee the satisfactory degree of inclusion in a decision class [11], [13].*

Sets of granules.

Let us assume that a set G of granules and the semantics $\|\bullet\|_{IS}$ in IS for granules from G have been defined. We extend $\|\bullet\|_{IS}$ on the family of sets $H \subseteq G$ by $\|H\|_{IS} = \{\|g\|_{IS} : g \in H\}$. One can consider as a parameter of any such granule its cardinality or its size (e.g., the length of such granule representation). In the first case, a typical problem is to search in a given family of granules for a granule of the smallest cardinality sufficiently close to a given one.

Example 4 *One can consider granules defined by sets of rules. Assume that there is a set of rules* $Rule\_Set = \{(\alpha_i, \beta_i) : i = 1, \ldots, k\}$. *The semantics of* $Rule\_Set$ *is defined by*

$$\|Rule\_Set\|_{IS} = \{\|(\alpha_i, \beta_i)\|_{IS} : i = 1, \ldots, k\}.$$

Mentioned above searching problem for the case of rules corresponds to searching for the simplest representation of a given rule collection by another set of rules (or a single rule) sufficiently close to the collection [1].

Example 5 *Let us consider a set* G *of elementary information granules describing possible situations together with decision table* DT_α *representing decision tables for any situation* $\alpha \in G$. *Assume* $Rule\_Set\,(DT_\alpha)$ *to be a set of decision rules generated from decision table* DT_α *(e.g., in the minimal form)[11]. Now let us consider a new granule*

$$\{(\alpha, Rule\_Set\,(DT_\alpha)) : \alpha \in G\}$$

with semantics defined by

$$\{\|(\alpha, Rule\_Set\,(DT_\alpha))\|_{DT} : \alpha \in G\} = \{(\|\alpha\|_{IS}, \|Rule\_Set\,(DT_\alpha)\|_{DT}) : \alpha \in G\}.$$

An example of a parameter to be tuned is the number of situation represented in such granule. A typical task is to search for a granule with the minimal number of situations creating together with the corresponding to them rule sets a granule sufficiently close to the original one.

Extension of granules defined by tolerance relation.

Now we present examples of granules obtained by application of a tolerance relation (i.e., reflexive and symmetric relation; for more information see e.g., [23]).

Example 6 *One can consider extension of elementary granules defined by tolerance relation. Let* $IS = (U, A)$ *be an information system and let* τ *be a tolerance relation on elementary granules of* IS. *Any pair* (α, τ) *is called a* τ-elementary granule. *The semantics* $\|(\alpha, \tau)\|_{IS}$ *of* (α, τ) *is the family* $\{\|\beta\|_{IS} : (\beta, \alpha) \in \tau\}$. *Parameters to be tuned in searching for relevant tolerance granule can be its support (represented by the number of supporting them objects) and its degree of their inclusion (or closeness) in some other granules as well as parameters specifying the tolerance relation.*

Example 7 *Let us consider granules defined by rules of tolerance information systems [23]. Let* $IS = (U, A)$ *be an information system and let* τ *be a tolerance relation on elementary granules of* IS. *If* ***if*** α ***then*** β *is a rule in* IS *then the semantics of a new information granule* $(\tau : \alpha, \beta)$ *is defined by* $\|(\tau : \alpha, \beta)\|_{IS} = \|(\alpha, \tau)\|_{IS} \times \|(\beta, \tau)\|_{IS}$. *Parameters to be tuned are the same as in the case of granules being sets of more elementary granules as well as parameters of the tolerance relation.*

Example 8 *We consider granules defined by sets of decision rules corresponding to a given evidence in tolerance decision tables. Let* $DT = (U, A, d)$ *be a decision table and let* τ *be a tolerance on elementary granules of* $IS = (U, A)$. *Now, any granule* $(\alpha, Rule\_Set\,(DT_\alpha))$ *can be considered as a representative of information granule cluster*

$$(\tau : (\alpha, Rule\_Set\,(DT_\alpha)))$$

with the semantics

$$\|(\tau : (\alpha, Rule\_Set\,(DT_\alpha)))\|_{DT} = \left\{\|(\beta, Rule\_Set\,(DT_\beta))\|_{DT} : (\beta, \alpha) \in \tau\right\}.$$

One can see that the considered case is a special case of information granules from Example 5 with G defined by tolerance relation.

Dynamic granules.

An elementary granule α of the information system IS is non-empty if $\|\alpha\|_{IS} \neq \emptyset$. A non-empty elementary granule β of IS is an extension of α if $\beta = \alpha \wedge \gamma$, where γ is an elementary granule. Let us consider granules defined by some subsets of

$$\{(\beta, Rule\_Set\,(DT_\beta)) : \beta \text{ is an extension of } \alpha\}.$$

The semantics of these new granules is defined as in the case of sets of granules. Any set G of elementary granules and a granule α are specifying new granules

$$\{(\beta, Rule\_Set\,(DT_\beta)) : \beta \text{ is an extension of } \alpha \text{ and } \beta \in G\}$$

important for decision making in dynamically changing environment. Let us consider an example. A DT-path is any sequence $\pi = ((\alpha_1, R_1), \ldots, (\alpha_k, R_k))$ such that α_i is an elementary non-empty granule of IS, $R_i = Rule\_Set(DT_{\alpha_i})$ for $i = 1, ..., k$ and $\alpha_i = \alpha_{i-1} \wedge \gamma_{i-1}$ for some elementary atomic granule γ_{i-1} (e.g. selector $a = v$) with an attribute $a \in A$ not appearing in α_{i-1} for $i = 2, ..., k$. A granule α_{i-1} is called a guard of π if R_{i-1} is not sufficiently close to R_i (what we denote by $non(cl_p(R_{i-1}, R_i))$, where p is the closeness degree). By $Guard(\pi)$ we denote the subsequence of $\alpha_1, \ldots, \alpha_k$ consisting all guards of π. In applications it is important to search for a minimal (in cardinality) granule G satisfying the following condition: for any maximal DT-path π of extensions of α all guards β from $Guard(\pi)$ (i.e. all points in which it is sufficient to change the decision algorithm represented by the set of decision rules) are from G.

One can also consider dynamic granules with tolerance relation. Let $DT = (U, A, d)$ be a decision table and let τ be a tolerance relation on elementary granules of $IS = (U, A)$. Two DT-paths $\pi = ((\alpha_1, R_1), \ldots, (\alpha_k, R_k))$ and $\pi' = ((\beta_1, R'_1), \ldots, (\beta_l, R'_l))$ are *τ-similar* if and only if $(\alpha_{i_s}, \beta_{j_s}) \in \tau$ for $s = 1, \ldots, r$, where $Guard(\pi) = (\alpha_{i_1}, \ldots, \alpha_{i_r})$ and $Guard(\pi') = (\beta_{j_1}, \ldots, \beta_{j_r})$. Let us assume τ has the following property:

if $(\beta, \alpha) \in \tau$ then the granules $Rule\_Set\,(DT_\alpha)$ and $Rule\_Set\,(DT_\beta)$ are sufficiently close.

Having such tolerance relation one can search for a set G of guards of the smaller size. To specify the task is enough to change in the above formulated problem the condition for the maximal path to the following one: for any maximal path π of extensions of α there exists a τ-similar path π' to π such that all guards β from $Guard(\pi')$ (i.e. all points where it is sufficient to change the decision algorithm represented by the set of decision rules) are from G.

Labeled graph granules.

We discuss graph granules and labeled graph granules as notions extending previously introduced granules defined by tolerance relation and dynamic granules.

Example 9 *Let us consider granules defined by pairs (G, E), where G is a finite set of granules and $E \subseteq G \times G$. Let $IS = (U, A)$ be an information system. The semantics of a new information granule (G, E) is defined by $\|(G, E)\|_{IS} = (\|G\|_{IS}, \|E\|_{IS})$, where $\|G\|_{IS} = \{\|g\|_{IS} : g \in G\}$ and $\|E\|_{IS} = \{(\|g\|, \|g'\|) : (g, g') \in E\}$.*

Example 10 *Let G be a set of granules. Labeled graph granules over G are defined by (X, E, f, h), where $f : X \to G$ and $h : E \to P(G \times G)$. We also assume one additional condition*

$$\text{if } (x, y) \in E \text{ then } (f(x), f(y)) \in h(x, y).$$

The semantics of the labeled graph granule (X, E, f, h) *is defined by*

$$\{(\|f(x)\|_{IS}, \|h(x,y)\|_{IS}, \|f(y)\|_{IS}) : (x,y) \in E\}.$$

Let us summarize the above presented considerations. One can define the set of granules G as the least set containing a given set of elementary granules G_0 and closed with respect to the defined above operations of new granule construction.

We have the following examples of granule construction rules:

$$\frac{\alpha_1, \ldots, \alpha_k\text{- elementary granules}}{\{\alpha_1, \ldots, \alpha_k\}\text{- granule}}$$

$$\frac{\alpha_1, \alpha_2\text{- elementary granules}}{(\alpha_1, \alpha_2)\text{- granule}}$$

$$\frac{\alpha\text{- elementary granule }, \tau\text{- tolerance relation on elementary granules}}{(\tau : \alpha)\text{- granule}}$$

$$\frac{G\text{- a finite set of granules }, E \subseteq G \times G}{(G, E)\text{- granule}}$$

Let us observe that in case of granules constructed with application of tolerance relation we have the rule restricted to elementary granules. To obtain a more general rule like

$$\frac{\alpha\text{- graph granule, } \tau\text{- tolerance relation on graph granules}}{(\tau : \alpha)\text{- granule}}$$

it is necessary to extend the tolerance (similarity, closeness) relation on more complex objects.

