A Wistech Paradigm for Intelligent Systems

Andrzej Jankowski[1,2] and Andrzej Skowron[3]

[1] Institute of Decision Processes Support
[2] AdgaM Solutions Sp. z o.o.
Wąwozowa 9 lok. 64, 02-796 Warsaw, Poland
andrzejj@adgam.com.pl
[3] Institute of Mathematics,
Warsaw University
Banacha 2, 02-097 Warsaw, Poland
skowron@mimuw.edu.pl

If controversies were to arise, there would be no more need of disputation between two philosophers than between two accountants. For it would suffice to take their pencils in their hands, and say to each other: 'Let us calculate'.

– Gottfried Wilhelm Leibniz,
Dissertio de Arte Combinatoria (Leipzig, 1666).

... Languages are the best mirror of the human mind, and that a precise analysis of the signification of words would tell us more than anything else about the operations of the understanding.

– Gottfried Wilhelm Leibniz,
New Essays on Human Understanding (1705)
Translated and edited by
Peter Remnant and Jonathan Bennett
Cambridge: Cambridge UP, 1982

Abstract. The problem considered in this article is how does one go about discovering and designing intelligent systems. The solution to this problem is considered in the context of what is known as wisdom technology (wistech), an important computing and reasoning paradigm for intelligent systems. A rough-granular approach to wistech is proposed for developing one of its possible foundations. The proposed approach is, in a sense, the result of the evolution of computation models developed in the Rasiowa–Pawlak school. We also present a long-term program for implementation of what is known as a wisdom engine. The program is defined in the framework of cooperation of many Research & Development (R & D) institutions and is based on a wistech network (WN) organization.

Keywords: wisdom technology, adaptive rough-granular computing, rough sets, wisdom engine, open innovation, wisdom network.

J.F. Peters et al. (Eds.): Transactions on Rough Sets VI, LNCS 4374, pp. 94–132, 2007.

1 Introduction

Huge technological changes occurred during the second half of the 20th century affecting every one of us. These changes affect practically all objects manufactured by man such as spoons, clothing, books, and space rockets. There are many indications that we are currently witnessing the onset of an era of radical changes. These radical changes depend on the further advancement of technology to acquire, represent, store, process, discover, communicate and learn wisdom. In this paper, we call this technology *wisdom technology* (or wistech, for short). The term *wisdom* commonly means "judging rightly" [50]. This common notion can be refined. By *wisdom*, we understand an adaptive ability to make judgements correctly to a satisfactory degree (in particular, correct decisions) having in mind real-life constraints.

One of the basic objectives of the paper is to indicate the potential directions for the design and implementation of wistech computation models. An important aspect of wistech is that the complexity and uncertainty of real-life constraints mean that in practise we must reconcile ourselves to the fact that our judgements are based on non-crisp concepts and also do not take into account all the knowledge accumulated and available to us. This is why consequences of our judgements are usually imperfect. But as a consolation, we also learn to improve the quality of our judgements via observation and analysis of our experience during interaction with the environment. Satisfactory decision-making levels can be achieved as a result of improved judgements.

The intuitive nature of wisdom understood in this way can be expressed metaphorically as shown in (1).

$$wisdom = KSN + AJ + IP, \tag{1}$$

where *KSN, AJ, IP* denote *knowledge sources network, adaptive judgement*, and *interactive processes*, respectively. The combination of the technologies represented in (1) offers an intuitive starting point for a variety of approaches to designing and implementing computational models for wistech. In this paper, (1) is called the *wisdom equation.* There are many ways to build wistech computational models. In this paper, the focus is on an adaptive rough-granular approach.

The issues discussed in this article are relevant for the current research directions (see, e.g., [16,15,31,38,51,90,108] and the literature cited in these articles).

This paper is organized as follows.

2 Wisdom Technology

This section briefly introduces the wistech paradigm.

2.1 What Do We Mean by Wistech?

On the one hand, the idea expressed by (1) (the wisdom equation paradigm) is a step in the direction of a new philosophy for the use of computing machines in our daily life, referred to as ubiquitous computing (see [66]). This paradigm is strongly connected with various applications of autonomic computing [64]. On the other hand, it should be emphasized that the idea of integrating many basic AI concepts (e.g., interaction, knowledge, network, adaptation, assessment, pattern recognition, learning, network, simulation of behavior in an uncertain environment, planning and problem solving) is as old as the history of AI itself. Many examples of such an approach adopted by researchers in the middle of the 20th century can be found in [27]. This research was intensively continued in the second half of the 20th century. For example, the abstracts of thousands of interesting reports from the years 1954 -1985 can be found in [91,92].

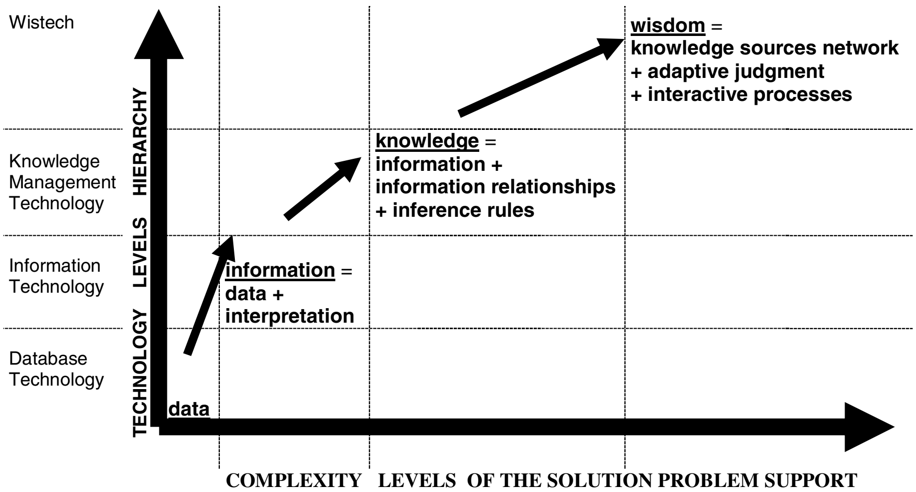
This paper contains the conclusions of the authors' experiences during numerous practical projects implementing wistech technologies in specific applications, e.g., fraud detection (MERIX – a prototype system for Bank of America), dialogue based search engine (EXCAVIO – intelligent search engine), UAV control (WITAS project), Intelligent marketing (data mining and optimization system for Ford Motor Company, General Motors), robotics, EVOLUTIONARY CHECKERS (adaptive checker R&D program at the University of North Carolina at Charlotte) and many other applications. These experiences are summarized by the authors in the metaphoric wisdom equation (1). This equation can also be illustrated using the following diagram presented in Figure 1.

In Figure 1 the term 'data' is understood as a stream of symbols without any interpretation of their meaning.

From the perspective of the metaphor expressed in the wisdom equation (1), wistech can be perceived as the integration of three technologies (corresponding to three components in the wisdom equation (1)). At the current stage two of them seem to be conceptually relatively clear, namely

1. *knowledge sources network* – by knowledge we traditionally understand every organized set of information along with the inference rules; in this context one can easily imagine the following examples illustrating the concept of knowledge sources network:
   – representation of states of reality perceived by our senses (or observed by the "receptors" of another observer) are integrated as a whole in our minds in a network of sources of knowledge and then stored in some part of our additional memory,
   – a network of knowledge levels represented by agents in some multi-agent system and the level of knowledge about the environment registered by means of receptors;
2. *interactive processes* – interaction understood as a sequence of stimuli and reactions over time; examples are:
   – the dialogue of two people,
   – a sequence of actions and reactions between an unmanned aircraft and the environment in which the flight takes place, or
   – a sequence of movements during some multi-player game.

Far more difficult conceptually seems to be the concept of adaptive judgement distinguishing wisdom from the general concept of problem solving. Intuitions behind this concept can be expressed as follows:



	Understanding	**Perception**	**Prediction**
Questions:	**Questions:** about data values and *additionally questions about data context, like:* ***Who? What? When? Where? How Much?***	**Questions:** information type questions and *additionally questions about explanation and prediction, like:* ***How? Why? What if?***	**Questions:** knowledge type *questions and additionally questions about correct judgments and decisions, action/ interaction planning / executing and justification, like:* ***What to do? Why to do it? When to do it? How to do it?***
Objects:	**Objects**: *Data and data explanation by a description, picture or other presentations.*	**Objects:** *Information and rules for information transformation (reasoning, ...), constrains, relationships between concepts, ideas and thought patterns.*	**Objects:** *Knowledge and correct judgments, decisions based on a hierarchy of being values or believes, action plans, incorporation of vision, design, plans and implementation standards based on being preferences.*
Time context:	**Time context:** *Usually information is a posteriori, known after the fact.*	**Time context:** *Usually knowledge is a priori, known before the fact and provides its meaning.*	**Time context:** *Usually deals with management interactions with environment to achieve future objectives.*
Measures:	**Measures:** *Logical values, uncertainty, completeness, amount.*	**Measures:** *Efficiency of problem solutions by applying theory to information, quality of problem solutions.*	**Measures:** *Priorities, culture values, profits, quality of action results and plan implementation.*

Fig. 1. Wisdom equation context

1. *adaptive judgement* – understood here as arriving at decisions resulting from the evaluation of patterns observed in sample objects. This form of judgement is made possible by mechanisms in a metalanguage (meta-reasoning) which on the basis of selection of available sources of knowledge and on the basis of understanding of history of interactive processes and their current status enable us to perform the following activities under real-life constraints:
   – identification and judgement of importance (for future judgement) of sample phenomena, available for observation, in the surrounding environment;

- planning current priorities for actions to be taken (in particular, on the basis of understanding of history of interactive processes and their current status) toward making optimal judgements;
- selection of fragments of ordered knowledge (hierarchies of information and judgement strategies) satisfactory for making a decision at the planned time (a decision here is understood as a commencing interaction with the environment or as selecting the future course to make judgements);
- prediction of important consequences of the planned interaction of processes;
- adaptive learning and, in particular, reaching conclusions deduced from patterns observed in sample objects leading to adaptive improvement in the adaptive judgement process.

