Rough-Granular Computing in Human-Centric Information Processing

Andrzej Jankowski and Andrzej Skowron

Abstract. In the area of ubiquitous computing, users will continuously interact with computing devices by suggesting strategies, hypothesis, by communicating some new facts from domain knowledge, explaining untypical cases in dialogs with devices (agents), etc. Hence, compound vague concepts used by humans should be understandable, at least in an approximate sense, by these devices. We discuss some basic issues of interactive computations in the framework of rough-granular computing for approximation of complex concepts. Among these issues are hierarchical modeling of granule structures and interactions between granules of different complexity. Interactions between granules on which computations are performed are among the fundamental concepts of Wisdom Technology (Wistech). Wistech is encompassing such areas as interactive computations, multiagent systems, cognitive computation, natural computing, complex adaptive and autonomous systems, or knowledge representation and reasoning about knowledge. We outline current results on approximation of compound vague concepts which are based on rough-granular computing. In particular, hierarchical methods are used for approximation of domain ontologies of vague concepts. The developed methodology has been tested on real-life problems related to such areas as unmanned area vehicle control, robotics, predicting of risk patterns from temporal medical and financial data, sun spot classification, bioinformatics.

Keywords: rough sets, granular computing, rough-granular computing, judgment, interaction, wisdom technology (Wistech).

Andrzej Jankowski
Institute of Decision Processes Support and AdgaM Solutions Sp. z o.o.
Wąwozowa 9/64, 02-796 Warsaw, Poland
e-mail: `andrzejj@adgam.com.pl`

Andrzej Skowron
Institute of Mathematics, The University of Warsaw,
Banacha 2, 02-097 Warsaw, Poland
e-mail: `skowron@mimuw.edu.pl`

K.A. Cyran et al. (Eds.): Man-Machine Interactions, AISC 59, pp. 23–42.
springerlink.com

1 Introduction

The radical changes in Knowledge Technology depend on the further advancement of technology to acquire, represent, store, process, discover, communicate and learn wisdom. We call this technology *wisdom technology* (or Wistech, for short). The term *wisdom* commonly means *rightly judging*. This common notion can be refined. By *wisdom*, we understand an adaptive ability to make judgments correctly to a satisfactory degree (in particular, correct decisions) having in mind real-life constraints. The intuitive nature of wisdom understood in this way can be expressed by the so called *wisdom equation* [18], metaphorically shown as follows.

$$wisdom = knowledge + adaptive\ judgment + interactions. \quad (1)$$

It is worthwhile mentioning that the wisdom concept was intensively discussed by many famous philosophers starting form ancient times. For example, in [58] one can find the following sentences:

Aristotle's man of practical wisdom, the phronimos, does not ignore rules and models, or dispense justice without criteria. He is observant of principles and, at the same time, open to their modification. He begins with nomoi – established law – and employs practical wisdom to determine how it should be applied in particular situations and when departures are warranted. Rules provide the guideposts for inquiry and critical reflection.

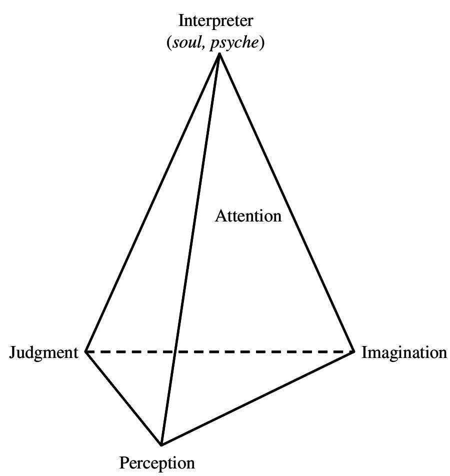
Wisdom can be treated as a special type of knowledge processing. In order to explain the specificity of this type of knowledge processing, let us assume that a control system of a given agent *Ag* consists of a society of agent control components interacting with the other agent *Ag* components and with the agent *Ag* environments. Moreover, there are special agent components, called as the agent coordination control components which are responsible for the coordination of control components. Any agent coordination control component mainly searches for answers for the following question: *What to do next*? or, more precisely: *Which of the agent's* Ag *control components should be activated now*? Of course, any agent control component has to process some kind of knowledge representation. In the context of agent perception, the agent *Ag* itself (by using, e.g., interactions, memory, and coordination among control components) is processing a very special type of knowledge reflecting the agent perception of the hierarchy of needs (objectives, plans, etc.) and the current agent or the environment constraints. This kind of knowledge processing mainly deals with complex vague concepts (such as risk or safety) from the point of view of the *selfish* agent needs. Usually, this kind of knowledge processing is not necessarily logical reasoning in terms of proving statements (i.e., labeling statements by truth values such as TRUE or FALSE). This knowledge processing is rather analogous to the judgment process in a court aiming at recognition of evidence which could be used as an argument *for* or *against*. Arguments *for* or *against* are used in order to make the final decision which one of the solutions is the best for the agent in the current situation (i.e., arguments are labeling statements by judgment values expressing the action priorities). The evaluation of currents needs by agent *Ag* is realized from the point of view of hierarchy of agent *Ag life* values/needs). Wisdom type of knowledge processing by the agent *Ag* is characterized by the ability to improve quality of the judgment process based on the agent *Ag* experiences. In order to emphasize the importance of this ability, we use the concept of *adaptive judgment* in the wisdom equation instead of just *judgment*. An agent who is able to perform adaptive judgment in the above sense, we simply call as a *judge*.

The adaptivity aspects are also crucial from the point of view of interactions [17, 27, 33, 54]. The need for adaptation follows, e.g., from the fact that complex vague concepts on the basis of which the judgment is performed by the agent *Ag* are approximated by classification algorithms (classifiers) which are very often drifting in time following changes in data and represented knowledge.

An important aspect of Wistech is that the complexity and uncertainty of real-life constraints mean that in practice we must reconcile ourselves to the fact that our judgments are based on non-crisp concepts (i.e., concepts with borderline cases) and also do not take into account all the knowledge accumulated and available to us. This is why our judgments are usually imperfect. But as a consolation, we also learn to improve the quality of our judgments via observation and analysis of our experience during interaction with the environment. Satisfactory decision-making levels can be achieved as a result of improved judgments.

Thus wisdom is directly responsible for the focusing of an agents attention (see Aristotle tetrahedron in Fig. 1) on problems and techniques of their solution which are important in terms of the agent judgment mechanism. This mechanism is based on the Maslow hierarchy of needs (see Fig. 2) and agent perception of ongoing interactions with other agents and environments. In particular, the agent's wisdom can be treated, as the control at the highest level of hierarchy of the agent's actions and reactions and is based on concept processing in the metaphoric Aristotle tetrahedron (Fig. 1). One can use the following conceptual simplification of agent wisdom. Agent wisdom is an efficient and an on-line agent judgment mechanism making it