One more interesting class of information granules create classifiers [19]. Parameters to be tuned are voting strategies, matching strategies of objects against rules as well as other discussed above parameters like closeness of granule in the target granule.

In presented examples we have discussed parameterized information granules. We have pointed out that the process of the parameters 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.

In the following section we discuss the problem of inclusion and closeness definition for information granules.

4 Granule Inclusion and Closeness

In this section we will discuss inclusion and closeness of different information granules introduced in the previous section. Let us mention that 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' of degree at least p will be denoted by $\nu_p(G, G')$. Similarly, the closeness relation between granules G, G' of 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 construction of hierarchical granules and their closeness can be described by the following recursive meta-rule: if granules of order $\leq k$ and their closeness have been defined then the closeness $cl_p(G, G')$ (at least in 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 $IS = (U, A)$. They are defined by $EF_B(x)$, where EF_B is a conjunction of selectors of the form $a = a(x)$, $B \subseteq A$ and $x \in U$. Let $G_{IS} = \{EF_B(x) : B \subseteq A \;\&\; x \in U\}$. In the standard rough set model

[15] elementary granules describe indiscernibility classes with respect to some subsets of attributes. In a more general setting see e.g. [23] tolerance (similarity) classes are described. The crisp inclusion of α in β, where $\alpha, \beta \in \{EF_B(x) : B \subseteq A \ \& \ x \in U\}$ is defined by $\|\alpha\|_{IS} \subseteq \|\beta\|_{IS}$, where $\|\alpha\|_{IS}$ and $\|\beta\|_{IS}$ are sets of objects from IS satisfying α and β, respectively. The non-crisp inclusion, known in KDD [2], for the case of association rules is defined by means of two thresholds t and t':

$$\begin{aligned} support_{IS}(\alpha,\beta) &= card\left(\|\alpha \wedge \beta\|_{IS}\right) \geq t, \\ accuracy_{IS}(\alpha,\beta) &= \frac{support_{IS}(\alpha,\beta)}{card\left(\|\alpha\|_{IS}\right)} \geq t'. \end{aligned}$$

Elementary granule inclusion in a given information system IS can be defined using different schemes, e.g., by

$\nu_{t,t'}^{IS}(\alpha,\beta)$ if and only if $support_{IS}(\alpha,\beta) \geq t$ and $accuracy_{IS}(\alpha,\beta) \geq t'$

or

$\nu_t^{IS}(\alpha,\beta)$ if and only if $accuracy_{IS}(\alpha,\beta) \geq t$.

The closeness of granules can be defined by

$cl_{t,t'}^{IS}(\alpha,\beta)$ if and only if $\nu_{t,t'}^{IS}(\alpha,\beta)$ and $\nu_{t,t'}^{IS}(\beta,\alpha)$ hold.

Decision rules as granules.

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

$\nu_{t,t'}^{IS}(g,g')$ if and only if $\nu_{t,t'}^{IS}(\alpha,\alpha')$ and $\nu_{t,t'}^{IS}(\beta,\beta')$.

The closeness can be defined by

$cl_{t,t'}^{IS}(g,g')$ if and only if $\nu_{t,t'}^{IS}(g,g')$ and $\nu_{t,t'}^{IS}(g',g)$.

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

$\nu_t^{IS}((\alpha,\beta),(\alpha',\beta'))$ if and only if $\nu_{t_1,t_2}^{IS}(\alpha,\alpha')$ and $\nu_{t_1,t_2}^{IS}(\beta,\beta')$ and $t = w_1 \cdot t_1 + w_2 \cdot t_2$

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:

$\nu_{t,t'}^{IS}((\alpha,\tau)(\beta,\tau))$ if and only if

$\nu_{t,t'}^{IS}(\alpha',\beta')$ for any α',β' such that $(\alpha,\alpha') \in \tau$ and $(\beta,\beta') \in \tau$

and the following closeness measure:

$cl_{t,t'}^{IS}((\alpha,\tau)(\beta,\tau))$ if and only if $\nu_{t,t'}^{IS}((\alpha,\tau)(\beta,\tau))$ and $\nu_{t,t'}^{IS}((\beta,\tau)(\alpha,\tau))$.

It can be important for some applications to define 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 one:

$cl_{t,t'}^{IS}(\alpha,(\alpha,\tau))$ if and only if $cl_{t,t'}^{IS}(\alpha,\beta)$ for any $(\alpha,\beta) \in \tau$.

Sets of rules.

An important problem related to association rules is that the number of such rules generated even from simple data table can be large. Hence, one should search for methods of aggregating close association rules. We suggest that 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

$$Rule\_Set = \{(\alpha_i,\beta_i) : i = 1,\ldots,k\},$$

$$Rule\_Set' = \{(\alpha'_i,\beta'_i) : i = 1,\ldots,k'\}.$$

One can treat them as higher order information granules. These new granules *Rule_Set*, *Rule_Set'* can be treated as close in a degree at least t (in IS) 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 IS) in 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 IS) in degree at least t.

Another way of defining 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. By $I(\beta_i')$ we denote the set $\{j : cl_p^{IS}(\beta_j', \beta_i')\}$ for any $i = 1, \ldots, k'$.

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

$$cl_p^{IS}\left(\bigvee_{j \in I(\beta_i')} \beta_j', \bigvee_{j \in J} \beta_j\right) \text{ and } cl_p^{IS}\left(\bigvee_{j \in I(\beta_i')} \alpha_j', \bigvee_{j \in J} \alpha_j\right)$$

and for closeness we assume

$cl_p^{IS}(Rule\_Set, Rule\_Set')$ if and only if

$$\nu_p^{IS}(Rule\_Set, Rule\_Set') \text{ and } \nu_p^{IS}(Rule\_Set', Rule\_Set).$$

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

$$cl_p(G_1, G_2) \text{ if and only if } cl_p(\alpha_1 \vee \alpha_2 \vee \alpha_3, \gamma_1) \text{ and } cl_p(\beta_1 \vee \beta_2, \gamma_2).$$

One can consider a searching problem for a granule *Rule_Set'* of minimal size such that *Rule_Set* and *Rule_Set'* are close (see e.g. [1]).

Granules defined by sets of granules.

The previously discussed methods of inclusion and closeness definition can be easily adopted for the case of 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

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

and the closeness by

$$cl_{t,t'}^{IS}(G, H) \text{ if and only if } \nu_{t,t'}^{IS}(G, H) \text{ and } \nu_{t,t'}^{IS}(H, G).$$

Inclusion for complex granules specified by inclusion of their parts is symbolized in Figure 2.

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

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

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

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

$$\frac{cl_p(\alpha, \alpha'), cl_p(\beta, \beta')}{cl_p((\alpha, \beta), (\alpha', \beta'))}$$

$$\frac{\text{for any } \alpha' \in \tau(\alpha) \text{ there is } \beta' \in \tau(\beta) \text{ such that } \nu_p(\alpha', \beta')}{\nu_p((\tau : \alpha), (\tau : \beta))}$$

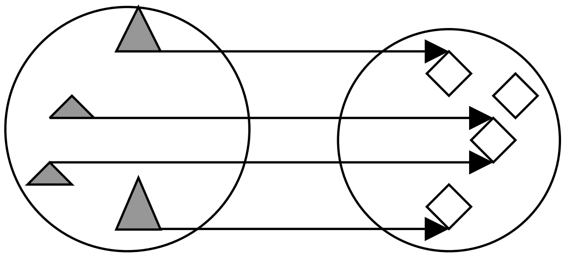

Figure 2: Two Sets of Granules

$$\frac{cl_p(G, G'), cl_p(E, E')}{cl_p((G, E), (G', E'))}$$

where $\alpha, \alpha', \beta, \beta'$ are elementary granules, G, G' are finite sets of elementary granules.

One can also present other discussed 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 IS (under the chosen definition of inclusion and closeness). Some of them are derivable from others. We will see in the next part of our paper that there are also some operations of new granules construction specific for a given information system. In this case one should extract from existing data the specific inference rules.

5 Information Granule Construction by Schemes of Approximate Reasoning

In this section we elaborate a general scheme for information granule construction in distributed systems introduced in [25]. To explain the basic constructions we use terminology from multi-agent area [9].

Teams of agents are organized, e.g., along the schemes of decomposition of complex objects (e.g., representing situations on the road) into trees. The trees are represented by expressions called *terms*. Two granules are defined for any term t under a given valuation val of leaf agents of t in the set of input granules. They are called the lower and upper approximations of t under val.

The necessity to consider rather approximation of granule returned by a given term t under a given valuation val than the exact value of t under val is a consequence of the ability of agents to perceive in approximate sense only information granules received from other agents. Hence, approximate reasoning in distributed environment requires a construction of interfaces between agents (information sources or units) enabling effective learning by agents of concepts definable by other agents. In the paper, we suggest a solution based on exchanging views of agents on objects with respect to a given concept. An agent delivering a concept is submitting positive and negative examples (objects) with respect to a given concept. The agent receiving this information describes objects using its own attributes. In this way a data table (called a *decision table*) is created and the approximate description of concept can be extracted by the receiving agent. Our solution is based on the rough set approach. We propose to use the parameterized approximation spaces [25, 26] to allow appropriate tuning of concept perception by agents (see Figures 3 and 4).