One of the main barriers hindering an acceleration in the development of wistech applications lies in developing satisfactory computation models implementing the functioning of "adaptive judgement". This difficulty primarily consists in overcoming the complexity of the process of integrating the local assimilation and processing of changing non-crisp and incomplete concepts necessary to make correct judgements. 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 global models of analyzed phenomena (see, e.g., [110,62,64,45,25,21]). However, we can only approximate global models by integrating the various incomplete perspectives of problem perception. One of the potential computation models for "adaptive judgement" might be the *rough-granular approach*.

2.2 Main Differences Between Wisdom and Inference Engine

In natural language, the concept of wisdom is used in various semantic contexts. In particular, it is frequently semantically associated with such concepts as inference, reasoning, deduction, problem solving, judging rightly as a result of pattern recognition, common sense reasoning, reasoning by analogy, and others. As a consequence this semantic proximity may lead to misunderstandings. For example, one could begin to wonder what the difference is between the widely known and applied concept in AI of "inference engine" and the concept of "wisdom engine" defined in this paper? In order to avoid this type of misunderstanding it is worth explaining the basic difference between the understanding of wisdom and such concepts as inference, reasoning, deduction and others.

Above all, let as start with explaining how we understand the difference between problem solving and wisdom. The widespread concept of problem solving is described as some slight modification of this notion defined in the context of solving mathematical problems by George Pólya in [84]. The concept of problem solving is understood in [84] as the following set of activities:

1. *First, you have to understand the problem.*
2. *After understanding, then make a plan.*

3. *Carry out the plan.*
4. *Look back on your work. How could it be better?*

An attempt at explaining the concept of wisdom can be taken using the concept of *problem solving* in the following manner: wisdom is the ability to identify important problems, search for sufficiently correct solutions to them, having in mind real life, available knowledge sources, personal experience, constraints, etc. Having in mind this understanding of wisdom we get at once the first important difference. Namely, in the problem solving process we do not have the following important wisdom factor in the above sequence (1-4) of activities:

0. Learning to recognize patterns that identify important problems and problem solution constraints.

Certainly, this is not the only difference. Therefore, one can illustrate the general difference between the concept of problem solving and wisdom as the difference between the concept of flying in an artificially controlled environment (e.g., using a flying simulator and problem solving procedures) and the concept of flying Boeing 767 aeroplane in real-life dangerous environment (wisdom in a particular domain).

One can therefore think that wisdom is very similar to *the ability of problem solving in a particular domain of application*, which in the context of the world of computing machines is frequently understood as an *inference engine*. The commonly accepted definition of the concept of inference engine can be found for example in Wikipedia (http://en.wikipedia.org/wiki/Inference_engine). It refers to understanding of "problem solving" in the spirit of the book [84]. It reads as follows:

An inference engine is a computer program that tries to derive answers from a knowledge base. It is the "brain" that expert systems use to reason about the information in the knowledge base, for the ultimate purpose of formulating new conclusions.

An inference engine has three main elements. They are:

1. An interpreter. The interpreter executes the chosen agenda items by applying the corresponding base rules.
2. A scheduler. The scheduler maintains control over the agenda by estimating the effects of applying inference rules in light of item priorities or other criteria on the agenda.
3. A consistency enforcer. The consistency enforcer attempts to maintain a consistent representation of the emerging solution.

In other words, the concept of inference engine relates to generating strategies for the inference planning from potentially varied sources of knowledge which are in interaction together. So this concept is conceptually related to the following two elements of the wisdom equation:

1. knowledge sources network,
2. interactive processes.

However, it should be remembered that wisdom in our understanding is not only some general concept of *inference*. The basic characteristic of wisdom, distinguishing this concept from the general understanding of inference, is *adaptive ability to make correct judgements having in mind real-life constraints*. The significant characteristic differentiating wisdom from the general understanding of such concepts as problem solving or inference engine is adaptive judgement.

In analogy to what we did in the case of *problem solving*, we can now attempt to explain the concept of wisdom based on the notion of an *inference engine* in the following manner: Wisdom is an inference engine interacting with a real-life environment, which is able to identify important problems and to find for them sufficiently correct solutions having in mind real-life constraints, available knowledge sources and personal experience. In this case, one can also illustrate the difference between the concept of inference engine and the concept of wisdom using the metaphor of flying a plane.

One could ask the question of which is the more general concept: wisdom or problem solving? Wisdom is a concept carrying a certain additional structure of adaptive judgement which in a continuously improving manner assists us in identifying the most important problem to resolve in a given set of constraints and what an acceptable compromise between the quality of the solution and the possibility of achieving a better solution is. Therefore, the question of what the more general concept is closely resembles the question from mathematics: What is the more general concept in mathematics: the concept of a field (problem solving), or the concept of the vector space over a field (wisdom understood as problem solving + adaptive judgement)? The vector space is a richer mathematical structure due to the action on vectors. Analogously to wisdom it is a richer process (it includes adaptive judgement - a kind of meta-judgement that encompasses recognition of patterns common to a set of sample objects that leads to judgements relating to problem solving). On the other hand, research into single-dimensional space can be treated as the research of fields. In this sense, the concept of vector space over a field is more general than the concept of a field.

2.3 Why Does Wistech Seem to Be One of the Most Important Future Technologies?

Nobody today doubts that technologies based on computing machines are among the most important technology groups of the 20th century, and, to a considerable degree, have been instrumental in the progress of other technologies. Analyzing the stages in the development of computing machines, one can quite clearly distinguish the following three stages in their development in the 20th century:

1. Database Technology (gathering and processing of transaction data).
2. Information Technology (understood as adding to the database technology the ability to automate analysis, processing and visualization of information).
3. Knowledge Management Technology (understood as systems supporting organization of large data sets and the automatic support for knowledge processing and discovery (see, e.g., [59,18])).

The three stages of development in computing machine technology show us the trends for the further development in applications of these technologies. These trends can be easily imagined using the further advancement of complexity of information processing (Shannon Dimension) and advancement of complexity of dialogue intelligence (Turing Dimension), viz.,

– *Shannon Dimension* level of information processing complexity (representation, search, use);
– *Turing Dimension* the complexity of queries that a machine is capable of understanding and answering correctly. One of the objectives of AI is for computing machines to reach the point in Turing Dimension that is well-known Turing Test (see [114]).

In this framework, the development trends in the application of computing machines technology can be illustrated in Figure 2.

Technology	**Additional attributes**	**Shannon Dimensions**	**Turing Dimensions**
Database Technology	data is the most basic level	*How to represent information?*	*SQL*
Information Technology	information = data + interpretation	*Where to find information?*	*Who? What? When? Where? How much?*
Knowledge Management Technology	knowledge = information + information relationships + inference rules	*How to use information?*	*How? Why? What if?*

Fig. 2. Computing machines technology

Immediately from the beginning of the new millennium one can see more and more clearly the following new application of computing machine technology, viz., wisdom technology (wistech) which put simply can be presented in table (see Figure 3, being an extension of the table presented in Figure 2).

In other words, the trends in the development of technology of computing machines can be presented using the so-called DIKW hierarchy (i.e., Data, Information, Knowledge, Wisdom). Intuitively speaking, each level of the DIKW hierarchy adds certain attributes over and above the previous one. The hierarchy is presented graphically in Figure 4.

Technology	Additional attributes	Shannon Dimensions	Turing Dimensions
Wisdom Technology (Wistech)	Wisdom equation, i.e. wisdom = knowledge sources network + adaptive judgment + interactive processes	*Learn when to use information* *Learn how to get important information*	*How to make correct judgements (in. particular correct decisions) heaving in mind real-life constraints?*

Fig. 3. Computing machine technology (continued)

DIKW hierarchy can be traced back to the well-known poem by T. S. Eliot, "The Rock", written in 1932. He wrote:

Where is the life we have lost in living?
Where is the wisdom we have lost in knowledge?
Where is the knowledge we have lost in information?

It is a truism to state that the effects of any activity depend to a decisive degree on the wisdom of the decisions taken, both in the start, during the implementation, improvement and completion of the activity. The main objective of wistech is to automate support for the process leading to wise actions. These activities cover all areas of man's activities, from the economy, through medicine, education, research, development, etc.