Fig. 1 Relationships between imagination, judgment, perception and psyche



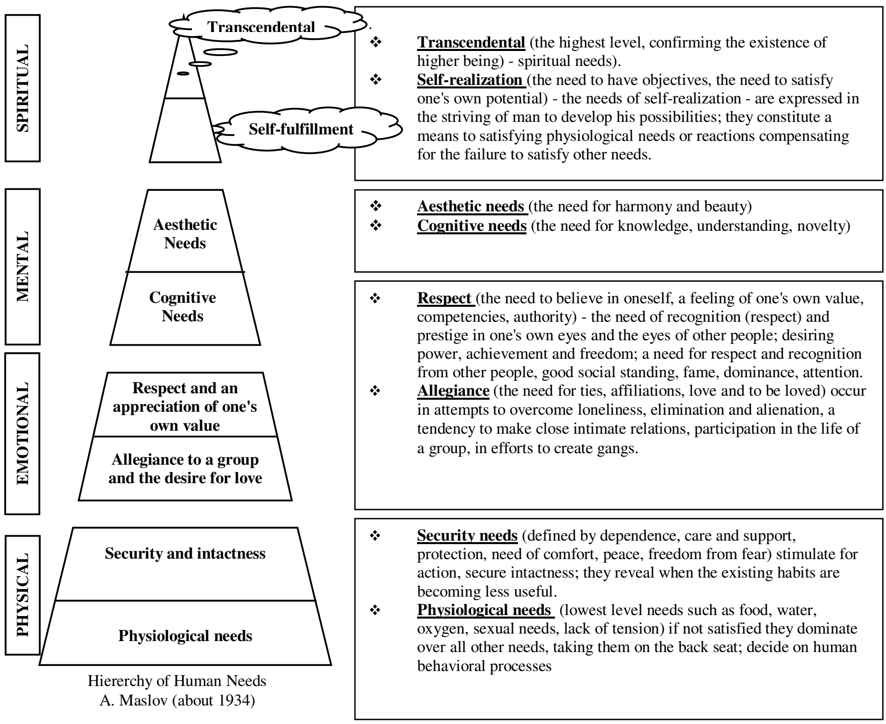


Fig. 2 The Maslov Hierarchy of human needs (about 1934) as an example of judge hierarchy of habit controls

possible for agent to answer the following questions: (i) How to currently construct the most important priority list of problems to be solved? (ii) How to solve the top priority problems under real life constraints? (iii) What to do next?

One of the main barriers hindering an acceleration in the development of Wistech applications lies in developing satisfactory computational models implementing the functioning of *adaptive judgment*. This difficulty primarily consists in overcoming the complexity of integrating the local assimilation and processing of changing non-crisp and incompletely specified concepts necessary to make correct judgments. In other words, we are only able to model tested phenomena using local (subjective) models and interactions between them. In practical applications, usually, we are not able to give perfect global models of analyzed phenomena. However, we can only approximate global models by integrating the various incomplete perspectives of problem perception.

Wisdom techniques include approximate reasoning by agents or teams of agents about vague concepts concerning real-life dynamically changing, usually distributed, systems in which these agents are operating. Such systems consist of other autonomous agents operating in highly unpredictable environments and interacting with each other.

Wistech is based on techniques of reasoning about knowledge, information and data which helps apply the current knowledge in problem solving in real-life highly unpredictable environments and autonomous multiagent systems. This includes such methods as identification of the current situation on the basis of interactions or dialogs, extraction of relevant fragments of knowledge from knowledge networks, judgment for prediction for relevant actions or plans in the current situation, or judgment of the current plan reconfiguration.

In [18, 19, 20, 21] Wisdom Technology (Wistech) is discussed as one of the main paradigms for development of new applications in intelligent systems.

Gottfried Wilhelm Leibniz should be considered a precursor of modern *Granular Computing* (GC) understood as a calculus of human thoughts [23, 24, 20, 21]. Through centuries mathematicians have been developing tools to deal with such a calculus. Unfortunately, the tools developed in *crisp* mathematics, in particular, in classical mathematical logic do not yet allow for the understanding natural language used by humans to express thoughts and reasoning about theses thoughts, an understanding which will allow us to construct truly intelligent systems.

One of the reasons is that humans, capable of efficiently solving many real-life problems, are able to express their thoughts by means of vague, uncertain, imprecise concepts and reason with such concepts. Lotfi Zadeh proposed to base the calculus of thoughts using fuzzy logic to move from computing with numbers to computing with words, from manipulations of measurements to manipulations of perceptions, and further to Granular Computing. This idea has been developed by Lotfi Zadeh himself in a number of papers (see, e.g., [63, 64, 65, 62, 67]) and by other researchers, also using rough set methods (see, e.g., [35, 42]).

Solving complex problems, e.g., by multi-agent systems requires new approximate reasoning methods based on new computing paradigms. One such recently emerging computing paradigm is *Rough Granular Computing* RGC (see, e.g., [42]).

The research on the foundations on RGC is based on the rough set approach. The rough set concept, due to Pawlak [38, 39, 41] is based on classical two valued logic. The rough set approach has been developed to deal with uncertainty and vagueness. The approach makes it possible to reason about the approximations of vague concepts. These approximations are temporary, subjective, and change adaptively with changes in environments [11, 47, 49].

In this paper, we discuss some basic issues on RGC emphasizing the role of hierarchical modeling of granular structures (Sections 2-6) and, in particular, some issues on interactive granular computations (Sect. 6).

2 Granular Computing and Rough-Granular Computing in Hierarchical Learning

The hierarchical learning approach takes advantage of additional domain knowledge provided by human experts. In order to best employ this knowledge, it relies on the observation that human thinking and perception in general, and their reasoning while performing classification tasks in particular, can:

- inherently comprise different levels of abstraction,
- display a natural ability to switch focus from one level to another,
- operate on several levels simultaneously.

Such processes are natural subjects for the GC paradigm, which encompasses theories, methods, techniques and tools for such fields as problem solving, information processing, human perception evaluation, analysis of complex systems and many others.

The concept of information granules is closely related to the imprecise nature of human reasoning and perception. GC therefore provides excellent tools and methodologies for problems involving flexible operations on imprecise or approximated concepts expressed in natural language.

One of the possible approaches in developing methods for compound concept approximations can be based on the layered (hierarchical) learning [55, 12]. Inducing concept approximation should be developed hierarchically starting from concepts that can be directly approximated using sensor measurements toward compound target concepts related to perception. This general idea can be realized using additional domain knowledge represented in natural language. For example, one can use some rules of behavior on the roads, expressed in natural language, to assess from recordings (made, e.g., by camera and other sensors) of actual traffic situations, if a particular situation is safe or not (see, e.g., [29, 9, 8, 14]). The hierarchical learning has been also used for identification of risk patterns in medical data and extended for therapy planning (see, e.g. [7, 6]). Another application of hierarchical learning for sunspot classification is reported in [32]. To deal with such problems one should develop methods for concept approximations together with methods aiming at approximation of reasoning schemes (over such concepts) expressed in natural language. The foundations of such an approach, called Rough-Granular Computing, creating a core of perception logic, are based on rough set theory [39, 41, 14] and its extension rough mereology [44, 48, 35]. The (approximate) Boolean reasoning methods can be scaled to the case of compound concept approximation.

RGC is an approach to the constructive definition of computations over objects, called granules, aiming at searching for solutions of problems which are specified using vague concepts. Computations in RGC are performed on granules representing often vague, partially specified, and compound concepts delivered by agents engaged in tasks such as knowledge representation, communication with other agents, and reasoning. Granules are obtained through the process of granulation (degranulation). Granulation can be viewed as a human way of achieving data compression and it plays a key role in implementing the divide-and-conquer strategy in human problem-solving [64, 67]. The approach combines rough set methods with other soft computing methods, and methods based on granular computing. RGC is used for developing one of the possible Wistech foundations based on approximate reasoning using vague concepts. The RGC approach combines rough set methods with methods based on granular computing [2, 42, 67], borrowing also from other soft computing paradigms.