One can consider different problems related to synthesis of schemes of approximate reasoning defined by terms. For example, one can look for a strategy returning for any valuation val (representing global situation) a term (agent scheme) t with the following property: the lower and upper

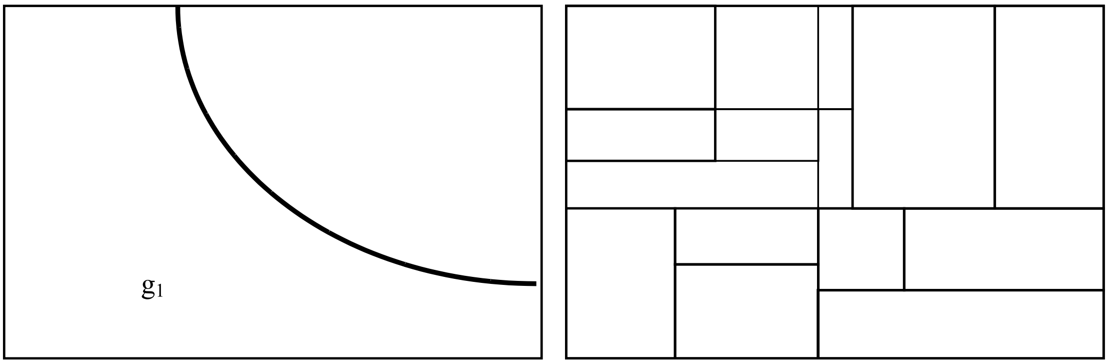

Figure 3: Concept g_1 – Information Granule of $ag_1 \in Ag$ (left) and Communication Interface Defined by Data Table (right)

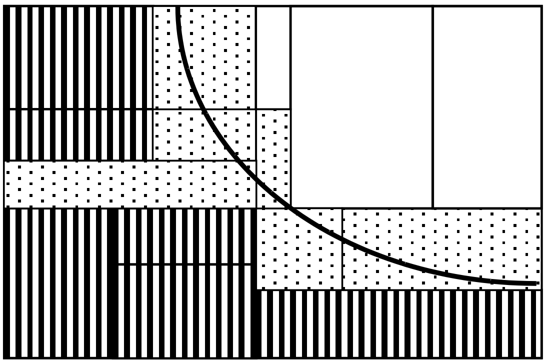

Figure 4: Lower and Upper Approximation of g_1 by $ag \in Ag$

values of t under val are sufficiently close to a given soft granule given by a specification if and only if the global situation represented by val really matches this specification.

We assume any non leaf-agent ag is equipped with an operation $o\,(ag) : U_{ag}^{(1)} \times \ldots \times U_{ag}^{(k)} \rightarrow U_{ag}^{(0)}$ and has different approximation spaces $AS_{ag}^{(i)} = \left(U_{ag}^{(i)}, I_{ag}^{(i)}, \nu_{SRI}\right)$, where $i = 0, \ldots, k$. We assume that the agent ag is perceiving objects by measuring values of some available attributes. Hence some objects can become indiscernible [15]. This influences the specification of any operation $o\,(ag)$. We consider a case when arguments and values of operations are represented by attribute value vectors. Hence instead of the operation $o\,(ag)$ we have its inexact specification $o^*\,(ag)$ taking as arguments $I_{ag}^{(1)}\,(x_1), \ldots, I_{ag}^{(k)}\,(x_k)$ for some $x_1 \in U_{ag}^{(1)}, \ldots, x_k \in U_{ag}^{(k)}$ and returning the value $I_{ag}^{(0)}\,(o(ag)(x_1, \ldots, x_k))$ if $o(ag)(x_1, ..., x_k)$ is defined, otherwise the empty set. This operation can be extended to the operation $o^*(ag)$ with arguments being definable sets (in approximation spaces attached to arguments) and with values in the family of all non-empty subsets of $U_{ag}^{(0)}$. Let $X_1, \ldots, X_k$ be definable sets. We define

$$o^*\,(ag)\,(X_1, \ldots, X_k) = \bigcup_{x_1 \in X_1, \ldots, x_k \in X_k} o^*\,(ag)\left(I_{ag}^{(1)}\,(x_1), \ldots, I_{ag}^{(k)}\,(x_k)\right).$$

In the sequel, for simplicity of notation, we write $o\,(ag)$ instead of $o^*\,(ag)$.

This idea can be formalized as follows. First we define terms representing schemes of agents.

Let $X_{ag}, Y_{ag}, \ldots$ be agent variables for any leaf-agent $ag \in Ag$. Let $o\,(ag)$ denote a function of arity k. We have mentioned that it is an operation from Cartesian product of

$$Def\_Sets(AS_{ag}^{(1)}), \ldots, Def\_Sets(AS_{ag}^{(k)}) \text{ into } P\left(U_{ag}^{(0)}\right),$$

where $Def\_Sets(AS_{ag}^{(i)})$ denotes the family of sets definable in $AS_{ag}^{(i)}$. Using the above variables and functors we define terms in a standard way, for example

$$t = o\,(ag)\,(X_{ag_1}, X_{ag_2})\,.$$

Such terms can be treated as description of complex information granules. By a valuation we mean any function *val* defined on the agent variables with values being definable sets satisfying $val(X_{ag}) \subseteq U_{ag}$ for any leaf-agent $ag \in Ag$. Now we can define the lower and the upper values of any term t under the valuation *val* with respect to a given approximation space $AS_{ag}^{(i)}$ of an agent ag

1. If t is of the form X_{ag_i} and $val(t) \subseteq U_{ag}^{(i)}$ then

$$val\left(LOW, AS_{ag}^{(i)}\right)(t) = LOW\left(AS_{ag}^{(i)}, val(t)\right)$$

$$val\left(UPP, AS_{ag}^{(i)}\right)(t) = UPP\left(AS_{ag}^{(i)}, val(t)\right)$$

   else the lower and the upper values are undefined.

2. If $t = o(ag)(t_1, \ldots, t_k)$, where $t_1, \ldots, t_k$ are terms and $o\,(ag)$ is an operation of arity k, then

   (a) if $val\left(LOW, AS_{ag}^{(i)}\right)(t_i)$ is defined for $i = 1, \ldots, k$ then
   $val\left(LOW, AS_{ag}^{(0)}\right)(t) =$
   $$LOW\left(AS_{ag}^{(0)}, o\,(ag)\left(val\left(LOW, AS_{ag}^{(1)}\right)(t_1), \ldots, val\left(LOW, AS_{ag}^{(k)}\right)(t_k)\right)\right)$$
   else $val\left(LOW, AS_{ag}^{(0)}\right)(t)$ is undefined,

   (b) if $val\left(UPP, AS_{ag}^{(i)}\right)(t_i)$ is defined for $i = 1, \ldots, k$ then
   $val\left(UPP, AS_{ag}^{(0)}\right)(t) =$
   $$UPP\left(AS_{ag}^{(0)}, o\,(ag)\left(val\left(UPP, AS_{ag}^{(1)}\right)(t_1), \ldots, val\left(UPP, AS_{ag}^{(k)}\right)(t_k)\right)\right)$$
   else $val\left(UPP, AS_{ag}^{(0)}\right)(t)$ is undefined.

For illustrative example of computation of the lower and upper approximations of terms the reader is referred to [27].

Let us observe that the set $val(UPP, AS_{ag}^{(0)})(t) - val(LOW, AS_{ag}^{(0)})(t)$ can be treated as the boundary region of t under *val*. Moreover, in the process of term construction we have additional parameters to be tuned for obtaining sufficiently high support and accuracy, namely the approximation operations.

A concept X specified by the customer-agent is ***sufficiently close to*** t ***under a given set*** Val ***of valuations*** if X is included in the upper approximation of t under any $val \in Val$ and X includes the lower approximation of t under any $val \in Val$ as well as the size of the boundary region of t under Val, i.e.,

$$card\left(\bigcap_{val \in Val} val\left(UPP, AS_{ag}^{(0)}\right)(t) - \bigcup_{val \in Val} val\left(LOW, AS_{ag}^{(0)}\right)(t)\right),$$

is sufficiently small relatively to $\bigcap_{val \in Val} val\left(UPP, AS_{ag}^{(0)}\right)(t)$.

6 Robustness of Granules

Extracting robust patterns (i.e., patterns with properties not deviating to much under small disturbances of parameters) from data is an important task for many applications, in particular related to KDD and spatio-temporal reasoning.

Robust granules can be constructed by means of robust schemes of approximate reasoning (schemes of granule construction) represented by clusters of approximate reasoning schemes. We will show how such clusters can be represented using an approach developed in rough mereology (see e.g., [18]).

The robustness of constructed granules can be defined using a notion analogous to the continuity of function in a given point.