In this context, one can clearly see how important role may have the wistech development in the future. The following comment from G.W. Leibniz on the idea to automate the processing of concepts representing thoughts should not surprise us either:

No one else, I believe, has noticed this, because if they had ... they would have dropped everything in order to deal with it; because there is nothing greater that man could do.

2.4 A General Approach to Wistech Computation Model

In order to create general Wistech computational models let us start an analysis of the concept of adaptive judgement.

For better familiarization of adaptive judgement we shall use the visualization of processes based on the IDEFO standard. Put simply, this means of visualisation is described in the diagram presented in Figure 5.

An intrinsic part of the concept of judgement is relating it to the entity implementing the judgement. Intuitively this can be a person, animal, machine, abstract agent, society of agents, etc. In general, we shall call the entity making

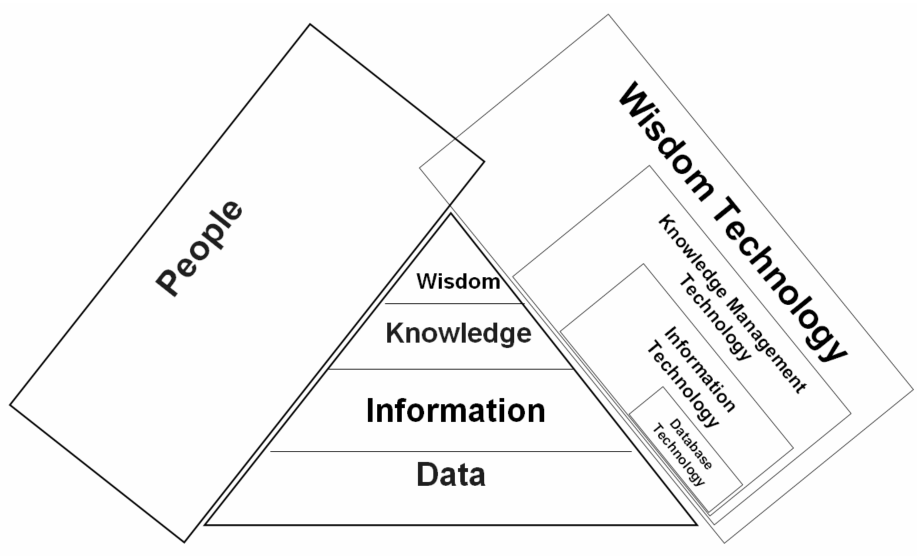


Fig. 4. DIKW hierarchy

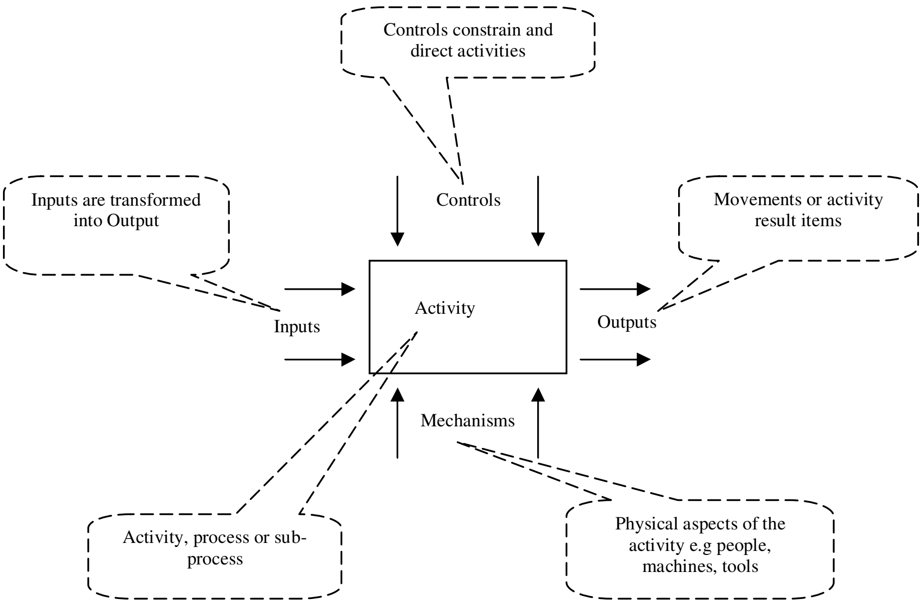


Fig. 5. Activity

a judgement the *judge*. We shall also assume that knowledge sources network is divided into external sources, i.e., sources of knowledge that are also available to other judges, internal sources, which are only available to the specific judge in question.

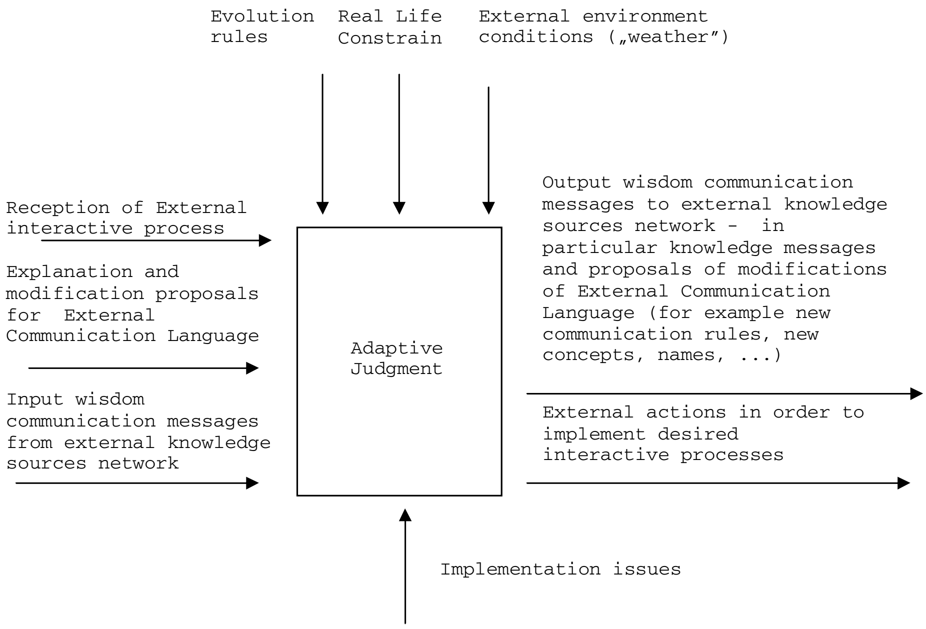


Fig. 6. The first level of the model

The first level of the model is presented in Figure 6. Of course, successive levels of the model are more complex. Its details may depend on the assumed paradigms for the implementation of adaptive judgement. However, these details should include such elements as:

1. Learning of the *External Communication Language* understood as a language based on concepts used to communicate and process knowledge with a network of external sources of knowledge;
2. Learning of the *Internal Communication Language* understood as a hierarchy of meta-languages based on concepts used to process and improve External Communication Language and a language based on concepts used to communicate and process knowledge with a network of internal sources of knowledge;
3. Receiving in memory signals from signal receptors and interactive processes and expressing their significance in the *External Communication Language and the Internal Communication Language*;
4. Planning the current priorities for internal actions (mainly related to the processing of wisdom) on the basis of an assessment in relation to the hierarchy of values controlling the adaptive judgement process;
5. Selection of fragments of ordered knowledge (hierarchies of information and judgement strategies) sufficient to take a decision at the planned time (a decision here is understood as commencing interaction with the environment or selecting the future course to resolve the problem);

6. Output wisdom communication messages to external knowledge sources network, in particular, knowledge messages and proposals of modifications of the External Communication Language (e.g., new communication rules, new concepts, names);
7. External actions in order to implement the desired interactive processes.

All elements occurring in the above list are very complex and important but the following two problems are particularly important for adaptive judgement computational models:

1. *Concept learning and integration* - this is the problem of computational models for implementation of learning concepts important for the representation, processing and communicating of wisdom and, in particular, this relates to learning of concepts improving the quality of approximation of the integration of incomplete local perceptions of a problem (arising during local assimilation and processing of vague and incomplete concepts (see, *e.g.*, [78,79])).
2. *Judge hierarchy of habit habits controls* - this is the problem of computational models for implementation of process of the functioning of a hierarchy of habit controls by a judge controlling the judgement process in an adaptive way.

Now, we sketch the idea of a framework for solution of the problem of implementation of judge hierarchy of habit controls. In this paper, we treat a concept of *habit* as an elementary and repeatable part of behavioral pattern. In this context, the meaning of elementary should be considered by comparison to the required reasoning (knowledge usage) complexity necessary for the behavioral pattern implementation. In other words, by a habit we mean any regularly repeated behavioral pattern that requires little or no reasoning effort (knowledge usage). In general, any behavioral pattern could be treated as a sequence of habits and other activities which use knowledge intensively. Among such activities those leading to new habits are especially important. We assume that such habit processing is controlled by so-called *habit controls* which support the following aspects of adaptive judgement process for a considered situation by a *judge*:

1. *Continuous habit prioritization* to be used in a particular situation after identification of habits. This is a prioritization from the point of view of the following three criteria:
   – The predicted consequences of the phenomena observed in a considered situation;
   – Knowledge available to a judge;
   – The actual plans of a judge's action.
2. *Knowledge prioritization* is used if we do not identify any habit to be used in a considered situation, then we have to make prioritization of pieces of available knowledge which could be used to choose the best habit or for a construction of a new habit for the considered situation.
3. *Habit control assessment* for continuous improvement of adaptive judgement process and for construction of new habits and habit controls.