Let us observe that hierarchical modeling employs some general mechanisms emphasized in [19] dealing with a kind of ‘interplay’ between syntax and semantics. The key observation is that the syntax on one level is used to define semantical structures (or their clusters) on the next level of hierarchy. One can interpret them in the framework of the Bairwise classifications [3] as operations on such classifications or as a kind of sums of information systems [50]. They allow us gradually to model structures of granules representing *wider* context of perceived objects. In this way, it is possible to construct more compound granules interpreted, e.g., as patterns representing properties of, e.g., time windows of states, sequences of such time windows, sets of such sequences, etc.

3 Hierarchical Modeling of Granule Structures

Modeling relevant granules such as patterns, approximation spaces, clusters or classifiers starts from relational structures corresponding to their attributes. On can distinguish two kinds of attributes. The attributes of the first kind are like sensors, their values are obtained as the result of interaction of (the agent possessing them) with the environment. The attribute of the second kind are defined over already defined attributes. For any attribute (feature) a we consider a relational structure $\mathscr{R}_a = (V_a, \{r_i\}_{i\in I})$, where V_a is a set of values of the attribute a. Examples of such relational structures defined over the attribute-value set V_a are: $(V_a, =)$, $(V_a, \leq)$, where $\leq$ is a linear order on V_a, or $(V_a, \leq, +, \cdot, 0, 1)$, where $V_a = \mathbb{R}$ and $\mathbb{R}$ is the set of reals. Certainly, V_a may consists complex values, e.g., relational structures. By L_a we denote a set of formulas interpreted over $\mathscr{R}_a$ as subsets of V_a. It means that if $\alpha \in L_a$ then its semantics (an object corresponding to its meaning) $\|\alpha\|_{\mathscr{R}_a}$ is a subset of V_a. Let us note that one can define attributes by sets of formulas. To explain this idea, we assume that $\mathscr{F}_a \subseteq L_a$ is a set of formulas satisfying the following two conditions: (i) for any $v \in V_a$ there exits $\alpha \in \mathscr{F}_a$ true for v; and (ii) for any two different formulas from $\mathscr{F}_a$ their conjunction is false for any $v \in V_a$. Hence, for any $v \in V_a$ there is a unique formula in $\mathscr{F}_a$ true on v. One can consider an example of discretization of $\mathbb{R}$ by formulas $\alpha_1, \ldots, \alpha_k$ with interpretation over $\mathscr{R}_a = (\mathbb{R}, \leq, +, \cdot, 0, 1)$, where $\|\alpha_i\|_{\mathscr{R}_a}$ for $i = 1, \ldots, k$ create a partition of $\mathbb{R}$ into intervals.

If $\mathscr{A} = (U, A)$ is an information system and $a \in A$ then $\|\alpha\|_{\mathscr{R}_a}$ can be used to define semantics of α over $\mathscr{A}$ by assuming $\|\alpha\|_{\mathscr{A}} = \{x \in U : \ a(x) \in \|\alpha\|_{\mathscr{R}_a}\}$. Hence, any formula α can be treated as a new binary attribute of objects from U (see Fig. 3). If $\mathscr{A}^* = (U^*, A^*)$ is an extension of $\mathscr{A} = (U, A)$, i.e., $U \subseteq U^*$, $A^* = \{a^* : \ a \in A\}$, and $a^*(x) = a(x)$ for $x \in U$, then $\|\alpha\|_{\mathscr{A}} \subseteq \|\alpha\|_{\mathscr{A}^*}$.

In the next step of modeling, relational structures corresponding to attributes can be fused. Let us consider an illustrative example. We assume $\mathscr{R}_{a_i} = (V_{a_i}, r_{\mathscr{R}_{a_i}})$ are relational structures with binary relation $r_{\mathscr{R}_{a_i}}$ for $i = 1, \ldots, k$. Then, by $\mathscr{R}_{a_1} \times \cdots \times \mathscr{R}_{a_k}$ we denote their fusion defined by a relational structure over $(V_{a_1} \times \cdots \times V_{a_k})^2$ consisting of relation $r \subseteq (V_{a_1} \times \cdots \times V_{a_k})^2$ such that for any $(v_1, \ldots, v_k)$, $(v'_1, \ldots, v'_k) \in V_{a_1} \times \cdots \times V_{a_k}$ we have $(v_1, \ldots, v_k) r (v'_1, \ldots, v'_k)$ if and only if $v_i r_{\mathscr{R}_{a_i}} v'_i$ for $i = 1, \ldots, k$. One can extend this example by imposing some additional constraints.

In the process of searching for (sub-)optimal approximation spaces, different strategies may be used. Let us consider an example of such strategy presented in [53]. In this example, $DT = (U, A, d)$ denotes a decision system (a given sample of

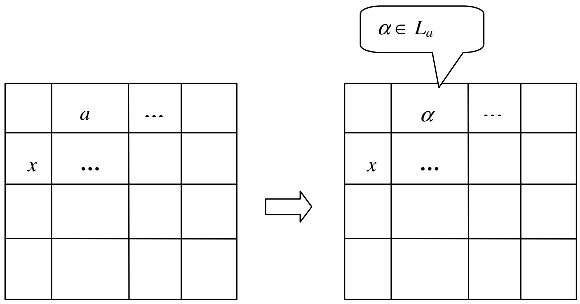


Fig. 3 New attribute defined by a formula α from L_a

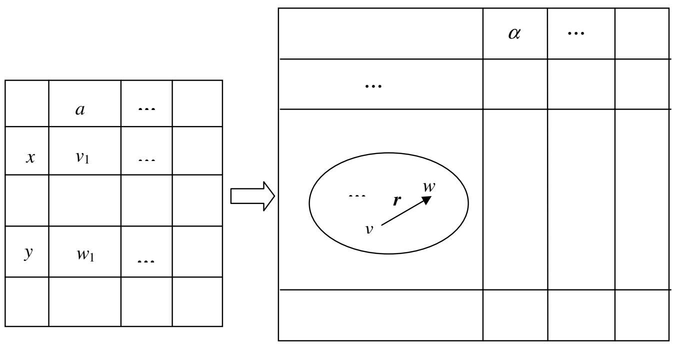


Fig. 4 Granulation to tolerance classes. r is a similarity (tolerance) relation defined over signatures of objects

data), where U is a set of objects, A is a set of attributes and d is a decision. We assume that for any object $x \in U$, only partial information equal to the A-signature of x (object signature, for short) is accessible, i.e., $Inf_A(x) = \{(a, a(x)) : a \in A\}$. Analogously, for any concept we are only given a partial information about this concept by means of a sample of objects, e.g., in the form of decision table. One can use object signatures as new objects in a new relational structure $\mathscr{R}$. In this relational structure $\mathscr{R}$ some relations between object signatures are also modeled, e.g., defined by the similarities of these object signatures (see Fig. 4).