Definition 11 *Let $f : X_1 \times \ldots \times X_k \to X$ and let us assume closeness relations $cl_p^{(i)}(\bullet,\bullet)$ and $cl_p(\bullet,\bullet)$ in X_i for $i = 1,\ldots,k$ and X are given. We say that f is $(\varepsilon;\varepsilon_1,...,\varepsilon_k)$-*robust *in a given point $(x_1,\ldots,x_k)$ from $X_1 \times \ldots \times X_k$ if and only if for any $(y_1,\ldots,y_k)$ from $X_1 \times \ldots \times X_k$ if $cl_{1-\varepsilon_i}^{(i)}(x_i,y_i)$ for $i = 1,\ldots,k$ then $cl_{1-\varepsilon}(f(x_1,\ldots,x_k)), f(y_1,\ldots,y_k))$.*

We can formulate simple but useful property for robustness checking.

Proposition 12 *Let an operation*

$$F(ag) : \prod_{i=1}^{k} Def\_Sets\left(AS_{ag}^{(i)}\right) \to Def\_Sets\left(AS_{ag}^{(0)}\right)$$

be $(\varepsilon;\varepsilon_1,\ldots,\varepsilon_k)$ - robust in $(g_1,\ldots,g_k)$ and let for every agent ag_i operation

$$F(ag_i) : \prod_{j=1}^{l_i} Def\_Sets\left(AS_{ag}^{(j)}\right) \to Def\_Sets\left(AS_{ag}^{(i)}\right)$$

be $(\varepsilon_i;\varepsilon_1^{(i)},\ldots,\varepsilon_{l_i}^{(i)})$ - robust in $(g_1^{(i)},\ldots,g_{l_i}^{(i)})$ for $i = 1,\ldots,k$. Then the operation

$$F\left(ag\right)\left(F\left(ag_1\right)\left(\ldots\right),\ldots,F\left(ag_k\right)\left(\ldots\right)\right)$$

being the superposition of $F(ag)$ and $F(ag_1),\ldots,F(ag_k)$ is

$$(\varepsilon;\varepsilon_1^{(1)},\ldots,\varepsilon_{l_1}^{(1)},\ldots,\varepsilon_1^{(k)},\ldots,\varepsilon_{l_k}^{(k)}) - robust$$

in

$$\left(g_1^{(1)},\ldots,g_{l_1}^{(1)},\ldots,g_1^{(k)},\ldots,g_{l_k}^{(k)}\right).$$

In Proposition 12 we consider operations with values being definable sets. These operations can be received from $o^*(ag)$ by applying the lower (or upper) approximation operation of $AS_{ag}^{(0)}$ to the arguments and values of $o^*(ag)$. Using Proposition 12 one can easily derive the robustness condition for terms describing the construction of information granules.

An important practical problem is to discover rules for decomposition of a given threshold ε into thresholds $\varepsilon_1,\ldots,\varepsilon_k$ for a given operation

$$F(ag) : \prod_{i=1}^{k} Def\_Sets\left(AS_{ag}^{(i)}\right) \to Def\_Sets\left(AS_{ag}^{(0)}\right)$$

in such a way that $F(ag)$ is $(\varepsilon;\varepsilon_1,\ldots,\varepsilon_k)$ - robust in a given point [11].

Let us consider a more general problem assuming a set of agents Ag together with their closeness measures $cl_p^{ag}(\bullet,\bullet)$ of information granules are given for ag from Ag. Let $ag \in Ag$ and let g be a standard (prototype) from $Def\_Sets\left(AS_{ag}^{(0)}\right)$ together with an uncertainty coefficient $\varepsilon \in [0,1)$. Let us assume that $In(Ag)$ is a given subset of input (inventory [11]) agents from Ag. A task is to synthesize a term $t(X_{ag_1},\ldots,X_{ag_k})$, where $X_{ag_1},\ldots,X_{ag_k}$ are some variables of agents from $In(Ag)$ and a valuation val of variables such that:

- $val(LOW, AS_{ag}^{(0)})(t) \subseteq g \subseteq val(UPP, AS_{ag}^{(0)})(t)$,
- $cl_{1-\varepsilon/2}^{ag}(g, val(LOW, AS_{ag}^{(0)})(t))$ and $cl_{1-\varepsilon/2}^{ag}(g, val(UPP, AS_{ag}^{(0)})(t))$,
- the operations

$$t_{LOW}(val) = val(LOW, AS_{ag}^{(0)})(t),$$
$$t_{UPP}(val) = val(UPP, AS_{ag}^{(0)})(t)$$

defined by $t(X_{ag_1}, \ldots, X_{ag_k})$ are $(\varepsilon/2; \varepsilon_1, \ldots, \varepsilon_k)$-robust in a point

$$(val(X_{ag_1}), \ldots, val(X_{ag_k}))$$

for some $\varepsilon_1, \ldots, \varepsilon_k$ greater than a given threshold $\delta > 0$.

Now, we will discuss applications of closeness between granules for deducing that they share close sets of properties. Informally speaking, close granules should share similar properties. Assuming $\mathcal{G}$ is a given set of granules and $R \subseteq \mathcal{G} \times \mathcal{G}$ is a relation describing similarity (closeness), indiscernibility or similar functionality of information granules and $\mathcal{P}$ is a given set of properties (unary predicates) defined on information granules we can formulate the relationship between the relation R and the set $\mathcal{P}$ of properties as follows:

$$\forall g, g' \in \mathcal{G} \forall P \in \mathcal{P}[R(g, g') \rightarrow (P(g) \rightarrow P(g'))].$$

The property is called $(R, \mathcal{P})$ *-relationship*. Searching methods for relation R satisfying the specified above condition is an important problem which we will discuss in our next paper.

In case of multi-agent environment one should consider different $(R, \mathcal{P})$ -relationships for different agents. One of the fundamental question is how to deduce that close complex information granules share similar properties. Following an idea developed in rough mereological approach we have presented in [27] the main steps for such deduction. One can derive sufficient conditions for synthesis of robust complex information granules by analysis of derivations from given granules using decomposition rules [27].

The main role in such deduction play, so called *decomposition rules* [18], [27]. They have the following form:

$$o(ag) : \frac{P_{ag}(\bullet), \varepsilon, st(ag), cl_{1-\varepsilon}^{ag}(\bullet, \bullet)}{P_{ag_1}^{(1)}(\bullet), \varepsilon_1, st(ag_1), cl_{1-\varepsilon_1}^{ag_1}(\bullet, \bullet); \ldots; P_{ag_k}^{(k)}(\bullet), \varepsilon_k, st(ag_k), cl_{1-\varepsilon_k}^{ag_k}(\bullet, \bullet)}$$

where $o(ag)$ is k-ary operation of ag; $st(ag)$, $st(ag_1), \ldots, st(ag_k)$ are standard information granules (prototypes); $P_{ag}(\bullet)$, $P_{ag_1}^{(1)}(\bullet), \ldots, P_{ag_k}^{(k)}(\bullet)$ are properties; ε, $\varepsilon_1, \ldots, \varepsilon_k$ are uncertainty coefficients; $cl_{1-\varepsilon}^{ag}$, $cl_{1-\varepsilon_1}^{ag_1}, \ldots, cl_{1-\varepsilon_k}^{ag_k}$ are closeness relations attached to agents ag, $ag_1, \ldots, ag_k$, respectively.

We assume the decomposition rule is true if the following conditions are satisfied:

- $o(ag)(st(ag_1), \ldots, st(ag_k)) = st(ag)$,
- $P_{ag_1}^{(1)}(st(ag_1)) \wedge \ldots \wedge P_{ag_k}^{(k)}(st(ag_k)) \rightarrow P_{ag}(st(ag))$,
- $cl_{1-\varepsilon_1}^{ag_1}(st(ag_1), g_1) \wedge \ldots \wedge cl_{1-\varepsilon_k}^{ag_k}(st(ag_k), g_k) \rightarrow cl_{1-\varepsilon}^{ag}(st(ag), g)$ for any $g_1, \ldots, g_k$ from $\mathcal{G}$, where $g = o(ag)(g_1, \ldots, g_k)$,
- $cl_{1-\varepsilon}^{ag}(g, g') \rightarrow (P_{ag}(g) \rightarrow P_{ag}(g'))$ for any $g, g' \in \mathcal{G}$.

The first condition states that the standard granule at ag is equal to the result of operation $o(ag)$ on standard granules $st(ag_1), \ldots, st(ag_k)$. The second condition allows to deduce that the standard granule $st(ag)$ has property P_{ag} if the standard granules $st(ag_1), \ldots, st(ag_k)$ have properties $P_{ag_1}^{(1)}, \ldots, P_{ag_k}^{(k)}$, respectively. The next condition allows to infer that $st(ag)$ is close to $o(ag)(g_1, \ldots, g_k)$ in degree at least $1 - \varepsilon$ if the standard granule $st(ag_i)$ is close to g_i in degree at least $1 - \varepsilon_i$ for any $i = 1, \ldots, k$. The last condition means that if granules g, g' are close in degree at least $1 - \varepsilon$ and g has property P_{ag} then g' has also this property.

Hence we obtain the following basic lemma [18], [27]:

Lemma 13 *Assuming the decomposition rule for $o(ag)$ is true and the following conditions*

$$P^{(1)}_{ag_1}(st(ag_1)), cl^{ag_1}_{1-\varepsilon_1}(st(ag_1), g_1), \ldots, P^{(k)}_{ag_k}(st(ag_k)), cl^{ag_k}_{1-\varepsilon_k}(st(ag_k), g_k)$$

hold we obtain that $P_{ag}(g)$ holds too, where $g = o\,(ag)\,(g_1, \ldots, g_k)$.