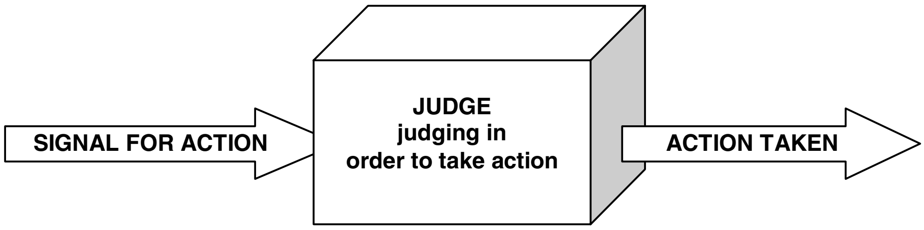


Fig. 7. Judge diagram

As it can be seen from the above considerations, one of the key components of wistech, judge hierarchy of habit control, is essential for optimal decision making and is closely correlated with the knowledge held and interactions with the environment. Judge hierarchy also means the desire of the judge to satisfy his/her needs in interactions with his/her environment. Put very simply, the judge receives and sends out signals according to the diagram presented in Figure 7.

The interior of the box is the place for the judge to process signals and to take an action. By the judge environment adaptation we understand the interaction of the following two adaptive processes:

1. *adaptation of the environment*, in which the judge *lives* to the *needs and objectives* of the judge so as to best fit the *needs and objectives* of the environment,
2. *adaptation of the internal processes taking place in a judge* in such a way as to best realize his/her *needs and objectives* based on the resources available in the environment.

The judge environment adaptation is the basis for computational models of judge learning. The key part of this is the evolution of judge hierarchy of habit controls. The judge hierarchy of habits controls constitutes a catalyst for evolutionary processes in the environment, and also constitutes an approach to expressing various paradigms of computation models to be used in the machine implementation of this concept. For example, these paradigms can be based on the metaphorically understood principle of Newtonian dynamics (e.g., action = reaction), thermodynamics (e.g., increase in entropy of information), quantum mechanics (the principle of it being impossible to determine *location and speed simultaneously*) and quantum computational models [44], psychology (e.g., based on metaphorical understanding of Maslow's hierarchy of needs; see also [53,80,40]). Particularly worthy of attention in relation to wistech is the metaphoric approach to Maslow's hierarchy of needs in reference to the abstractly understood community of agents. Put simply, this hierarchy looks as in Figure 8. It could be used for direct constructions of computational models of judge hierarchy of habit controls.

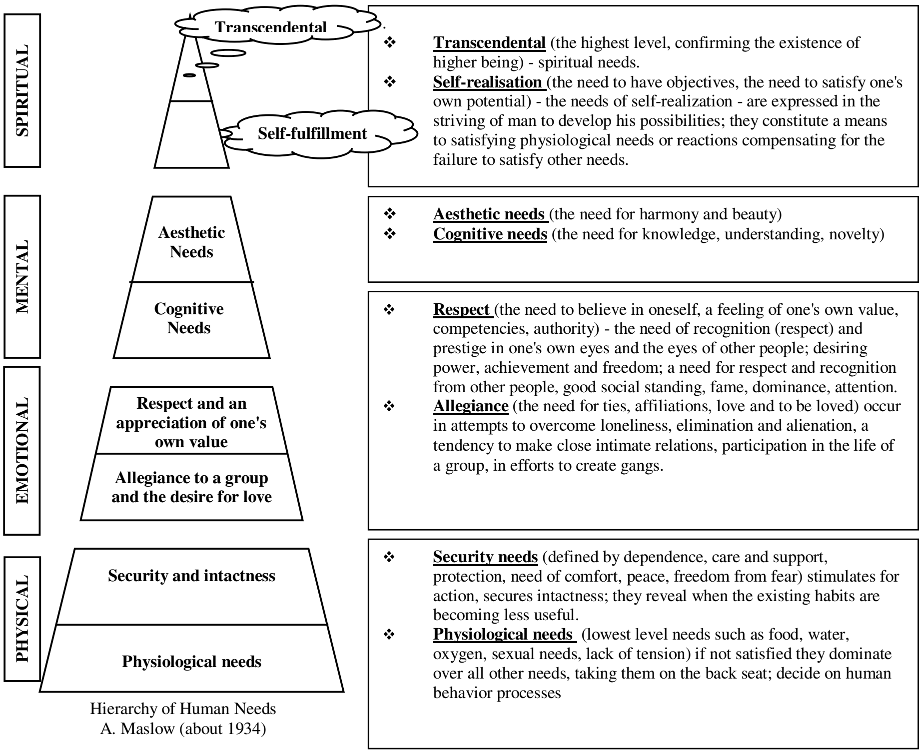


Fig. 8. The Maslow Hierarchy of human needs (about 1934) as an example of judge hierarchy of habit controls

2.5 A Rough-Granular Computing Approach to Wistech Computation Models

In this section, we outline basic ideas for the rough-granular approach to wisdom.

2.5.1 Evolution of Reasoning Computation Models in the Rasiowa–Pawlak School

By the Rasiowa–Pawlak school we mean a continuation of approaches to computational models of approximate reasoning developed by Rasiowa [86], Pawlak [74,87], and their students. In some sense, it is a continuation of ideas initiated by Leibniz, Boole and currently continued in a variety of forms over the world. Of course, the Rasiowa–Pawlak school is also some kind of continuation of the Polish School of Mathematics and Logics. The achievements of this school led to the development of the modern understanding of the basic computational aspects of logic, epistemology, ontology, foundations of mathematics and natural deduction (S. Banach, S. Eilenberg, R. Ingarden, S. Jaśkowski, K. Kuratowski,

S. Leśniewski, A. Lindenbaum, J. Łukasiewicz, S. Mazur, A. Mostowski, H. Rasiowa, R. Sikorski, W. Sierpiński, A. Tarski, S. Ulam, and many others). Two fundamental tools of the Rasiowa–Pawlak school are the following:

– *Computation models of a logical concept (especially of such concepts as deduction or algebraic many-valued models for classical, modal, and constructive mathematics).*
  The Rasiowa–Pawlak approach is based on the method of treating the sets of logically equivalent statements (or formulas) as abstract algebras known as the Lindenbaum–Tarski algebras.
– *Computation models of vague concept.*
  Łukasiewicz originally has proposed to treat uncertainty (or vague concepts) in logic as concepts of many-valued logic. However, software developed for today's computers is based on two-valued Boolean algebra. Therefore it is more practical to treat uncertainty and vagueness using the classical logic concept based on two-valued Boolean algebra. The concept of a rough set introduced by Pawlak [74] and developed in the Rasiowa–Pawlak school is based on the classical two-valued logic and, hence, the rough set approach is important and suitable for the applications mentioned above. The rough set approach intended to deal with uncertainty and vagueness has been developed to deal with uncertainty and vagueness. The rough set approach makes it possible to reason precisely about approximations of vague concepts. These approximations are tentative, subjective, and varying accordingly to changes in the environment [75,76,77,8].

Both the above mentioned fundamental tools can be applied in many contexts. It is interesting to illustrate evolution of the both above fundamental tools from the Rasiowa–Pawlak school perspective (see Figure 9 and Figure 10).

2.5.2 Rough-Granular Computing (RGC)

Solving complex problems by multi-agent systems in distributed environments requires approximate reasoning methods based on new computing paradigms. One such emerging recently computing paradigm is RGC. Computations in RGC are performed on information 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.

We discuss the rough-granular approach for modeling computations in complex adaptive systems and multiagent systems.