Discovery of relevant relations between object signatures is an important step in searching for relevant approximation spaces. In this way, a class of relational structures representing perception of objects and their parts is constructed. In the next step, we select a language $\mathscr{L}$ consisting of formulas expressing properties over the defined relational structures and we search for relevant formulas in $\mathscr{L}$. The semantics of formulas (e.g., with one free variable) from $\mathscr{L}$ are subsets of object signatures. Note, that each object signature defines a neighborhood of objects from a given sample (e.g., decision table DT) and another set on the whole universe of objects being an extension of U. Thus, each formula from $\mathscr{L}$ defines a family of sets of objects over the sample and also another family of sets over the universe of all

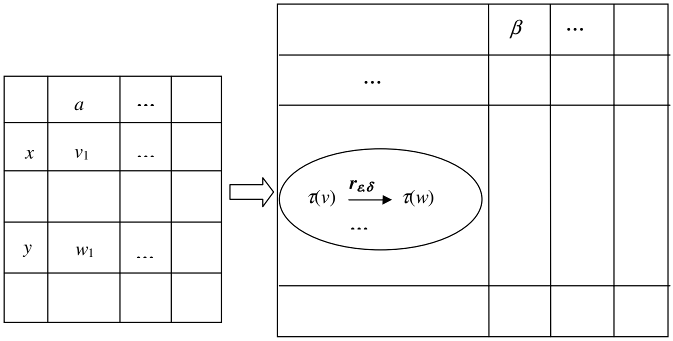


Fig. 5 Granulation of tolerance relational structures to clusters of such structures. $r_{\varepsilon,\delta}$ is a relation with parameters ε,δ on similarity (tolerance) classes

objects. Such families can be used to define new neighborhoods for a new approximation space by, e.g., taking their unions. In the process of searching for relevant neighborhoods, we use information encoded in the available sample. More relevant neighborhoods make it possible to define more relevant approximation spaces (from the point of view of the optimization criterion). Following this scheme, the next level of granulation may be related to clusters of objects (relational structures) for a current level (see Fig. 5).

In Fig. 5 τ denotes a similarity (tolerance) relation on vectors of attribute values, $\tau(v) = \{u : \ v \ \tau \ u\}$, $\tau(v) \ r_{\varepsilon,\delta} \ \tau(w)$ iff $\mathrm{dist}(\tau(v),\tau(w)) \in [\varepsilon-\delta,\varepsilon+\delta]$, and $\mathrm{dist}(\tau(v),\tau(w)) = \inf\{\mathrm{dist}(v',w') : (v',w') \in \tau(v)\times\tau(w)\}$ where dist is a distance function on vectors of attribute values.

One more example is illustrated in Fig. 6, where the next level of hierarchical modeling is created by defining an information system in which objects are time windows and attributes are (time-related) properties of these windows.

It is worth mentioning that quite often this searching process is even more sophisticated. For example, one can discover several relational structures (e.g., corresponding

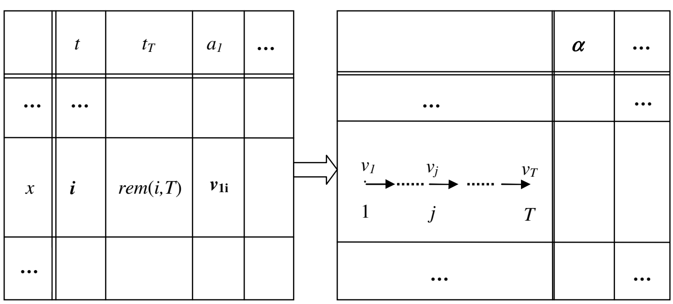


Fig. 6 Granulation of time points into time windows. T is the time window length, $v_j = (v_{1j},\ldots,v_{Tj})$ for $j = 1,\ldots,T$, $rem(i,T)$ is the remainder from division of i by T, α is an attribute defined over time windows

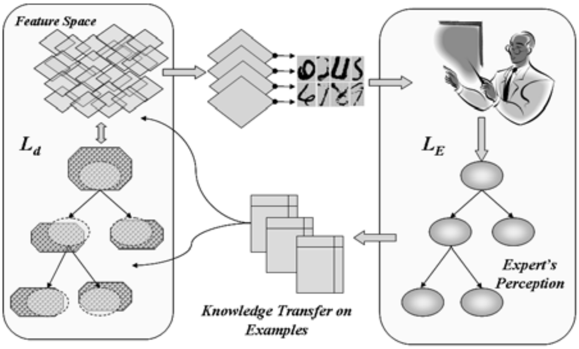


Fig. 7 Expert's knowledge elicitation

to different attributes) and formulas over such structures defining different families of neighborhoods from the original approximation space. As a next step, such families of neighborhoods can be merged into neighborhoods in a new, higher degree approximation space.

The proposed approach is making it possible to construct information systems (or decision tables) on a given level of hierarchical modeling from information systems from lower level(s) by using some constraints in joining objects from underlying information systems. In this way, structural objects can be modeled and their properties can be expressed in constructed information systems by selecting relevant attributes. These attributes are defined by means of a language that makes use of attributes of systems from the lower hierarchical level as well as relations for defining constraints [50, 48, 3]. In some sense, the objects on the next level of hierarchical modeling are defined using the syntax from the lover level of the hierarchy. Domain knowledge is used to aid the discovery of relevant attributes (features) on each level of hierarchy. This domain knowledge can be provided, e.g., by concept ontology together with samples of objects illustrating concepts from this ontology. Such knowledge is making it feasible to search for relevant attributes (features) on different levels of hierarchical modeling (see Sect. 4). In Fig. 7 we symbolically illustrate the transfer of knowledge in a particular application. It is a depiction of how the knowledge about outliers in handwritten digit recognition is transferred from expert to a software system. We call this process *knowledge elicitation*. Observe, that the explanations given by expert(s) are expressed using a subset of natural language limited by using concepts from provided ontology only. Concepts from higher levels of ontology are gradually approximated by the system from concepts on lower levels. This kind of approach is typical for hierarchical modeling [8]. This is, in particular, the case when we search for a relevant approximation space for objects composed from parts for which some approximation spaces, relevant to components, have already been found. We find that hierarchical modeling is required for approximation of complex vague concepts, as in [33, 43].

4 Ontologies as Complex Granules and Their Approximation in RGC

Approximation of complex, possibly vague concepts requires a hierarchical modeling and approximation of more elementary concepts on subsequent levels in the hierarchy along with utilization of domain knowledge. Due to the complexity of these concepts and processes on top levels in the hierarchy one can not assume that fully automatic construction of their models, or the discovery of data patterns required to approximate their components, would be straightforward. We propose to include in this process the discovery of approximations of complex vague concepts, performed interactively with co-operation of domain experts. Such interaction allows for more precise control over the complexity of discovery process, therefore making it computationally more feasible. Thus, the proposed approach transforms a typical data mining system into an equivalent of experimental laboratory (in particular, for *ontology approximation*) in which the software system, aided by human experts, attempts to discover: (i) approximation of complex vague concepts from data under some domain constraints, (ii) patterns relevant to user (researcher), e.g., required in the approximation of vague components of complex concepts.

The research direction aiming at interactive knowledge construction has been pursued by our team, in particular, toward the construction of classifiers for complex concepts (see, e.g., [8, 10, 7, 6, 5, 4] and also [14, 29, 30, 31, 32]) aided by domain knowledge integration. Advances in recent years indicate a possible expansion of the research conducted so far into discovery of models for processes involving complex objects from temporal or spatio-temporal data.

The novelty of the proposed RGC approach for the discovery of approximations of complex concepts from data and domain knowledge lies in combining, on one side, a number of novel methods of granular computing developed using the rough set methods and other known approaches to the approximation of vague, complex concepts (see, e.g., [8, 10, 11, 7, 6, 5, 4, 20, 29, 30, 31, 32, 38, 39, 41, 42, 66, 67]) with, on the other side, the discovery of structures from data through an interactive collaboration with domain experts (see, e.g., [8, 10, 11, 7, 6, 5, 4, 20, 29, 30, 31, 32, 42]). The developed methodology based on RGC was applied, to various extent, in real-life projects including: unmanned area vehicle control, robotics, prediction of risk patterns from temporal medical and financial data, sunspot classification, and bioinformatics. For technical details please refer to [8, 10, 11, 7, 6, 5, 4] and [20, 29, 30, 31, 32, 42]).