One can observe that it is possible to derive sufficient conditions for synthesis of robust information granules by composition of the above discussed rules [18], [27].

A challenge for knowledge discovery is to discover decomposition rules corresponding to operations available by local agents [14].

We will present a searching method for decomposition of specifications given by data tables.

7 Synthesis of Complex Objects by Decomposition

In this part of the paper we consider a synthesis method of complex objects by multi-agent system S [14]. In the system S, agents are connected in a tree structure. Every agent ag of the system can perform n tasks specified by *standard table of ag* consisting of descriptions of standard objects $st_i(ag)$, for $i = 1, ..., n$. It is assumed that for any agent ag not being a leaf, there is the set of composition rules:

$$\{R_j : o_j(ag) \leftarrow (o_j(ag_1), ..., o_j(ag_k))\}$$

producing objects $o_j(ag)$ at ag from objects $o_j(ag_1), ..., o_j(ag_k)$ delivered by children $ag_1, ..., ag_k$; in particular $st_j(ag)$ at ag is constructed from standards $st_j(ag_1)$,..., $st_j(ag_k)$ delivered by children $ag_1, ..., ag_k$ of ag in the agent structure. In addition, every composition rule R_j is given together with a tolerance composition condition $\epsilon = \varphi_j(\epsilon_1, \epsilon_2, ..., \epsilon_k)$, which means that if children-agents $ag_1, ..., ag_k$ realize their tasks satisfying specifications $st_j(ag_1)$,..., $st_j(ag_k)$ with closeness greater than $1-\epsilon_1, 1-\epsilon_2, ..., 1-\epsilon_k$, respectively, then the parent-agent ag can perform a task specified by $st_j(ag)$ in degree greater than $1-\epsilon$. Using such assumptions, approximate reasoning schemes of synthesis are constructed [18]. We consider the case of system S in which objects constructed by any agent are described by means of attributes and specifications of tasks represented by information systems. We investigate a searching problem for standards and tolerance decomposition conditions from specification. We analyze the complexity of this problem and propose some heuristics.

7.1 Decomposition Problem of Complete Task Specification

Let us consider the agent ag and his two sub-agents ag_1 and ag_2. We assume that agents ag, ag_1 and ag_2 can synthesize products described by attribute sets A, A_1 and A_2, respectively, where $A = A_1 \cup A_2$ and $A_1 \cap A_2 = \emptyset$. Any information system $IS = (U, A)$ (resp. $IS_1 = (U_1, A_1)$ and $IS_2 = (U_2, A_2)$) is called the *standard* table of ag (for ag_1 and ag_2, respectively.).

We assume that the agent ag is obtaining a specification of tasks in the form of an information system $IS = (U, A)$ (standard table). Two standard tables $IS_1 = (U_1, A_1)$ and $IS_2 = (U_2, A_2)$ (of sub-agents ag_1 and ag_2, respecitvely) are said to be *consistent with* IS if

$$IS_1 \otimes IS_2 \subseteq IS$$

A *consistent covering of* IS is any set of pairs of standard tables $P = \{(IS^i_1, IS^i_2) : i = 1, 2, ...\}$ satisfying $\bigcup_i (IS^i_1 \otimes IS^i_2) = IS$.

A consistent covering (of IS) is *optimal* if it consists of the smallest number of standard pairs among all consistent coverings (of IS).

Two sets of attributes $A_1 \subseteq A$ and $A_2 \subseteq A$ define a partition of $IS = (U, A)$ into two tables $IS|_{A_1} = (U, A_1)$ and $IS|_{A_2} = (U, A_2)$. Let us observe that if $P = \{(IS^i_1, IS^i_2) : i = 1, 2, ...\}$ is consistent covering of IS, then $\bigcup IS^i_1 = IS|_{A_1}$ and $\bigcup IS^i_2 = IS|_{A_2}$. Hence, the problem of searching for consistent covering of specification IS can be called the "*task decomposition problem*". We consider some optimization problems for task decomposition in multi agent systems.

1. Problem of searching for a consistent pair of standard tables $IS_1 = (U_1, A_1)$ and $IS_2 = (U_2, A_2)$ such that $card(IS_1 \otimes IS_2)$ is maximal.

2. Problem of searching for consistent covering of IS consisting of the minimal number of pairs of standard tables.

Let us start from formal definition of decomposition problems. Let $IS = (U, A)$ be an information system specifying the task for agent ag. We assume there is a predefined partition of the attribute set A into two disjoint subsets A_1 and A_2 of A, which can be explored by children ag_1 and ag_2 of ag, respectively. Let $IS_1 = (U, A_1)$ and $IS_2 = (U, A_2)$ denote the restrictions of information system IS to A_1 and A_2, respectively. The consistent covering problem can be now redefined as searching for a set $P = \{(IS_1^i, IS_2^i) : i = 1, 2, ...\}$ where $IS_1^i \subseteq IS_1, IS_2^i \subseteq IS_2$ with the minimal $card(P)$ satisfying $\bigcup_i (IS_1^i \otimes IS_2^i) = IS$.

Let us consider now a bipartite graph $G = (V_1 \cup V_2, E)$, where V_i is the set of objects of an information system IS_i for $i = 1, 2$. A pair of vertices (v_1, v_2) belongs to E iff $Inf_{A_1}^{IS}(v_1) \otimes Inf_{A_2}^{IS}(v_2) \in INF^{IS}(A)$. Hence every pair of consistent standard tables $IS_1 = (U_1, A_1)$ and $IS_2 = (U_2, A_2)$ (i.e., satisfying $(IS_1 \otimes IS_2) \subseteq IS$) corresponds exactly to one complete bipartite subgraph $G' = (U_1 \cup U_2, E')$ and consistent covering set P can be represented by a set of complete bipartite subgraphs covering the graph G. From this observation one can transform the covering problem by pairs of standard tables to the covering problem a bipartite graph by complete subgraphs and a searching problem for a maximal consistent pair of standard tables to the searching problem for a maximal complete subgraph in a bipartite graph.

The considered decomposition problems are (polynomially) equivalent to the bipartite graph problems. Therefore computational complexity of the decomposition problems follows from complexity of corresponding graph problems (which are known to be NP-hard problems).

The decomposition problem is closely related to the bipartite graph problem, therefore every approximating algorithm for the graph problem is a good heuristic for the decomposition problem. We consider here two decomposition problems. The first one is the searching problem for the minimal covering set consisting of consistent standard pairs. The second problem is the searching problem for one optimal consistent pair of standard tables. The former problem is equivalent to the bipartite graph covering problem. One of the simplest heuristic can be described as follows.

HEURISTIC (Related to covering by pairs of standard tables)

Input: Bipartite graph $G = (V_1 \cup V_2, E)$.

Output: A set S of complete subgraphs covering the graph G.

Step 1. $S = \emptyset$.

Step 2. Search for a maximal complete subgraph G_0 of G.

Step 3. $S = S \cup \{G_0\}$; $G = G - G_0$.

Step 4. If $G = \emptyset$ then stop, else go to Step 2.

The searching problem for one consistent pair of standard table is equivalent to the maximal complete subgraph problem. The main idea of the presented below heuristic is to gradually construct a complete subgraph using vertices with the maximal degree.

HEURISTIC (Related to searching for maximal pair of standard tables)

Input: Bipartite graph $G = (V_1 \cup V_2, E)$.

Output: A complete subgraph $G' = (U_1 \cup U_2, E')$ of G with semi-maximal $card(U_1 \times U_2)$.

Step 1. $G' = \emptyset$.

Step 2. Find a vertex $v \in G$ with the maximal degree.

Step 3. If $v \in V_1$ (or $v \in V_2$) then restrict V_2 (or V_1) to a set of neighbors of v, i.e., $V_2 = \{v_2 : (v, v_2) \in E\}$ $(V_1 = \{v_1 : (v_1, v) \in E\})$.

Step 4. Append v to G' and remove from G vertices not connected with v.

Step 5. If $G = \emptyset$ then stop else go to Step 2.

Step 6. Among returned possible complete subgraphs G' find a graph with the maximal number of edges $(card(U_1 \times U_2))$.

7.2 Decomposition of Incomplete Specification

In Section 7.1 we have investigated a decomposition problem for such specification, that consists of complete information about task. However, complete specification can be of very large size being infeasible for complex tasks. Usually in practice, we have to deal with partial specification of tasks only. A case when only partial information about specification is available will be considered in this section.

Let $DT = (U, A, d)$ be a given decision table describing the partial specification of tasks for the agent ag, where $d : U \to \{+, -\}$. Any $u \in U$ is called *an example*. We say that u is a *positive example* (satisfies the specification) if $d(u) =$ '+', and u is a *negative example* (does not satisfy the specification) if $d(u) =$ '−'. Thus, the decision attribute d defines a partition of DT into two information systems $DT^+ = (U^+, A)$ and $DT^- = (U^-, A)$, where

$$U^+ = \{u \in U : d(u) = '+'\ \} \text{ and } U^- = \{u \in U : d(u) = '-'\}.$$

The tables $\underline{\underline{DT}} = DT^+$ and $\overline{\overline{DT}} = UNIVERSE(A) - DT^-$ are called the *the lower and upper approximation* of the specification described by DT, respectively.