Information granules are any objects constructed when modeling of computations, and in approximating compound concepts, and approximate reasoning about these concepts. Information granules are constructed in an optimization process based on the minimal length principle. This process is aiming at constructing approximations of concepts satisfying some (vague and/or uncertain) constraints. Examples of information granules are information systems and decision systems, elementary information granules defined by indiscernibility neighborhoods, families of elementary granules (e.g., partitions and coverings),

Domain & Operators	*Natural Numbers Calculus*	*Algebra of subsets*	*Boolean Algebra*	*Logical concepts in Lindenbaum – Tarski algebra*	*Semantical models of constructive mathematics*	*Topoi*	*Wisdom Granular Computing for a given application domain*
X < Y	X is smaller than Y	X is a subset of Y	X is smaller than Y in Boolean algebra	Y can be deduced from Y	Logical value of X is smaller than logical value of Y in a Heyting algebra	Morphism from X to Y	Wisdom granule Y is a consequence of wisdom granule X in the domain
0	Zero	Empty set	The smallest element	Falsity	0 in a Heyting algebra	Initial element	The smallest wisdom granule in the domain
1	One	Full set	The biggest element	Truth	1 in a Heyting algebra	Final element	The biggest wisdom granule in the domain
+	Addition	Join of two sets	Maximum	Disjunction	Maximum	Coproduct	Relative coproduct of two wisdom granules
*	Multiplication	Intersection of two sets	Minimum	Conjunction	Minimum	Product	Relative product of two wisdom granules
X^Y	Exponentia-tion of X to power Y	Join of (–Y) and X	Join of (–Y) and X	Implication (Y implies X)	Relative pseudo – complementation Y→X in a Heyting algebra	Object corresponding to all morphisms from Y to X	Granule corresponding to all consequences from granule Y to granule X
Mod (X)	Modulo X calculus	Quotient algebra of the filter generated by set X	Quotient Boolean algebra of the filter generated by set X	Lindenbaum – Tarski algebra for a theory generated by a set of axioms X	Models for a theory generated by axioms X	Category of sheaves over X	All consequences from a given granule X
Logical values	True False	True False	True False	Algebra of logical values	Elements of Heyting Algebra	Subobject classifier	Identification of subgranules of granules

ANCIENT	**CONTEMPORARY**	**FUTURE**

Fig. 9. Evolution of computational models of logical concepts from the Rasiowa–Pawlak school perspective (the last column is hypothetical for a further research)

relational structures obtained by granulation of objects or classes of relational structures (representing structured objects and their classes), elementary and compound patterns (e.g., clusters of already defined patterns, hierarchical or behavioral patterns, protocols of cooperation), decision rules on different levels, interaction patterns, sets of decision rules, strategies of searching for relevant features, rough inclusions, approximation spaces, fusion operations on information granules, negotiation and conflict resolution strategies, classifiers constructed for compound and vague concepts. We discuss some aspects of rough set-based foundations for information granule calculi and methods for inducing relevant information granule constructions from data and background knowledge.

RGC has been applied for solving complex problems in areas such as identification of objects or behavioral patterns by autonomous systems, web mining, and sensor fusion (see, e.g., [3,5,6,7,8,22,68,69,93,94,95,98,99,100,101,102,105,106]).

2.5.3 Vague Concept Approximation

The RGC methods should make it possible to construct vague concept approximation and to perform approximate reasoning about such concepts. There is a long debate in philosophy on vague concepts [54]. Nowadays, computer scientists are also interested in vague (imprecise) concepts, e.g., many intelligent systems

Evolution of AI models of computing in the Rasiowa – Pawlak School

Mathematical concepts and tools

	Many-valued and nonclassical logic ⇒	**Approximate reasoning in distributed environments & natural computing: perception based computing**
Logic	➢ Computability, uncertainty natural deduction, algebraic semantics and language algebraic properties of different types of logic, especially in: • intuitionistic • modal • Post, intermediate • with strong negation • implicative • algorithmic • program • non Frege • with infinite logical operators ➢ Abstract logics, relationship between them and characterization of classical and other logics ➢ Hierarchy of metalogics ➢ Logical aspects programming paradigms ➢ Interpretation of logical operator in models of computation (generalized quantifiers, model operators, Post operators)	➢ Evolution of concepts: hierarchy of metalogics created by interactions with environment ➢ Society of agents, represented by a set of modal operators ➢ Consensus and emotional states as modal operators over non-classical "truth" values ➢ Logic for distributed systems ➢ Reasoning under uncertainty in distributed systems ➢ Vague concept approximation ➢ Boolean approximate reasoning: RSES ➢ Conflicts, negotiations, cooperation ➢ RS, FS, combination with nonmonotonic reasoning. ➢ Approximate reasoning about knowledge ➢ Common sense reasoning ➢ Perception logic: evolving system of interacting local logics ➢ Computational models based on perception ➢ Computational models of behavior ➢ Learning and adaptation ➢ Autonomous computing
Algebra	➢ Algebraic models for nonclassical and abstract predicate calculus (Q-algebras), generalization of Rasiowa – Sikorski Lemma ➢ Lattice theory, Boolean, Heyting, Brower, Post and other algebras ➢ Syntax and semantics as adjoint concepts ("Galois connections") ➢ Topos theory approach ➢ Internal representation of deduction by sheaves over closure spaces	➢ Algebraic structures for reasoning under uncertainty ➢ RS algebras, FS algebras ➢ Relational calculi ➢ Partial algebras ➢ Calculi of approximation spaces. ➢ Mereological calculi of information granules
Geometry	➢ Topological properties of spaces of models and concepts ➢ „Distance" between theories which represent knowledge of agents ➢ Geometry of computations ➢ Cantor Space, as a geometric space of models for classical propositional calculus ➢ Topological interpretation of modal operators ➢ Closure spaces as generalized geometric spaces ➢ Heuristics based on geometry of computation space	➢ Measures of proximity (similarity): states and set of states of computations and concepts ➢ Similarity of cases and case-based reasoning ➢ Geometry of concepts ➢ Similarity of theories ➢ Granular space, information granulation and granular computing ➢ Discovery of granularity levels from data, e.g., relevant multivalued logics
	Philosophy, CS, Biology, Psychology, Sociology, … ⇒	**Approximate reasoning about complex vague concepts and objects in distributed & dynamically changing environments**

Inspirations outside mathematics

Fig. 10. Evolution of computational models of vagueness from the Rasiowa–Pawlak school perspective

should satisfy some constraints specified by vague concepts. Hence, the problem of vague concept approximation as well as preserving vague dependencies (especially in dynamically changing environments) is important for such systems.

Lotfi Zadeh [120] introduced a very successful approach to vagueness. In this approach, sets are defined by partial membership in contrast to crisp membership used in the classical definition of a set.

Rough set theory [74] expresses vagueness not by means of membership but by employing the boundary region of a set. If the boundary region of a set is empty it means that a particular set is crisp, otherwise the set is rough (inexact). The non-empty boundary region of the set means that our knowledge about the set is not sufficient to define the set precisely. Inductive extensions of approximation spaces and adaptive concept learning yield better understanding of vague concepts [8]. A discussion on vagueness in the context of fuzzy sets and rough sets can be found in [88].

The central role of the rough set approach in RGC comes from the necessity of modeling different interactions between agents. These interactions are operations on information granules. Different agents use different languages to describe information granules. Information granules expressed in one language often cannot be expressed precisely in another language. Hence, there is the need for developing methods which could be used by judges in approximation of (vague) concepts (partially) specified by other judges. Initially, the approximation spaces, which can be treated as information granules, were introduced for decision tables (samples of objects). The assumption was made that the partial information about objects is given by values of attributes and that the approximations of subsets of objects from the universe restricted to sample have been defined on the basis of such information about objects [74]. Starting at least from the early 1990s, many researchers have been using the rough set approach for constructing classification algorithms (classifiers) defined over extensions of samples. This is based on the assumption that available information about concepts is partial.

In recent years, there have been reported methods based on approximation spaces and operations on approximation spaces to develop for approximation of concepts over the extensions of samples (see, e.g., [8,102]). Among the basic operations on extension of samples related to concept approximation are inductive extensions of approximation spaces (see, e.g., [8,102]). Neighborhoods of objects are the basic components of approximation spaces. They are defined by the available information about objects and rough inclusion functions between sets of objects. Observe that searching for relevant (to approximation of concepts) extensions of approximation spaces requires tuning more parameters than in the case of approximation of concepts on samples. The important conclusion is that the inductive extensions defining classification algorithms (classifiers) are defined by arguments "for" and "against" the concepts. Each argument is defined by a tuple consisting of a degree of inclusion of objects into a pattern and a degree of inclusion of the pattern into the concepts. In the case of rule-based classifiers, patterns can be interpreted as the left-hand sides of decision rules. The arguments are discovered from available data and can be treated as the basic information granules used in the concept approximation process. For any new object, it is possible to check the satisfiability of arguments and to select the arguments which are satisfied (at least to a satisfactory degree). Such selected arguments are fused by conflict resolution strategies for obtaining the classification decision.

Searching for relevant approximation spaces in the case of approximations over extensions of samples requires discovery of many parameters and patterns including selection of relevant attributes defining information about objects, discovery of relevant patterns for approximated concepts, selection of measures (similarity or closeness) of objects into the discovered patterns for concepts, the structure and parameters of conflict resolution strategy. This causes infeasibility of the searching process in the case of more compound concepts the searching process becomes infeasible (see, e.g., [12,32,81,116]).

We have proposed to use additional domain knowledge as hints in searching for relevant approximation spaces for compound concepts. This additional knowledge is represented by a concept ontology [3,5,6,7,8] including concepts expressed in natural language and some dependencies between them. We assume that the ontology of concepts has a hierarchical structure. Moreover, we assume that for each concept from the ontology a labeled set of examples of objects is given. The labels reflect the degree of membership of objects relative to the approximated concepts. The aim is to discover relevant conditional attributes for concepts on different levels of the hierarchy. Such attributes can be constructed using the so-called production rules, productions, and approximate reasoning schemes (AR schemes, for short) discovered from data (see, e.g. [3,5,6,7,8,100,101,102,105,106]). Searching for relevant arguments "for" and "against" for more compound concepts can be simplified because it can be organized along the derivations over the ontology using the domain knowledge.