5 Toward RGC for Process Mining

The rapid expansion of the Internet has resulted not only in the ever growing amount of data therein stored, but also in the burgeoning complexity of the concepts and phenomena pertaining to those data. This issue has been vividly compared in [16] to the advances in human mobility from the period of walking afoot to the era of jet travel. These essential changes in data have brought new challenges to the development of new data mining methods, especially that the treatment of these data increasingly involves complex processes that elude classic modeling paradigms. Types of datasets currently regarded 'hot', like biomedical, financial or net user behavior data are just a few examples. Mining such temporal or complex data streams is on the agenda of many research centers and companies worldwide (see, e.g., [1, 46]). In the data mining community, there is a rapidly growing interest in developing methods for *process mining*, e.g., for discovery of structures of temporal processes from observations (recorded data). Works on process mining, e.g., [13, 26, 59, 61] have recently been undertaken by many renowned centers worldwide[1]. This research is also related to functional data analysis (cf. [45]), cognitive networks (cf. [37]), and dynamical system modeling in biology [15].

In [28, 27] we outlined an approach to discovery of processes from data and domain knowledge which is based on RGC philosophy.

In Sect. 6, we discuss some issues related to granule interactions also in process mining.

6 Toward Interactive RGC

Interactions between granules are rudimentary for understanding the nature of *interactive computations* [17]. In the RGC framework, it is possible to model interactive computations performed on granules of different complexity aiming at construction of approximations of complex vague concepts. Approximations of such concepts are capable of adaptive adjustment with the changes of underlying data and domain knowledge. Hence, the decision making algorithm based on the approximation of such vague concepts is also adaptively changing. Hence, our decision making algorithms are different from the classical algorithms which 'are metaphorically dump and blind because they cannot adapt interactively while they compute' [60].

In this section, we discuss some examples of interactions of granules showing the richness and complexity of granule interactions which should be modeled in RGC. The first example is related to discovery of concurrent systems from information systems.

Back in 1992, Zdzisław Pawlak (cf. [40]) proposed to use data tables (information systems) as specifications of concurrent systems. In this approach, any information system can be considered as a representation of a (traditional) concurrent system: attributes are interpreted as local processes of the concurrent system, values of attributes – as states of these local processes, and objects – as global states of the considered system. Several methods for synthesis of concurrent systems from data have been developed (see, e.g., [36, 51, 52, 57]). These methods are based on the following steps. First, for a given information system S we generate its (formal) theory $Th(S)$ consisting of a set of selected rules over descriptors defined by this system.

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

These rules describe the coexistence constraints of local states within global states specified by S. Next, we define a maximal extension $Ext(S)$ of S consisting of all objects having descriptions consistent with all rules in $Th(S)$. Finally, a Petri net with the set of reachable markings equal to $Ext(S)$ is generated. There have been also developed methods for synthesis of Petri nets from information systems based on decomposition of information systems into the so called components defined by reducts. This approach is making it possible to represent a given information system by a set of interacting local processes defined by some functional dependencies extracted from data. Interactions between local processes are represented by rules over descriptors extracted from data too. It is worth mentioning that the ability to produce from an information system a structure that is essentially (is similar to) a Petri net brings significant profits. Petri nets and similar structures have been studied for decades, and nowadays we have quite potent collection of tools that make use of these notions, at our disposal.

Our second example is related to learning of state changes for agents interacting with dynamically changing environments. One possible approach can be analogous to modeling by differential equations. However, instead of assuming the definition of the functions describing these changes we propose to approximate these functions from experimental data using domain knowledge [28, 27].

Let us assume that changes of the environment state $e(t)$ and agent state $s(t)$ interacting over time t are described by the following scheme of equations:

$$\begin{aligned} \Delta s(t) &= F(t, \Delta t, s(t), e(t)), \\ \Delta e(t) &= G(t, \Delta t, s(t), e(t)). \end{aligned} \tag{2}$$

Because the functions F, G are often highly nonlinear one can hardly expect that assuming some linear models one can obtain satisfactory solutions by tuning parameters in these models [15, 37, 22].

Due to uncertainty of information about states $s(t), e(t)$ one can only look for approximations of functions F, G from available data. When we learn approximations of functions F, G it is necessary to develop methods for computing approximations of trajectories of solutions based on interaction of approximations of functions F, G with granules representing uncertain information about states. We couple of function approximations with descriptions of indiscernibility (similarity) classes in which the current state is included in order to identify indiscernibility (similarity) classes for the next state(s). This requires some special interaction of granule representing uncertain information about the current state and the granule represented by approximation of functions describing changes between consecutive states. First, the granule of object is interacting with components of function approximation. This step is, in some sense, analogous to fuzzification in fuzzy control. In the case of rule based classifier, this step involves search for inclusion degrees of object granule and patterns represented by the left hand sides (antecendents) of rules. This may be perceived as matching membership degrees in fuzzy controller. Finally, the results of the interaction are fused to form a granule representing the next state. Again, this step is analogous to defuzzification in fuzzy controller. In the case of rule based classifier, this step is based on the conflict resolution strategy or voting strategy making it possible to select or construct the final decision granule in presence of possibly contradictory, partially matching rules. We perceive the idea described above as very important direction for further research on methods for discovery of process trajectory approximation from data and domain knowledge.

More advanced interaction of processes may occur if we consider the situation when each path in a given process is represented by a vector of attribute values. Such a situation may occur when, for instance, paths from the lower level undergo clustering. Then, some additional constraints can be related to paths of the resulting process constructed from paths of interacting, lower-level processes. They may represent results of synchronization of two or more processes. For example, in any path of the process obtained as a result of interaction between two lower-level processes states with a certain distinguished property should separate (appear in-between) states with another specific property.

It should be noted that in practical approaches to modeling it is often necessary to use relevant names (labels) for the constructed processes, tantamount to their position and rôle in concept hierarchy (or corresponding ontology). To answer to this requirement one may use methods of inducing, e.g., Petri nets from examples of paths (see, e.g., [26]).

Another way of looking at modeling of interactions is by employing the agent-oriented framework. The depiction of agents' interactions with environment(s) is essentially based on observation, that each agent perceives only a partial (and possibly vague) information about environment. On the basis of the perceived information and its own state the agent derives (creates) some granules, with the goal of changing the state of environment to its favor. These granules are involved in interactions with the environment and granules originating in other agents. Using either competitive or cooperative strategies (coalitions of) agents involved in interactions form a resulting action which changes the environment(s) in a way that is in some accordance with components (agent-specific granules). The approaches that use elements of such interactive agent co-operation are nowadays popular in multiagent systems [25, 56].

In the following, final example we describe an application of domain knowledge in modeling of interactions. We use sentences from (a limited subset of) the natural language coupled with so called *behavioral graphs* [8] to define relationships (interactions) that occur between parts of a complex object. In this example we show such description for the task of recognizing whether at a given moment the observed road situation leads to imminent danger or not. The modeling of the system that ultimately is capable of recognizing the extremely compound concept of *dangerous situation* on the basis of low-level measurements, is indeed hierarchical. In Fig. 8 we present a behavioral graph for a single object-vehicle on a road. This behavioral graph appears in between the lowest level (sensor measurements) and the highest level (dangerous situation) in the hierarchy of concepts.