Two standard tables $IS_1 = (U_1, A_1)$ and $IS_2 = (U_2, A_2)$ are said to be *satisfying the partial specification DT* if

$$IS_1 \otimes IS_2 \subseteq DT^+$$

and are said to be *consistent with the partial specification DT* if

$$(IS_1 \otimes IS_2) \cap DT^- = \emptyset.$$

The following function defines a *consistency degree* of a standard table IS with respect to a given partial specification DT :

$$Satisfy(IS|DT) = \frac{card(IS \cap DT^+)}{card(IS)}.$$

The *confidence degree* of consistent pair of standard tables (IS_1, IS_2) with respect to DT is defined by

$$Satisfy(IS_1 \otimes IS_2|IS) = \frac{(card(IS_1 \otimes IS_2) \cap IS^+)}{card(IS_1 \otimes IS_2)}.$$

The pair of standard tables (IS_1, IS_2) is satisfying DT if and only if $Satisfy(IS_1 \otimes IS_2|IS) = 1$. The pair of standard tables (IS_1, IS_2) is called *ϵ-consistent* with DT if $Satisfy(IS_1 \otimes IS_2|IS) > 1 - \epsilon$. Properties of the function $Satisfy$ are described by

Theorem 14 *If* $Satisfy(IS_1|DT) > 1 - \epsilon_1$ *and* $Satisfy(IS_2|DT) > 1 - \epsilon_2$ *then*

$$Satisfy(IS_1 \otimes IS_2|DT) > 1 - (\epsilon_1 + \epsilon_2).$$

Let A_1 and A_2 be disjoint subsets of attributes from A and let (IS_1, IS_2) (where $IS_1 = (U_1, A_1|_{U_1})$ and $IS_2 = (U_2, A_2|_{U_2})$) be an ϵ-consistent pair of standard tables with $DT = (U, A, d)$ then one can construct partial task specifications $DT_1 = (\mathbf{U_1}, A_1|_{\mathbf{U_1}}, d_1)$ and $DT_2 = (\mathbf{U_2}, A_2|_{\mathbf{U_2}}, d_2)$ (for sub-agents ag_1 and ag_2) assuming

$$\begin{aligned} \mathbf{U}_1^+ &= U_1; \quad \mathbf{U}_1^- = \{u : (u \otimes U_2) \cap U^- \neq \emptyset\}; \\ \mathbf{U}_2^+ &= U_2; \quad \mathbf{U}_2^- = \{u : (U_1 \otimes u) \cap U^- \neq \emptyset\} \end{aligned}$$

The defined (DT_1, DT_2) pair of specifications is also said to be ϵ-consistent with DT.

A consistent covering of partial specification DT is any set of consistent pairs of standard tables $P = \{(IS_1^i, IS_2^i) : i = 1, 2, ...\}$ satisfying the following condition: $\underline{\underline{DT}} \subset \bigcup_i (IS_1^i \otimes IS_2^i) \subset \overline{\overline{DT}}$, i.e.,

$$\bigcup_i (IS_1^i \otimes IS_2^i) \cap DT^- = \emptyset \text{ and } DT^+ \subset \bigcup_i (IS_1^i \otimes IS_2^i).$$

Now the task decomposition problems for a partial specification DT can be redefined assuming ϵ to be a given parameter.

1. Searching problem for an ϵ-consistent pair of standard tables $IS_1 = (U_1, A_1)$ and $IS_2 = (U_2, A_2)$ with the maximal cardinality of the set $IS_1 \otimes IS_2$.
2. Searching problem for an consistent covering of DT consisting of the minimal number of ϵ-consistent pairs of standard tables among all consistent coverings of DT by ϵ-consistent pairs of standard tables.

Now we will discuss a searching method for ϵ-consistent standard tables from a partial specification according to the definition proposed in the previous section. One can see that the solutions of new decomposition problems can be found by solving some equivalent graph problems. Let the decision table $DT = (U, A, d)$ define a partial specification for the agent ag, and let $A = A_1 \cup A_2$ be the predefined partition of the attribute set A into disjoint subsets. The bipartite graph $G = (V_1 \cup V_1, E)$ for a specification DT is constructed analogously to a graph presented before with following modification: The pair of vertices (v_1, v_2) belongs to E iff $d([Inf_{A_1}(v_1) \otimes Inf_{A_2}(v_2)]) \neq$ '–'. This condition is equivalent to

$$d([Inf_{\mathbf{A_1}}(v_1) \otimes Inf_{A_2}(v_2)]) = \text{'+' or}$$

$$[Inf_{A_1}(v_1) \otimes Inf_{A_2}(v_2)] \notin IS.$$

The edges of a graph are labeled by **0** or **1**. An edge is labeled by **1** if

$$d([Inf_{A_1}(v_1) \otimes Inf_{A_2}(v_2)]) = \text{'+'}$$

and by **0** if

$$[Inf_{A_1}(v_1) \otimes Inf_{A_2}(v_2)] \notin IS$$

The searching problem for an ϵ-consistent pair of standard tables (IS_1, IS_2) with the maximal cardinality of $IS_1 \otimes IS_2$ among ϵ-consistent pairs satisfying $Satisfy(IS_1 \otimes IS_2|DT) > 1 - \epsilon$ can be reduced to the searching problem for a complete sub-graph of $G' = (U_1 \cup U_1, E)$ with the maximal cardinality of $U_1 \times U_2$ and number of edges labeled by **0** not exceeding the value of $\epsilon \cdot |U_1| \cdot |U_2|$. A heuristic for now discussed problem can be constructed by modifying the heuristic discussed previously for the second graph problem.

Now we discuss an example illustrating our method of partial specification decomposition [14]. A car (complex object) assemply room consists of five factories (agents). They are denoted by ag (the main factory), ag_1 (the factory of engines), ag_2 (the factory of bodies), ag_3 (body shop), ag_4 (paint shop) (see Figure 5). Products of agents are described by attributes from

$$A = \{\text{Engine capacity, Body type, Body color}\}$$

Figure 5: The system with five agents. Attributes are denoted by: $a =$"Engine capacity", $b =$"Body type", $c =$"Body color"

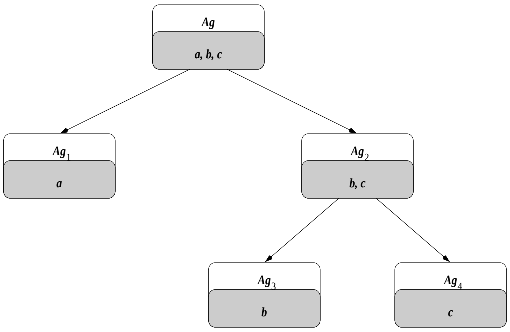

Figure 6: The specification of task for the root agent ag in form of the decision table

DT	**Engine capacity**	**Body type**	**Body color**	**Decision**
u_1	1300	coupe	white	+
u_2	1300	coupe	gray	+
u_3	1300	coupe	green	+
u_4	1300	saloon	white	+
u_5	1300	saloon	red	−
u_6	1300	saloon	gray	+
u_7	1500	coupe	white	+
u_8	1500	coupe	gray	+
u_9	1500	coupe	green	+
u_{10}	1500	saloon	white	+
u_{11}	1500	saloon	gray	+
u_{12}	1500	saloon	black	+
u_{13}	1600	coupe	white	+
u_{14}	1600	coupe	green	+
u_{15}	1600	coupe	gray	+
u_{16}	1600	coupe	red	−
u_{17}	1600	saloon	white	+
u_{18}	1600	saloon	gray	+
u_{19}	1600	saloon	black	+
u_{20}	1600	estate	black	+
u_{21}	1600	estate	white	−
u_{22}	1800	saloon	black	+
u_{23}	1800	saloon	gray	+
u_{24}	1800	saloon	white	−
u_{25}	1800	estate	black	+
u_{26}	1800	estate	white	+
u_{27}	1800	limousine	black	+
u_{28}	1800	limousine	white	−
u_{29}	2000	limousine	black	+
u_{30}	2000	limousine	white	−

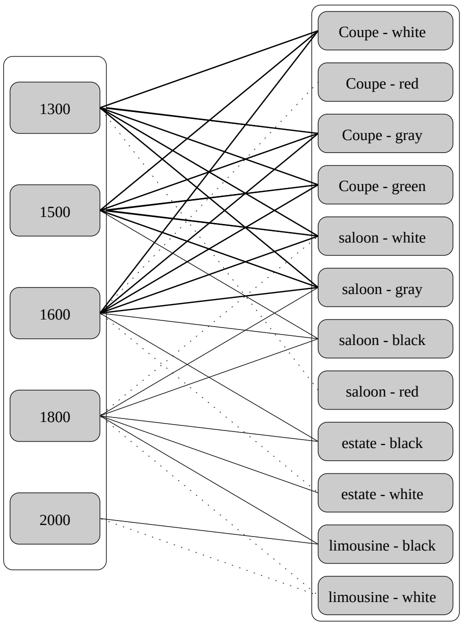

Figure 7: The graph representation of the specification: solid lines – positive examples, dotted lines – negative examples

$\mathbf{A}_1$	**Engine capacity**	d_1
x_1	1300	+
x_2	1500	+
x_3	1600	+
x_4	1800	−

$\otimes$

$\mathbf{A}_2$	**Body type**	**Body colour**	d_2
y_1	coupe	white	+
y_2	coupe	gray	+
y_3	coupe	green	+
y_4	saloon	white	+
y_5	saloon	gray	+
y_6	coupe	red	−
y_7	saloon	red	−
y_8	estate	white	−

$\mathbf{A}_3$	**Body type**	d_3
z_1	coupe	+
z_2	saloon	+
z_3	estate	−

$\otimes$

$\mathbf{A}_4$	**Body colour**	d_4
t_1	white	+
t_2	gray	+
t_3	red	−

Figure 8: The maximal pairs of standard tables $(DT_1,\ DT_2)$, $(DT_3,\ DT_4)$ found for the root agent and for the agent ag_2, respectively

.