It should be mentioned that the searching process for relevant approximation spaces is driven by some selected quality measures. While in some learning problems such measures can be selected in a relatively easy way and remain unchanged during learning, in other learning processes they can only be approximated on the basis of a partial information about such measures received, e.g., as the result of interaction with the environment. This applies to, e.g., adaptive learning. We present an example illustrating the complexity of the searching process for relevant approximation spaces in different tasks of adaptive learning [21]. In our recent projects, we develop methods for adaptation of observation to an agent's scheme, incremental learning, reinforcement learning, and adaptive planning.

Our discussion is presented within the framework of MAS. The main conclusion is that the approximation of concepts in adaptive learning requires new advanced methods for modeling of computations based on information granules. Among them are those which, in particular, will make it possible to approximate the quality measures together with approximation of concepts.

In adaptive learning, the approximation of concepts is constructed gradually and the tentative approximations change dynamically in the learning process where we try to achieve the approximation of the desired quality. In particular, this changes boundary conditions during the learning process in which we attempt to find the relevant approximation of the boundary regions of vague concepts. This is consistent with the requirement of the higher-order vagueness [54] stating that the borderline cases of vague concepts are not crisp sets. This paper is a continuation of our research (see, e.g., [3,5,6,7,8,68,69], [93,94,95,98,99], [100,101,102,105,106]) on approximation spaces and vague concept approximation processes conducted for several years.

In this paper, we concentrate on some issues of RGC relevant to adaptive processes [8,21,62,64]. In particular, we present some basic schemes relevant for adaptive concept learning. The aim is to illustrate the complexity of spaces in which RGC should enable us to construct relevant information granules and to develop searching methods for many compound kinds of relevant information granules. These information granules should make it possible to construct high quality approximation of concepts and to reason efficiently about performed computations. In such granular computations many compound information granules are involved.

Let us consider an example of the measure of approximation quality. When searching for relevant approximation of the compound concepts, methods for constructing appropriate measures are necessary. At a given step of the learning process, only partial information about such a measure is available. On the basis of such information we construct approximation of the measure and we use it for inducing approximation spaces relevant for concept approximation. However, at the next stages of the learning process, it may happen that after receiving new information from the environment, it will be necessary to reconstruct the approximation of the quality measure, and in this way we obtain a new "driving force" in searching for relevant approximation spaces during the learning process. Adaptive learning strategies create information granules of higher level. Evolutionary techniques for modeling computations on such information granules to synthesize relevant learning strategies (toward achieving the given goals) are critical for many applications. In the next two sections, we outline some basic concepts related to agents and their interactions. The agents are called judges to emphasize that among tasks they perform are judgements. We discuss two more examples.

2.5.4 Basic Concepts Relative to Judges

In this section, we discuss basic components of agents (called judges here) which perform their tasks. In particular, they interact with other agents in the environment, approximate vague concepts and have the ability of reasoning about concepts. Each judge can also be treated as an information granule. The basic concepts relative to the judge J are the following:

- *Information granules accessible by J.* N is the set of information granules accessible by J (e.g., neighborhoods of objects, sets defined by the left-hand sides of decision rules, sets defined by classifiers together with the classifiers, granules accessible by J through interaction with other judges, e.g., representing other sources of knowledge). Information granules are all constructive objects definable (accessible, generated) by the judge J which are used by J for representing knowledge, approximate reasoning, and interaction with other judges and/or the environment.

– *Goals (Targets) of* J. There are some goals (targets) by the judge J to be achieved, e.g., preservation of some constraints with some priorities or achievement of a state with a given property. G denotes the granule representing goals for J. In particular, constraints and targets are defined by means of information granules.
– *Environment of* J. ENV_J denotes the set of all judges interacting with J (directly or indirectly).
– *Information Function of* J. $Inf : States(ENV_J) \longrightarrow N$ is the information function about states of the environment from the set

$$States(ENV_J)$$

perceived by J. By N_{Inf} we denote the set $Inf(States(ENV_J))$.
– *Judgemental Strategies of* J. S is the set of judgemental strategies of J. Some examples of judgemental strategies are listed below. First let us introduce some notation. $\models^+_{deg}$, $\models^-_{deg}$ are binary relations in $N \times N$, called the rough inclusions of J, with the following intended meaning: $u \models^+_{deg} u'$ if and only if the granule u matches the granule u' to a degree at least deg; $u \models^-_{deg} u'$ if and only if the granule u matches the granule u' to a degree at most deg. For simplicity of reasoning we assume that $deg \in [0, 1]$. Assume that a set D of information granules (e.g., the set of decision classes for a rule-based classifier) is given. Let t_+, t_- be two thresholds from the interval $[0, 1]$ and let

$$N^+(u) = \{u' \in D : \exists deg > t_+(u \models^+_{deg} u')\}$$

for $u \in N_{Inf}$. Granule u votes "for" granules from $N^+(u)$ (relative to t_+) (see [46]). Let us assume

$$N^-(u) = \{u' \in D : \exists deg < t_-(u \models^-_{deg} u')\}$$

for $u \in N_{Inf}$. Then granule u votes "against" granules from $N^-(u)$ (relative to t_-). We assume that B is a distinguished set of information granules called behavioral patterns of J (e.g., decisions, actions, plans [34,115]) and $Lab : D \longrightarrow B$ is the (partial) labeling function assigning the behavioral patterns to (some) information granules from D.

  • S is one of the judgemental strategies of J making it possible to select a particular behavioral pattern as a reaction to the perceived information about the environment. In particular, S uses granules from $Lab(N^+(u))$ and $Lab(N^-(u))$, where $u = Inf(x)$ and x is the current state of the environment, and the labeling of these sets of granules by behavioral patterns. Observe that the strategy S should resolve conflicts arising due to the fact that information granules should satisfy some constraints. For example, some information granules cannot be matched by one information granule to a degree higher than a given threshold t_+.
  • *Quality strategy of* J. Q is the quality strategy of J for estimation of the closeness (similarity) between granules. The closeness estimation is

based on arguments "for" and "against" the satisfiability of the compound concept of "closeness" represented by Q. In this judgement J uses relevant granules from available granules representing knowledge accessible for J, often distributed among other judges, as well as the relationships between granules represented by matching degrees.

- *Adaptation strategy of* J*.* *Adap* is the adaptation strategy transforming a tuple

$$(N, G, Inf, B, Lab, \models^{+}_{deg}, \models^{-}_{deg}, S, Q)$$

  into a new such tuple. Observe that judgements performed by J during adaptation can, in particular, lead to construction of new granules (e.g., through cooperation with other judges [2]), changing some strategies such as the matching strategy, the labeling strategy, the selection strategy for relevant behavioral patterns, and the strategy for estimation of closeness of granules. *Adap* can also be changed, e.g., by tuning some of its parameters.

2.5.5 Basic Cycle of Judge

Each judge realizes some goals using behavioral patterns. The basic cycle of each judge J is the following:

1. *Step 1: Initialization.*
   $(N, G, Inf, B, Lab, \models^{+}_{deg}, \models^{-}_{deg}, S, Q) :=$
   $(N_0, G_0, Inf_0, B_0, Lab_0, \models^{+}_{deg,0}, \models^{-}_{deg,0}, S_0, Q_0).$
2. *Step 2: Perception granule construction by* J *representing the current state.*

$$u := Inf(x);$$

   where u is the granule representing perception by J of the current environment state x.
3. *Step 3:* J *selects the relevant granules from* $N^{+}(u)$*,* $N^{-}(u)$ *and performs judgements to select (construct) the relevant behavior* b *toward achieving the current goal (target).* During selection of b the judge J is also predicting the information $Inf_{pred}(b, x)$ returned from ENV_J as a reaction to the behavior b applied to the current state x of ENV_J. This is realized by another special judgemental strategy of J. By applying S to $Lab(N^{+}(u))$ and $Lab(N^{-}(u))$ J searches for a relevant behavior b.
4. *Step 4: Estimation of the closeness.*
   The judge J uses the quality measure Q for estimation of the closeness (similarity) of $Inf_{pred}(b, x)$ and $Inf_{real}(b, x)$ by

$$Q(Inf_{pred}(b, x), Inf_{real}(b, x)),$$

   where $Inf_{real}(b, x)$ is information about the real reaction of the environment in state x to the behavior b.

5. *Step 5: J uses a special judgemental strategy in testing whether the closeness is satisfactory.*
   If the closeness is satisfactory, then J continues from $Step2$; otherwise J goes to the next step.
6. Step 6: Adaptation step.
   $(N, G, Inf, B, Lab, \models^{+}_{deg}, \models^{-}_{deg}, S, Q) :=$
   $\quad Adapt(N, G, Inf, B, Lab, \models^{+}_{deg}, \models^{-}_{deg}, S, Q).$
7. Step 7: Go to Step 2.

All constructive objects involved in computations realized by means of the above judgement schemes are information granules.

2.5.6 Remark on Task Solving by Systems of Judges

The above examples illustrate the complexity and richness of the information granule spaces we deal with when modeling adaptive processes and reasoning about such processes. Systems of judges solve tasks by searching in the information granule spaces for information granules satisfying the task specification to a satisfactory degree (not necessarily exactly), i.e., matching information granules representing the task specification to a satisfactory degree. The requirement of "matching to a degree" used instead of "matching exactly" often makes searching for solutions feasible in information granule spaces [122].