A composition of behavioral graphs, appearing on lower level in the hierarchy, can be used to represent behavior (and interaction) of a more compound part consisting of, e.g., two vehicles involved in the maneuver of overtaking (see Fig. 9).

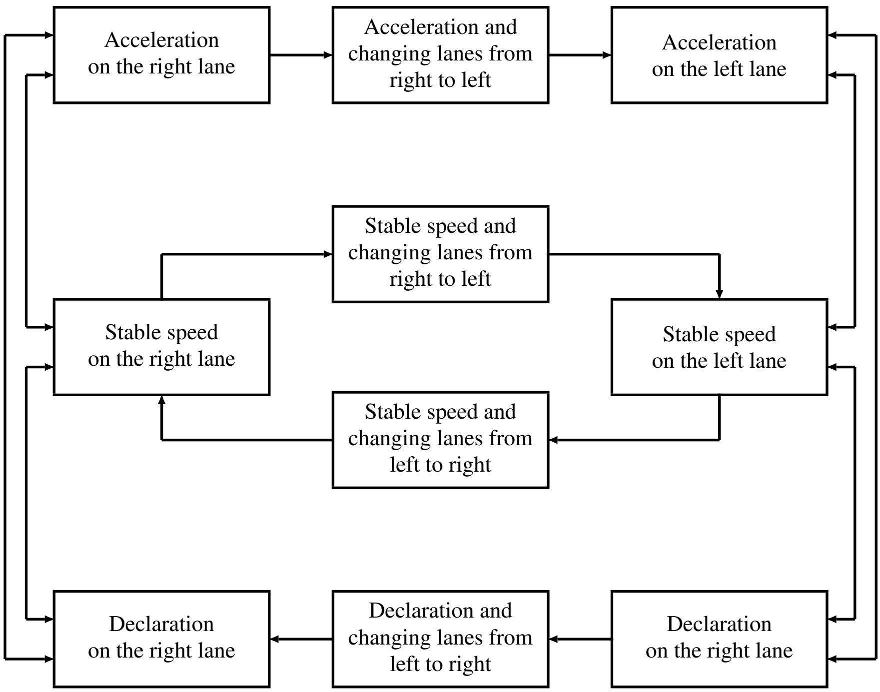


Fig. 8 A behavioral graph for a single object-vehicle

Please note, that the concept of *overtaking* is built of components which at some point were also approximated from the lower level concepts. This is a case of, e.g., *changing lane* or *A passing B* (refer to Fig. 9).

The identification of the behavioral pattern of a complex object on the basis of sensory data cannot go forward without (approximation of) ontology of concepts. It is this ontology that makes it possible to link the low level measurements (sensory concepts) with the high level description of behavioral patterns [8, 10, 11, 7, 5, 4, 20, 42]. By means of this ontology we establish that – following our road example – in order to know what the *overtaiking* is, one has to define a concept of *A passing B*, as well as link both *A* and *B* to an object-vehicle structure (see Fig. 8).

An example of behavioral graphs for medical application [4] is presented in Fig. 10. Behavioral graphs based on domain knowledge were also used in risk analysis for medical data [8, 10, 11, 7, 5, 4, 20, 42].

Models of behavioral patterns can also be represented by Petri nets, differential equations or using relevant formal theories. Such patterns are used in further steps for modeling more compound processes obtained by interaction of patterns representing local processes. It is important to note that one can hardly expect to discover such models fully automatically from data, without cooperation with experts.

The *RoughIce* platform for hierarchical modeling, in particular for modeling of interactions is available at `logic.mimuw.edu.pl/~bazan/roughice/`.

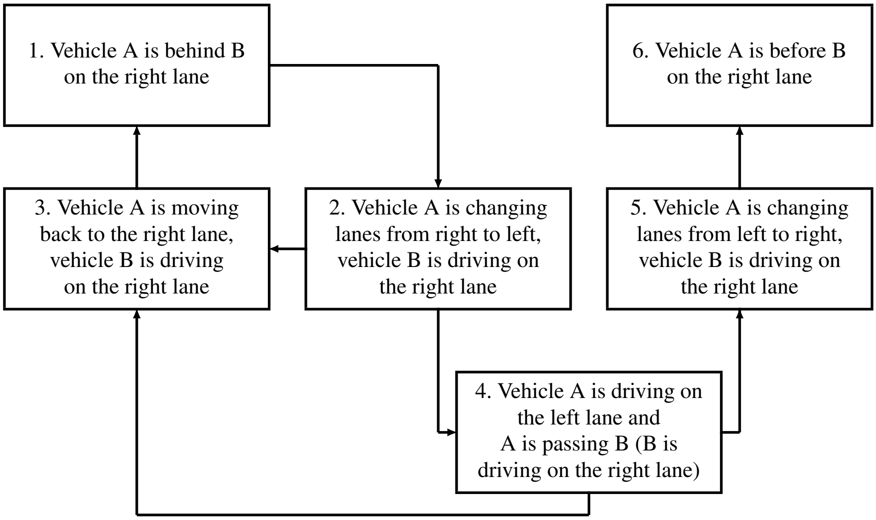


Fig. 9 A behavioral graph for the maneuver of overtaking

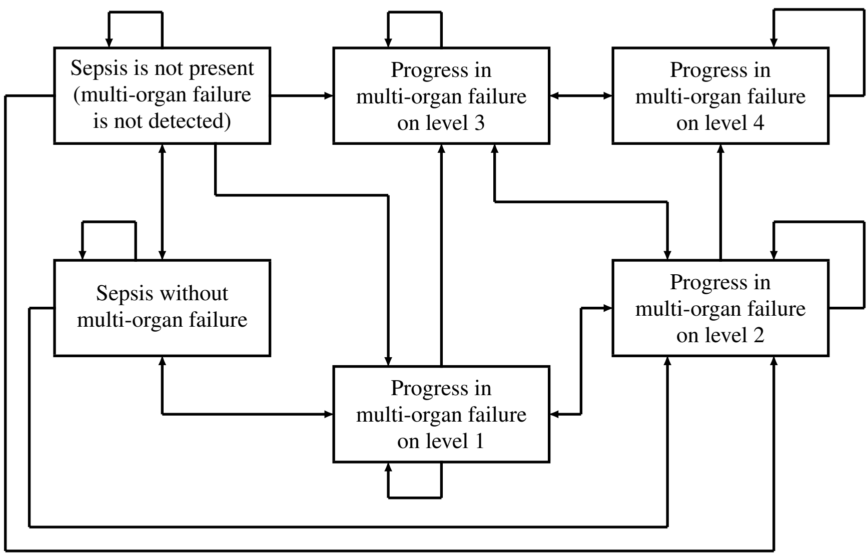


Fig. 10 A behavioral graph of sepsis by analyzing the multi-organ failure

7 Conclusions

We discussed some issues closely related to research directions within the *Wisdom Technology* (Wistech) research programme, as outlined recently in [18, 19, 20, 21]. There are possible different ways to build computational models that are based on

Wistech philosophy. We outlined some steps of such modeling for just one of them, which is based on the RGC approach. The approach is promising for developing new methods based on human-machine interactions.