We assume that the root agent ag (the main factory) has obtained the partial specification of the cars to be assemblied in the form of the decision table presented in Figure 6. The graph interpretation of the specification is represented in Figure 7.

Applying our algorithm for the root agent ag we obtain the maximal pair of decision tables DT_1 and DT_2 presented in Figure 8. Consequently, the table DT_2 can be decomposed into a pair of standard tables DT_3 and DT_4 (see Figure 8).

7.3 Decomposition Method Based on Decision Rules

In this section we propose another decomposition method using decision rules as a generalized knowledge extracted from decision table.

Let the decision table $DT = (U, A, d)$ define a partial specification for the agent ag, and let $A = A_1 \cup A_2$ be the predefined partition of the attribute set A into disjoint subsets. Let $Rule\_Set_{IS} = Rule\_Set_{+} \cup Rule\_Set_{-}$ be the set of decision rules for the table IS (with minimal number of descriptors) where $Rule\_Set_{+}$ is a set of positive rules and $Rule\_Set_{-}$ is a set of negative rules. We assume that $Rule\_Set_{+} = \{R_1, ..., R_m\}$ and any decision rule R_i has a form

$$R_i : D_i(A_1) \wedge D_i(A_2) \rightarrow (d = +)$$

where $D_i(A_1)$ is a conjunction of descriptors consisting of attributes from A_1 and $D_i(A_2)$ is a conjunction of descriptors containing attributes from A_2.

One can construct a new labelled bipartite graph $\Gamma(Rule\_Set_{+}) = \langle V_1 \cup V_2, E \rangle$ from the set of positive decision rules $Rule\_Set_{+}$ as follows:

$$\begin{aligned} V_1 &= \{D_i(A_1) : i \in \{1, .., m\}\} \text{ and } V_2 = \{D_i(A_2) : i \in \{1, .., m\}\} \\ E &= \{(D_i(A_1), D_j(A_2)) : i, j \in \{1, .., m\} \text{ and} \\ &\quad D_i(A_1) \wedge D_j(A_2) \rightarrow (d = +) \text{ is a consistent decision rule } \} \end{aligned}$$

Any edge $e = (D_i(A_1), D_j(A_2))$ is labelled by the number of objects from decision table DT supporting the decision rule "**if** $D_i(A_1) \wedge D_j(A_2)$ **then** $(d = +)$". This number is called *the weight of the edge* e and denoted by $w(e)$.

Hence, the task decomposition problem can be reduced to the searching problem for a complete bipartite sub-graph of $\Gamma(Rule\_Set_{+})$ with maximal sum of edge weights.

It is easy to modify the approximate algorithm presented before to solve this problem. Let us assume that the pair (V_1', V_2') of vertices is a solution of the graph problem for $\Gamma(Rule\_Set_{+})$, where $V_1' = \{D_{i_1}(A_1), D_{i_2}(A_1), ...\}$ and $V_2' = \{D_{j_1}(A_2), D_{j_2}(A_2), ...\}$ then (V_1', V_2') establishes a pair of task specifications for sub-agents.

7.4 Decomposition of Approximate Specification

The notion of *approximate specification* has been introduced in [16] as a new reasoning approach for agents under uncertainty. Assuming that every agent ag is equipped with a function Sim_{ag} assigning for any object x and any specification Φ, a degree $Sim_{ag}(x, \Phi) \in [0, 1]$ of satisfying Φ by x. Then, the approximate specification for the agent ag is defined by any pair (Φ, ϵ) denoting the ϵ-neighborhood of specification Φ, i.e., the set

$$\Theta_{ag}(\Phi, \epsilon) = \{x : Sim_{ag}(x, \Phi) > 1 - \epsilon\}.$$

The inference rules of agent ag whose children are ag_1, ..., ag_k are in the form

$$\textbf{if } (x_1 \in \Theta_{ag_1}(\Phi_1, \epsilon_1)) \wedge \ldots \wedge (x_k \in \Theta_{ag_k}(\Phi_k, \epsilon_k)) \textbf{ then } (x \in \Theta_{ag}(\Phi, \epsilon))$$

where $x_1, ..., x_k$ are objects submitted by $ag_1, ..., ag_k$, respectively, and x is the object composed by ag from $x_1, ..., x_k$.

We will now discuss relationships between ϵ and $\epsilon_1, ..., \epsilon_k$ for some well known functions Sim_{ag} in case of composition rule $\otimes$. For simplicity, we consider a "binary" agent system, in which every father-agent has two children ag_1 and ag_2. Let the specification task be defined by an information system $IS = (U, A)$ for the root-agent ag. Below we present some important functions defining a degree in which any object x is satisfying a specification IS. We start with some basic *distance* functions $d : UNIVERSE(A) \times UNIVERSE(A) \rightarrow \mathbb{R}^+ \cup \{0\}$ defined by

$$d_1(x, y) = card(\{a \in A : a(x) \neq a(y)\}) \qquad \textit{(the Hamming distance)}$$

$$d_2(x, y) = \max_{a \in A}\{|a(x) - a(y)|\} \qquad \textit{(the } \infty\textit{-Norm)}$$

$$d_3(x, y) = \sqrt{\sum_{a \in A}(a(x) - a(y))^2} \qquad \textit{(the Euclidean distance)}$$

We use the following normalized distance functions:

1. *Normalized Hamming distance*
$$d_H(x, y) = \frac{d_1(x, y)}{card(A)}$$

2. *Normalized ∞-Norm*
$$d_N(x, y) = \max_{a \in A} \frac{|a(x) - a(y)|}{a_{max} - a_{min}}.$$

3. *Normalized Euclidean distance*
$$d_S(x, y) = \frac{\sqrt{\sum_{a \in A}\left(\frac{a(x) - a(y)}{a_{max} - a_{min}}\right)^2}}{card(A)}$$

One can see that distance functions are defined for the space $UNIVERSE(A)$, where every object is characterized by all attributes in A. For any subset $B \subseteq A$ the distance functions can be modified by restricting the attribute set to B. We denote the distance function d restricted to B by $d|_B$. Having the distance function d the distance of object x to the set S is defined by $d(x, S) = min_{y \in S} d(x, y)$. Let $IS = (U, A)$ be a specification for an agent ag. Hence, the approximate specification for ag is defined by $\Theta(IS, \epsilon) = \{x : d|_A(x, U) < \epsilon\}$. Let $IS_1 = (U_1, A_1)$ and $IS_2 = (U_2, A_2)$ be the specifications for ag_1 and ag_2, respectively. Let $\Theta(IS_1, \epsilon_1)$ and $\Theta(IS_2, \epsilon_2)$ be the approximate specification defined by IS_1 and IS_2, respectively. The following observations show the relationship between $\Theta(IS_1, \epsilon_1)$, $\Theta(IS_2, \epsilon_2)$ and the composed approximate specification $\Theta(IS, \epsilon)$. For a given distance function d, we have the following

Theorem 15 *If $\Theta(IS_1, \epsilon_1)$ and $\Theta(IS_2, \epsilon_2)$ are approximate specifications for sub-agents ag_1 and ag_2, respectively then we have*

$$\Theta(IS_1, \epsilon_1) \otimes \Theta(IS_2, \epsilon_2) \subseteq \Theta(IS_1 \otimes IS_2, \epsilon)$$

where

1. $\epsilon = max(\epsilon_1, \epsilon_2)$ if d is the Hamming distance.

2. $\epsilon = max(\epsilon_1, \epsilon_2)$ if d is the ∞-Norm

3. $\epsilon = \epsilon_1 + \epsilon_2$ if d is the Euclidean distance.

The proof follows from the definitions of distance functions and the $\otimes$ operator.

Conclusions

Our approach can be viewed as a step towards the understanding of complex information granules and their role in many application areas like spatial reasoning or data mining and knowledge discovery. Methods for synthesis of complex information granules in distributed environment are important in these areas because of need for approximate fusion (merging) of information from different sources of information. We have discussed information granule syntax and semantics as well their inclusion and closeness. Several examples of information granules have been presented. An approach for synthesis and analysis of complex information granules in distributed environment has been outlined. It was emphasized the necessity of robust information granules synthesis. The robustness guarantees that a properly constructed scheme of reasoning on prototypes (standards) leads to conclusion that a given specification is also satisfied (in satisfactory degree) when the standards are substituted by sufficiently small deviations of them.