In a number of papers (see, e.g., [99,105,106]), we have developed methods for construction of information granules (satisfying a given specification to a satisfactory degree) by means of operations on information systems called constrained sums. In particular, this approach proved to be general enough for modeling compound spatio-temporal information granules (e.g., information granules representing processes or behavioral patterns specified by vague concepts) and interactions between them.

3 Wistech Network (WN)

In this section, we discuss shortly the organization of cooperation for the projects based on wistech.

3.1 What We Mean by Wistech Network

The huge complexity of the problem of designing effective wistech computation models means that wistech progress significantly depends on forming effective and systematic cooperation between the numerous interdisciplinary teams verifying the Wistech calculation models developed in practical experiments. Moreover, in order to make a really essential progress in wistech it is important to involve the best possible specialists for making it possible to combine in wistech based projects knowledge of such areas as: psychology, sociology, ethics and domain dependent knowledge, e.g., neuroscience, medicine, economics, security, law, robotics, telecommunications, banking. This research, like all other research, requires a significant effort in other fundamental sciences, such as logic, epistemology, ontology, mathematics, computer science, philosophy and others. Of course such activity is very expensive. Moreover, in general, research of this type does not translate directly into economic results. No private company can afford to implement such extensive research by itself. It is also unlikely that there would be any significant commitment by government agencies in the coordination and development of research on such a wide scale. Unfortunately, current attempts at extending the international coordination of such type of research are not effective.

A dilemma therefore arises whether to develop wistech within the framework of expensive and highly risky closed research programs, or to support open programs in which the costs and risk are spread among many entities? It is our opinion that both directions are equally important and the key to the success is an environment for creating and developing harmony mechanisms between open and closed research (see [19]). In [19], among others, the contrasting principles of closed and open innovation are clarified (see Figure 11).

At the current stage of building an environment for creating and developing harmony mechanisms between open and closed research it is very important to develop a powerful framework for effective Open Innovation Wistech R&D network. The current stage of development in wistech above all requires the development of coordinated interdisciplinary basic research with a well-coordinated and easily accessible environment for experiments. Such activities are not possible in hermetically sealed companies, which are paralyzed by security procedures and guided by the criterion of rapid economic return. This is also why it is proposed to start up mechanisms for the systematized and relatively coordinated cooperation of centers interested in developing Wistech under a Wistech Network (WN) cooperating with one another in accordance with jointly perfected open principles based on Open Innovation Principles. It is worth stressing that organizations preferring Closed Innovation Principles may also draw great benefits from active participation in WN. This participation gives the possibility of testing solutions that have little chance of giving rapid market results, and also in the case of the appearance of such opportunities they can be translated into economic results in accordance with the principles accepted. At the same time, in the case of basic research, which in general does not translate directly into market effects, the understanding of progress in basic research gives greater opportunities for developing new market applications of one's own. A further great benefit of active participation in WN should be the possibility of comparing the various paradigms for building calculation models for Wistech. The times have long since gone when people believed that there is only one perfect paradigm in AI. Hybrid solutions adapted to the specific nature of the sphere of application dominate in applications. Hybrid applications themselves also use a variety of construction paradigms in a platform for integrating various approaches. Similarly we are also assuming that the WN environment would be represented in

Contrasting Principles of Closed and Open Innovation	
Closed Innovation Principles	**Open Innovation Principles**
The smart people in our field work for us.	Not all smart people work for us. We need to work with smart people inside *and* outside of our company.
To profit from R&D, we must discover it, develop it, and ship it ourselves.	External R&D can create significant value; internal R&D in needed to claim some portion of that value.
If we discover it ourselves, we will get it to the market first.	We don't have to originate the research to profit from it.
The company that gets an innovation to the market first will win.	Building a better business model is better than getting to the market first.
If we create the most and the best ideas in the industry, we will win.	If we make the best use of internal and external ideas, we will win.
We should control our intellectual properties (IP), so that our competitors don't profit from our ideas.	We should profit from others' use of our IP, and we should buy others' IP whenever it advances our own business model.

Fig. 11. Contrasting principles of closed and open innovation [19]

the form of a sub-network with various paradigms for the construction of an integration platform. WN would provide the data and criteria to assess the results of experiments used for the assessment of various paradigms. In the remainder of this work we present, among others, a proposal to start up a sub-network based on a paradigm for the integration of various technologies based on an *adaptive rough-granular computing approach* (RGC).

3.2 A Potential Example Scenario of WN Establishment

3.2.1 WN Long-Term Vision and Role

The basic objectives of WN are supporting open innovation and the development of wistech and its applications through:

1. creating new paradigms and trends in Wistech and its applications,
2. creating a platform (e.g. intranet, symposia, training programs, e-learning, etc.) for communication and the exchange of knowledge and experience on the practical applications and achievements of basic research,
3. preparing educational and research programs,
4. starting up projects for specific practical applications, as well as for basic research,

5. establishing the conditions and criteria used to compare the quality of various approaches to Wistech (especially having in mind applications in medicine, economy, agriculture, energy and forex market),
6. popularization of Wistech.

3.2.2 WN Organization and Financial Support

We assume that participation in WN is absolutely voluntary in nature, and WN itself also does not assume any additional financial fees or obligatory participation in conferences. The organization is open in nature and any person or organization can take part in it. The form of organization is based on communities cooperating together, which jointly use and develop open software (see, e.g., http://www.opensource.org/).

At the same time we assume that at some stage WN may take part in commercial projects. The project participants will mutually agree upon the principles for cooperation in every such case. It is expected that in the long-term some products or components created by WN will function according to the principles of open software (e.g. similar to the principles of http://www.opensource.org/). We continue to assume the organization of working groups in the network which would deal with jointly agreed packets of problems and projects.

It is expected in our exemplary scenario that WN will develop in accordance with the stages for development of a mature organization modeled on the ideas of Carnegie Mellon Capability Maturity Model (http://www.sei.cmu.edu/cmm/). This model consists of the six stages presented in Figure 12 and Figure 13.

The basic assumption to WN is the realization of projects financed by WN participants who cover the costs and risk of their own activities in the network. It is also assumed that in WN there will be several specialist centers which will coordinate the activities in individual areas (competency centers), e.g. the multi-agent approach, the rough mereology approach. The coordination work of these centers would be financed from voluntary financial contributions from participants of the group in question. It follows from this that the intensity and quality of work in a given group will to a large degree depend on the level of financial support from participants in the group.

4 Wisdom Engine

We discuss some exemplary projects proposed as pilot projects in development of wistech.

4.1 Wisdom Engine Concept

By wisdom engine we understand a machine system which implements the concept of wisdom. In other words, the basic functions of the wisdom engine would be acquiring, processing, discovering, learning and communicating wisdom. One of the main first objectives of WN can be to create an open international R&D environment for the design and implementation of the concept of *universal domain-independent wisdom engine*. A universal wisdom engine implementation should

Name of stage	Organization	Content
ESTABLISHMENT	Starting up the first projects in the network and defining the principles for the cooperation of the first group of participants who confirm their participation in WN. Starting up the first forms of communication.	Developing the initial catalogue of paradigms for approaches to development of wistech (e.g., multi-agents, evolution, symbolic processing, neural nets, statistics, **adaptive rough granular approach**, formal concepts, ontology engineering, information semiotics, cognitive and epistemological approach, etc., and their combinations).
INITIAL	Developing a common language to describe the concepts relating to starting up, implementing and closing projects in WN.	The preliminary allocation of categorized paradigms for approaches to wistech to their respective competency centers. Allocating a paradigm to a competency center, e.g. multi-agent approach, **adaptive rough granular approach**, etc. This does not mean that at a given competency center only and exclusively this method will be developed. On the contrary, it is assumed that every competency center will develop hybrid solutions combining various approaches. At the same time, a competency center will particularly strongly develop aspects relating to the paradigms allocated to this center.

Fig. 12. Six stages of the Carnegie Mellon Capability Maturity Model

be independent of any specific application domain. At the same time, functionality of the universal wisdom engine should enable the configuration and tuning of modules for it in the form of a series of products dependent on specific application domains such as, e.g., medicine, economics, stock market, forex market, security, law, tourism, telecommunications, banking, job market. In particular universal wisdom engine should be able to learn domain knowledge by reading, discussing with experts and gathering wisdom from experience. Of course, the design and implementation of a universal wisdom engine is an extremely difficult task and probably unrealistic today in a short term. First of all, we have to do some experiments with several application domains and several different paradigms for wistech implementation. Based on an analysis of the results of such experiments we can create a more general wistech ontology which should provide a better formal framework for the implementation of a universal wisdom engine.