Acknowledgements. The research has been partially supported by the grant N N516 368334 from Ministry of Science and Higher Education of the Republic of Poland.

References

1. Aggarwal, C. (ed.): Data Streams: Models and Algorithms. Springer, Berlin (2007)
2. Bargiela, A., Pedrycz, W.: Granular Computing: An Introduction. Kluwer Academic Publishers, Dordrecht (2003)
3. Barwise, J., Seligman, J.: Information Flow: The Logic of Distributed Systems. Cambridge University Press, Cambridge (1997)
4. Bazan, J.: Hierarchical classifiers for complex spatio-temporal concepts. In: Peters, J.F., Skowron, A., Rybiński, H. (eds.) Transactions on Rough Sets IX. LNCS, vol. 5390, pp. 474–750. Springer, Heidelberg (2008)
5. Bazan, J.: Rough sets and granular computing in behavioral pattern identification and planning. In: Pedrycz, W., Skowron, A., Kreinovich, V. (eds.) Handbook of Granular Computing, pp. 777–800. John Wiley & Sons, New York (2008)
6. Bazan, J., Kruczek, P., Bazan-Socha, S., Skowron, A., Pietrzyk, J.J.: Automatic planning of treatment of infants with respiratory failure through rough set modeling. In: Greco, S., Hata, Y., Hirano, S., Inuiguchi, M., Miyamoto, S., Nguyen, H.S., Słowiński, R. (eds.) RSCTC 2006. LNCS (LNAI), vol. 4259, pp. 418–427. Springer, Heidelberg (2006)
7. Bazan, J., Kruczek, P., Bazan-Socha, S., Skowron, A., Pietrzyk, J.J.: Risk pattern identification in the treatment of infants with respiratory failure through rough set modeling. In: Proceedings of Information Processing and Management under Uncertainty in Knowledge-Based Systems, vol. 3, pp. 2650–2657. Editions E.D.K., Paris (2006)
8. Bazan, J., Peters, J.F., Skowron, A.: Behavioral pattern identification through rough set modelling. In: Ślęzak, D., Szczuka, M., Düntsch, I., Yao, Y. (eds.) RSFDGrC 2005. LNCS (LNAI), vol. 3642, pp. 688–697. Springer, Heidelberg (2005)
9. Bazan, J., Skowron, A.: Classifiers based on approximate reasoning schemes. In: Dunin-Kęplicz, B., Jankowski, A., Skowron, A., Szczuka, M. (eds.) Monitoring, Security, and Rescue Techniques in Multiagent Systems. Advances in Soft Computing, pp. 191–202. Springer, Heidelberg (2005)
10. Bazan, J., Skowron, A.: On-line elimination of non-relevant parts of complex objects in behavioral pattern identification. In: Pal, S.K., Bandoyopadhay, S., Biswas, S. (eds.) PReMI 2005. LNCS, vol. 3776, pp. 720–725. Springer, Heidelberg (2005)
11. Bazan, J., Skowron, A., Swiniarski, R.: Rough sets and vague concept approximation: From sample approximation to adaptive learning. In: Peters, J.F., Skowron, A. (eds.) Transactions on Rough Sets V. LNCS, vol. 4100, pp. 39–62. Springer, Heidelberg (2006)
12. Behnke, S.: Hierarchical Neural Networks for Image Interpretation. LNCS, vol. 2766. Springer, Heidelberg (2003)
13. Borrett, S.R., Bridewell, W., Arrigo, P.L.K.R.: A method for representing and developing process models. Ecological Complexity 4(1-2), 1–12 (2007)
14. Doherty, P., Łukaszewicz, W., Skowron, A., Szałas, A.: Knowledge Representation Techniques: A Rough Set Approach. Studies in Fuzziness and Soft Computing, vol. 202. Springer, Heidelberg (2006)

15. Feng, J., Jost, J., Minping, Q.: Network: From Biology to Theory. Springer, Heidelberg (2007)
16. Friedman, J.H.: Data mining and statistics. What's the connection? - Keynote address. In: Proceedings of the 29th Symposium on the Interface: Computing Science and Statistics, Houston, US (1997)
17. Goldin, D., Smolka, S., Wegner, P.: Interactive Computation: The New Paradigm. Springer, Heidelberg (2006)
18. Jankowski, A., Skowron, A.: A wistech paradigm for intelligent systems. In: Peters, J.F., Skowron, A., Düntsch, I., Grzymała-Busse, J.W., Orłowska, E., Polkowski, L. (eds.) Transactions on Rough Sets VI. LNCS, vol. 4374, pp. 94–132. Springer, Heidelberg (2007)
19. Jankowski, A., Skowron, A.: Logic for artificial intelligence: The Rasiowa-Pawlak school perspective. In: Ehrenfeucht, A., Marek, V., Srebrny, M. (eds.) Andrzej Mostowski and Foundational Studies, pp. 106–143. IOS Press, Amsterdam (2008)
20. Jankowski, A., Skowron, A.: Wisdom granular computing. In: Pedrycz, W., Skowron, A., Kreinovich, V. (eds.) Handbook of Granular Computing, pp. 329–346. John Wiley & Sons, New York (2008)
21. Jankowski, A., Skowron, A.: Wisdom technology: A rough-granular approach. In: Festschrift dedicated to Leonard Bolc, pp. 1–40. Springer, Heidelberg (2009) (in print)
22. Kleinberg, J., Papadimitriou, C., Raghavan, P.: A microeconomic view of data mining. Data Mining and Knowledge Discovery 2, 311–324 (1998)
23. Leibniz, G.: Dissertio de Arte Combinatoria. Lepzig, Germany (1666)
24. Leibniz, G.: New Essays on Human Understanding, Cambridge, UK (1982); Written in 1705, translated and edited by Remnant, P., Bennett, J.
25. Luck, M., McBurney, P., Preist, C.: Agent technology. Enabling next generation computing: A roadmap for agent based computing (2003), `www.agentlink.org`
26. de Medeiros, A.K.A., Weijters, A.J.M.M., van der Aalst, W.M.P.: Genetic process mining: An experimental evaluation. Data Mining and Knowledge Discovery 14, 245–304 (2007)
27. Nguyen, H.S., Jankowski, A., Skowron, A., Stepaniuk, J., Szczuka, M.: Discovery of process models from data and domain knowledge: A rough-granular approach. In: Yao, J.T. (ed.) Novel Developments in Granular Computing: Applications for Advanced Human Reasoning and Soft Computation, pp. 1–30. IGI Global, Hershey (2008) (in print)
28. Nguyen, H.S., Skowron, A.: A rough granular computing in discovery of process models from data and domain knowledge. Journal of Chongqing University of Post and Telecommunications 20(3), 341–347 (2008)
29. Nguyen, S.H., Bazan, J., Skowron, A., Nguyen, H.S.: Layered learning for concept synthesis. In: Peters, J.F., Skowron, A., Grzymała-Busse, J.W., Kostek, B.z., Świniarski, R.W., Szczuka, M.S. (eds.) Transactions on Rough Sets I(1). LNCS, vol. 3100, pp. 187–208. Springer, Heidelberg (2004)
30. Nguyen, T.T.: Eliciting domain knowledge in handwritten digit recognition. In: Pal, S.K., Bandoyopadhay, S., Biswas, S. (eds.) PReMI 2005. LNCS, vol. 3776, pp. 762–767. Springer, Heidelberg (2005)
31. Nguyen, T.T.: Outlier and exception analysis in rough sets and granular computing. In: Pedrycz, W., Skowron, A., Kreinovich, V. (eds.) Handbook of Granular Computing, pp. 823–834. John Wiley & Sons, New York (2008)
32. Nguyen, T.T., Paddon, C.P.W.D.J., Nguyen, S.H., Nguyen, H.S.: Learning sunspot classification. Fundamenta Informaticae 72(1-3), 295–309 (2006)