We have also investigated a decomposition problem of complex task in a multi-agent system having its specification. Two decomposition problems are discussed. The fist one concerns the decomposing such tasks under a complete specification. The second one is related to tasks with a partial specification. Both problems can be solved by transforming them to corresponding graph problems. For some predefined composition rules the proposed decomposition methods guarantee a low error rate of synthesis process.

The presented approach seems to be important for development of several applications mentioned in [3].

Finally, we would like to mention some further research directions (several others are listed in [27]):

- investigations of operations on approximation spaces corresponding to different levels of information granule construction;
- investigations of approximate schemes of reasoning (schemes of complex granule construction) matching schemes of reasoning based on linguistic variables.

Acknowledgement *This work has been prepared under the Research Grant No 8T11C 025 19 from the State Committee for Scientific Research (KBN) of the Republic of Poland and by the grant from the Wallenberg Foundation.*

References

[1] Ågotnes T., Komorowski J., Loken T.: Taming Large Rule Models in Rough Set Approaches, 3rd European Conference of Principles and Practice of Knowledge Discovery in Databases, September 15-18, 1999, Prague, Czech Republic, Lecture Notes in Artificial Intelligence 1704, 1999, 193–203.

[2] Agrawal R., Mannila H., Srikant R., Toivonen H., Verkano A.: Fast Discovery of Association Rules, in: Fayyad U.M., Piatetsky-Shapiro G., Smyth P., Uthurusamy R. (Eds.): Advances in Knowledge Discovery and Data Mining, The AAAI Press/The MIT Press 1996, 307–328.

[3] Amarel S.: PANEL on AI and Design, in: J. Mylopoulos and R. Reiter (Eds.), Proceedings Twelfth International Conference on Artificial Intelligence, Sydney, Australia, 1991, 563–565.

[4] Böhlen M. H., Jensen C. S., Scholl M. O.: Spatio–Temporal Database Management, Lecture Notes in Computer Science 1678, Springer Verlag, Berlin 1999.

[5] Cattaneo G.: Abstract Approximation Spaces for Rough Theories, L. Polkowski, A. Skowron (Eds.), Rough Sets in Knowledge Discovery 1. Methodology and Applications, Physica–Verlag, Heidelberg, 1998, 59–98.

[6] Dietterich T. G.: (2000). Ensemble Methods in Machine Learning, in: J. Kittler and F. Roli (Eds.) First International Workshop on Multiple Classifier Systems, Lecture Notes in Computer Science, Springer Verlag, New York 2000, 1-15.

[7] Escrig M.T., Toledo F.: Qualitative Spatial Reasoning: Theory and Practice, IOS Press, Amsterdam, 1998.

[8] Fayyad U.M., Piatetsky-Shapiro G., Smyth P., Uthurusamy R. (Eds.): Advances in Knowledge Discovery and Data Mining, The AAAI Press/The MIT Press, Cambridge 1996.

[9] Huhns M.N., Singh M.P. (Eds.): Readings in Agents, Morgan Kaufmann, San Mateo 1998.

[10] Kacprzyk J., Zadrozny S.: The paradigm of computing with words in intelligent database querying, In: [39], 383-398.

[11] Komorowski J., Pawlak Z., Polkowski L., Skowron A.: Rough Sets: A Tutorial, S.K. Pal, A.Skowron (Eds.), Rough-Fuzzy Hybridization: A New Trend in Decision Making, Springer-Verlag, Singapore, 1999, 3–98.

[12] Lin T.Y.: Granular Computing on Binary Relations I. Data Mining and Neighborhood Systems, in: L. Polkowski, A. Skowron (Eds.), Rough Sets in Knowledge Discovery 1. Methodology and Applications, Physica–Verlag, Heidelberg 1998, 107–121.

[13] Michell, T.M.: Machine Learning. Mc Graw-Hill, Portland, 1997.

[14] Nguyen H.S., Nguyen, S.H., Skowron A.: Decomposition of Task Specification. Proceedings of the 11th International Symposium on Foundations of Intelligent Systems, June 1999, Warsaw, Poland, Lecture Notes in Computer Science **1609**, Springer-Verlag, Berlin 1999, 310–318.

[15] Pawlak Z.: Rough Sets. Theoretical Aspects of Reasoning about Data, Kluwer Academic Publishers, Dordrecht 1991.

[16] Polkowski L., Skowron A.: Rough Mereology: A New Paradigm for Approximate Reasoning, International Journal of Approximate Reasoning **15(4)**, 1996, 333–365.

[17] Polkowski L., Skowron A.: Rough Mereological Foundations for Design, Analysis, Synthesis, and Control in Distributed Systems, Information Sciences An International Journal **104/1-2**, Elsevier Science, New York 1998, 129–156.

[18] Polkowski L., Skowron A.: Towards Adaptive Calculus of Granules, Zadeh L.A., Kacprzyk J. (Eds.): Computing with Words in Information/Intelligent Systems 1. Foundations, Physica-Verlag, Heidelberg, 1999.

[19] Polkowski L., Skowron A.: Computational Intelligence, 2001 (this issue).

[20] Renz J., Nebel B.: Efficient Methods for Qualitative Spatial Reasoning. H. Prade (Ed.), Proceedings of the 13th European Conference on Artificial Intelligence (ECAI'98), Wiley, New York 1998, 562–566.

[21] Roddick J.F., Spiliopoulou M.: A Bibliography of Temporal, Spatial, and Temporal Data Mining Research, Newsletter of the Special Interest Group (SIG) on Knowledge Discovery & Data Mining **1(1)** 1999, 34–38, http://research.microsoft.com/datamine/sigkdd/.

[22] Sriram D., Logcher R., Fukuda S.: *Computer - Aided Cooperative Product Development* Lecture Notes inComputer Science **492**, Springer-Verlag, Berlin 1991.

[23] Skowron A., Stepaniuk J.: Tolerance Approximation Spaces, Fundamenta Informaticae **27**, 1996, 245–253.

[24] Skowron A., Stepaniuk J.: Information Granules and Approximation Spaces, Proceedings of the 7th International Conference on Information Processing and Management of Uncertainty in Knowledge-based Systems (IPMU'98), Paris, France, July 6-10, 1998, 1354–1361.

[25] Skowron A., Stepaniuk J.: Information Granules in Distributed Environment, New Directions in Rough Sets, Data Mining, and Granular-Soft Computing, Yamaguchi, Japan, November 9–11, Lecture Notes in Artificial Intelligence **1711**, Springer-Verlag, Berlin 1999, 357–365.

[26] Skowron A., Stepaniuk J., Tsumoto S.: Information Granules for Spatial Reasoning, Bulletin of International Rough Set Society **3/4** 1999, 147–154.

[27] Skowron A., Stepaniuk J.: Information Granules: Towards Foundations of Granular Computing, International Journal of Intelligent Systems **16/1** 2001, 57–86.

[28] Sogo T., Ishiguro H., Ishida T.: Acquisition of Qualitative Spatial Representation by Visual Observation, IJCAI 1999, 1054–1060.

[29] WWW SPACENET page: http://agora.scs.leeds.ac.uk/spacenet/.

[30] Stepaniuk J.: Knowledge Discovery by Application of Rough Set Models, ICS PAS Report 887, Institute of Computer Science, Polish Academy of Sciences, Warsaw 1999, and also: L. Polkowski (ed.), Rough Sets: New Developments, Applications, Case Studies and Software Systems, Physica–Verlag, Heidelberg, 2000 (in print).

[31] Stone, P: Layered Learning in Multiagent Systems: A Winning Approach to Robotic Soccer, MIT Press, Cambridge 2000.

[32] WITAS project web page: http://www.ida.liu.se/ext/witas/eng.html.

[33] Yao Y.Y.: Granular computing using neighborhood systems, Advances in Soft Computing: Engineering Design and Manufacturing, The 3rd on-line World Conference on Soft Computing (WSC3), June 21-30, 1998, R. Roy, T. Furuhashi, P.K. Chawdhry (Eds.), Springer-Verlag, London 1999, 539-553.

[34] Yao Y.Y.: Granular computing: basic issues and possible solutions, Proceedings of the 5th Joint Conference on Information Sciences, Volume 1, Atlantic City, New Yersey, USA, February 27-March 3, 2000, P. P. Wang (Ed.), Association for Intelligent Machinery, 186-189.

[35] Yao Y.Y., Zhong N.: Potential applications of granular computing in knowledge discovery and data mining, Proceedings of World Multiconference on Systemics, Cybernetics and Informatics, Vol. 5, Computer Science and Engineering, Orlando, Florida, USA, July 31-August 4, 1999, M. Torres, B. Sanchez, J. Aguilar (Eds.), International Institute of Informatics and Systematics, 573-580.

[36] Zadeh L.A.: Fuzzy Logic = Computing with Words, IEEE Trans. on Fuzzy Systems **4**, 1996, 103–111.

[37] Zadeh L.A.: Toward a Theory of Fuzzy Information Granulation and Its Certainty in Human Reasoning and Fuzzy Logic, Fuzzy Sets and Systems **90**, 1997, 111–127.

[38] Zadeh L.A., Kacprzyk J. (Eds.): Computing with Words in Information/Intelligent Systems 1. Foundations, Physica-Verlag, Heidelberg, 1999.

[39] Zadeh L.A., Kacprzyk J. (Eds.): Computing with Words in Information/Intelligent Systems 2. Applications, Physica-Verlag, Heidelberg, 1999.