33. Nguyen, T.T., Skowron, A.: Rough-granular computing in human-centric information processing. In: Bargiela, A., Pedrycz, W. (eds.) Human-Centric Information Processing Through Granular Modelling. Studies in Computational Intelligence, vol. 182, pp. 1–30. Springer, Heidelberg (2009)
34. Pal, S.K., Bandyopadhyay, S., Biswas, S. (eds.): PReMI 2005. LNCS, vol. 3776. Springer, Heidelberg (2005)
35. Pal, S.K., Polkowski, L., Skowron, A. (eds.): Rough-Neural Computing: Techniques for Computing with Words. Cognitive Technologies. Springer, Heidelberg (2004)
36. Pancerz, K., Suraj, Z.: Discovering concurrent models from data tables with the ROSECON. Fundamenta Informaticae 60(1-4), 251–268 (2004)
37. Papageorgiou, E.I., Stylios, C.D.: Fuzzy cognitive maps. In: Pedrycz, W., Skowron, A., Kreinovich, V. (eds.) Handbook of Granular Computing, pp. 755–774. John Wiley & Sons, New York (2008)
38. Pawlak, Z.: Rough sets. International Journal of Computer and Information Sciences 11, 341–356 (1982)
39. Pawlak, Z.: Rough Sets: Theoretical Aspects of Reasoning about Data, System Theory. In: Knowledge Engineering and Problem Solving, vol. 9. Kluwer Academic Publishers, Dordrecht (1991)
40. Pawlak, Z.: Concurrent versus sequential the rough sets perspective. Bulletin of the EATCS 48, 178–190 (1992)
41. Pawlak, Z., Skowron, A.: Rudiments of rough sets; Rough sets: Some extensions; Rough sets and boolean reasoning. Information Sciences 177(1), 3–73 (2007)
42. Pedrycz, W., Skowron, A., Kreinovich, V. (eds.): Handbook of Granular Computing. John Wiley & Sons, New York (2008)
43. Poggio, T., Smale, S.: The mathematics of learning: Dealing with data. Notices of the AMS 50(5), 537–544 (2003)
44. Polkowski, L., Skowron, A.: Rough mereology: A new paradigm for approximate reasoning. International Journal of Approximate Reasoning 51, 333–365 (1996)
45. Ramsay, J.O., Silverman, B.W.: Applied Functional Data Analysis. Springer, Heidelberg (2002)
46. Roddick, J.F., Hornsby, K., Spiliopoulou, M.: An updated bibliography of temporal, spatial and spatio-temporal data mining research. In: Roddick, J.F., Hornsby, K. (eds.) TSDM 2000. LNCS (LNAI), vol. 2007, pp. 147–163. Springer, Heidelberg (2001)
47. Skowron, A.: Rough sets and vague concept. Fundamenta Informaticae 64, 417–431 (2005)
48. Skowron, A., Stepaniuk, J.: Information granules and rough-neural computing. In: Pal, S.K., Polkowski, L., Skowron, A. (eds.) Rough-Neural Computing: Techniques for Computing with Words, Cognitive Technologies, pp. 43–84. Springer, Heidelberg (2003)
49. Skowron, A., Stepaniuk, J., Peters, J., Swiniarski, R.: Calculi of approximation spaces. Fundamenta Informaticae 72(1-3), 363–378 (2006)
50. Skowron, A., Stepaniuk, J., Peters, J.F.: Rough sets and infomorphisms: Towards approximation of relations in distributed environments. Fundamenta Informaticae 54(2-3), 263–277 (2003)
51. Skowron, A., Suraj, Z.: Rough sets and concurrency. Bulletin of the Polish Academy of Sciences 41, 237–254 (1993)
52. Skowron, A., Suraj, Z.: Discovery of concurrent data models from experimental tables: A rough set approach. In: Proceedings of the 1st International Conference on Knowledge Discovery and Data Mining, pp. 288–293. AAAI Press, Menlo Park (1995)
53. Skowron, A., Synak, P.: Complex patterns. Fundamenta Informaticae 60(1-4), 351–366 (2004)

54. Skowron, A., Szczuka, M.: Toward interactive computations: A rough-granular approach. In: Koronacki, J., Wierzchon, S., Ras, Z., Kacprzyk, J. (eds.) Commemorative Volume to Honor Ryszard Michalski, pp. 1–20. Springer, Heidelberg (2009) (in print)
55. Stone, P.: Layered Learning in Multi-Agent Systems: A Winning Approach to Robotic Soccer. MIT Press, Cambridge (2000)
56. Sun, R. (ed.): Cognition and Multi-Agent Interaction. From Cognitive Modeling to Social Simulation. Cambridge University Press, Cambridge (2006)
57. Suraj, Z.: Rough set methods for the synthesis and analysis of concurrent processes. In: Polkowski, L., Lin, T., Tsumoto, S. (eds.) Rough Set Methods and Applications: New Developments in Knowledge Discovery in Information Systems. Studies in Fuzziness and Soft Computing, vol. 56, pp. 379–488. Springer, Heidelberg (2000)
58. Thiele, L.P.: The Heart of Judgment: Practical Wisdom, Neuroscience, and Narrative. Cambridge University Press, Edinburgh (2006)
59. Unnikrishnan, K.P., Ramakrishnan, N., Sastry, P.S., Uthurusamy, R. (eds.): Proceedings of the 4th Workshop on Temporal Data Mining: Network Reconstruction from Dynamic Data, Philadelphia, US (2006), `http://people.cs.vt.edu/~ramakris/kddtdm06/`
60. Wegner, P.: Why interaction is more powerful than algorithms. Communications of the ACM 40, 80–91 (1997)
61. Wu, F.X.: Inference of gene regulatory networks and its validation. Current Bioinformatics 2(2), 139–144 (2007)
62. Zadeh, L.: Foreword. In: Pal, S.K., Polkowski, L., Skowron, A. (eds.) Rough-Neural Computing: Techniques for Computing with Words, pp. IX–XI. Springer, Heidelberg (2004)
63. Zadeh, L.A.: Outline of a new approach to the analysis of complex systems and decision processes. IEEE Transactions on Systems, Man and Cybernetics 3, 28–44 (1973)
64. Zadeh, L.A.: Fuzzy sets and information granularity. In: Gupta, M., Ragade, R., Yager, R. (eds.) Advances in Fuzzy Set Theory and Applications, pp. 3–18. North-Holland Publishing Co., Amsterdam (1979)
65. Zadeh, L.A.: Outline of a computational approach to meaning and knowledge representation based on the concept of a generalized assignment statement. In: Thoma, M., Wyner, A. (eds.) Proceedings of the International Seminar on Artificial Intelligence and Man-Machine System, pp. 198–211. Springer, Heidelberg (1986)
66. Zadeh, L.A.: A new direction in AI - toward a computational theory of perceptions. Artificial Intelligence Magazine 22(1), 73–84 (2001)
67. Zadeh, L.A.: Generalized theory of uncertainty (GTU)-principal concepts and ideas. Computational Statistics and Data Analysis 51, 15–46 (2006)