REPEATABLE	Establishing the principles for selecting good practices specific for the implementation of a project in wistech, designed to repeat the successes of projects realized in similar conditions and to avoid failures. Establishing the list of first conditions and criteria used to compare the quality of various approaches to wistech.	Establishing the mutually tied objectives to achieve at the individual competency centers in order to verify the effectiveness and possibilities of developing various approaches.
DEFINED	Putting in writing and the effective implementation of a list of joint standards for organization and management of projects specific to wistech, that will be binding for the WTN community.	Starting up the first projects realized in the common standards by a variety of centers within the WN
MEASURABLE	Enhancing the standards arising at the previous stage to include sets of measurable indices used to verify and optimize the benefits to costs of wistech projects.	Starting up mechanisms for competitiveness between communities working on various approaches to wistech in the network.
CONTINUOUS IMPROVEMENT	Enhancing the standards and indices defined at the MEASURABLE stage to set out in writing and effectively implement procedures for continuously improving the functioning of WN.	Developing the optimum methods for harmonious co-operation between WN and commercial companies.

Fig. 13. Six stages of the Carnegie Mellon Capability Maturity Model (continued)

Thus, it is assumed that in parallel with the work on a universal concept of a wisdom engine, work would also be conducted on utilizing the wisdom engine in selected areas of application, e.g., medicine, economics, stock market, forex market, security, law, tourism, telecommunications, banking, or job market. The long-term vision is as follows: "wisdom engineers" will receive the task to create the configuration for the wisdom engine for applications in a specific field of life, and then, after having carried out the necessary analytical and design work, to configure the wisdom engine and to enter the necessary initial data. The wisdom engine should have properties for self-growth and adaptation to changing conditions of its environment, as well as advances in wisdom in the fields of application. This is why one should strongly emphasize the planned property of automatic adaptation of the system – a feature not taken into account in the construction of the numerous systems in the past that were intended to perform similar tasks. A classic example here is the long-standing MYCIN project implemented by Stanford University.

The implementation of the idea expressed by the wisdom equation is very difficult and it would be unreasonable to expect its full implementation in a short period of time. We assume that the creativity cycle for the first product prototypes implementing this concept would take several years of intensive work with cooperation of product managers, scientists, engineers, programmers and domain experts. On the other hand, it is not desirable to implement such long projects without any clear interim effects. This is why we assume that the wisdom engine implementation project would go through several phases. For example, initially we assume they will go through five phases in the implementation of the wisdom engine. We propose a route, to achieving the target wisdom engine products through continuously improving intermediary products that meet successive expansions in functionality. The five phases are called as follows:

1. Summary,
2. Spider,
3. Conceptual Clustering and Integration,
4. Wisdom Extraction,
5. Wisdom Assistant.

The effect of each of these phases will be a prototype product that after acceptance would be interesting for the WN community.

Stated in simple terms the functional effects of the individual phases would be as presented in Figure 14.

4.2 Examples of Wisdom Engine Domain-Dependent Product Lines

The above five phases (i.e., Summary, Spider, Conceptual Clustering and Integration, Wisdom Extraction, and Wisdom Assistant) should be applied to several directions for potential product lines which would be developed in the WN. Of course, it can theoretically be any product relating to applications in robotics, unmanned aircraft, space rockets, etc. However, if we wish to have as many people as possible cooperating in the WN, then the product lines must be chosen so that experimenting with them does not prove expensive. On the other hand, these product lines must be sufficiently attractive so as to interest as many people as possible. We propose that these product lines relate to applications in such areas as medicine, economics, the stock market, forex market, security, law, tourism, telecommunications, banking, job market and others.

The list of products that could be expanded in accordance with the above scheme is potentially unlimited. The proposals for the descriptions of specific

Phase	Summary	Spider	Conceptual Clustering and Integration	Adaptive Wisdom Extraction	Adaptive Wisdom Assistant
Key functions	key concept extraction	document searching related to key concept and indexing documents	document clustering based on key concept and Integration	extracting data structures, information structures, knowledge structures and adaptive wisdom structures from documents, in particular generating thesauruses, conceptual hierarchies and constraints to acceptable solutions	user query / answering processing in order to support users in solving their problems as effectively as possible

Fig. 14. Functional effects of the individual phases

Phase/Product	Summary	Spider	Conceptual Clustering	Wisdom Extraction	Wisdom Assistant
Document Manager	Document Summary	Document Spider	Document Conceptual Clustering	Document adaptive wisdom Extraction	Document adaptive wisdom Assistant
Job Market	Job Market Summary	Job Market Spider	Job Market Conceptual Clustering	Job Market adaptive wisdom Extraction	Job Market adaptive wisdom Assistant
Brand Monitoring	Brand Monitoring Summary	Brand Monitoring Spider	Brand Monitoring Conceptual Clustering	Brand Monitoring adaptive wisdom Extraction	Brand Monitoring adaptive wisdom Assistant
World Communication	World Communication Summary	World Communication Spider	World Communication Conceptual Clustering	World Communication adaptive wisdom Extraction	World Communication adaptive wisdom Assistant
World Forex	World Forex Summary	World Forex Spider	World Forex Conceptual Clustering	World Forex adaptive wisdom Extraction	World Forex adaptive wisdom Assistant
World Stock Market	World Stock Market Summary	World Stock Market Spider	World Stock Market Conceptual Clustering	World Stock Market adaptive wisdom Extraction	World Stock Market adaptive wisdom Assistant
World Tourist	World Tourist Summary	World Tourist Spider	World Tourist Conceptual Clustering	World Tourist adaptive wisdom Extraction	World Tourist adaptive wisdom Assistant
Physician	Physician Summary	Physician Spider	Physician Conceptual Clustering	Physician adaptive wisdom Extraction	Physician adaptive wisdom Assistant
Lawyer	Lawyer Summary	Lawyer Spider	Lawyer Conceptual Clustering	Lawyer adaptive wisdom Extraction	Lawyer adaptive wisdom Assistant
Economy Monitoring	Economy Monitoring Summary	Economy Monitoring Spider	Economy Monitoring Conceptual Clustering	Economy Monitoring adaptive wisdom Extraction	Economy Monitoring adaptive wisdom Assistant

Fig. 15. Proposed products

products, included in the later part of this report, should be treated as flexible and primarily constitute material for discussion, and not a final decision. On the other hand, the list of products described is not entirely accidental in nature.

Product / Phase	Summary	Spider	Conceptual Clustering	Adaptive Wisdom Extraction	Adaptive Wisdom Assistant
Document Manager	automatic summarizing of a document and groups of documents, the contents of which are not connected with any specific field	automatic searching and downloading of any documents	conceptual clustering of documents on any subject	extracting data structures, information structures, knowledge structures and adaptive wisdom structures from documents, in particular generating thesauruses, conceptual hierarchies and constraints to acceptable solutions in any domain	general user query / answering processing in order to support users in solving their problems as effectively as possible
Job Market	automatic summarizing of a document and groups of documents relating to job market, carried out from the perspective of the following groups of users: potential employers and potential employees	automatic searching and downloading of documents relating to job market, carried out with particular emphasis on the needs of the following groups of users: potential employers and potential employees	conceptual clustering of documents relating to job market with particular emphasis on the specific nature of queries submitted by the following types of users: potential employers and potential employees	extracting data structures, information structures, knowledge structures and adaptive wisdom structures from documents, in particular generating thesauruses, conceptual hierarchies and constraints to acceptable solutions in job market domain	job market related to user query / answering processing in order to support users in solving their problems as effectively as possible

Fig. 16. Functionality of individual products

This is because they form a certain logical continuity, connected both with the degree of difficulty in successive products and current preferences resulting from the previous experiences of the human resources that would be engaged to carry out the work on individual products. The initial selection of product lines is as follows:

- Document Manager,
- Job Market,
- Brand Monitoring,
- World Communication,
- World Forex,
- World Stock Market,
- World Tourist,
- Physician,
- Lawyer,
- Economy Monitoring.

Product / Phase	Summary	Spider	Conceptual Clustering	Adaptive Wisdom Extraction	Adaptive Wisdom Assistant
Brand Monitoring	automatic summarizing of a document and groups of documents relating to brand, carried out from the perspective of the following groups of users: brand owners, detectives looking for frauds, buyers	automatic searching and downloading of documents relating to brand, carried out with particular emphasis on the needs of the following groups of users: brand owners, detectives looking for frauds, buyers	conceptual clustering of documents relating to brand with particular emphasis on the specific nature of queries submitted by the following types of users: brand owners, detectives looking for frauds, buyers	extracting data structures, information structures, knowledge structures and adaptive wisdom structures from documents, in particular generating thesauruses, conceptual hierarchies and constraints to acceptable solutions in brand monitoring domain	brand monitoring related to user query / answering processing in order to support users in solving their problems as effectively as possible
World Communication	automatic summarizing of a document and groups of documents relating to communication , carried out from the perspective of people looking for optimal connections	automatic searching and downloading of documents relating to communication carried out with particular emphasis on the needs of the following groups of users: people looking for optimal connections	conceptual clustering of documents relating to communication with particular emphasis on the specific nature of queries submitted by the following types of users: people looking for optimal connections	extracting data structures, information structures, knowledge structures and adaptive wisdom structures from documents, in particular generating thesauruses, conceptual hierarchies and constraints to acceptable solutions in communication domain	communication related to user query / answering processing in order to support users in solving their problems as effectively as possible

Fig. 17. Functionality of individual products (continued)

This initial selection for the product list generates several dozen of products that would be the effect of work on the individual phases of implementing each of the products, i.e., Summary, Spider, Conceptual Clustering, Wisdom Extraction, Wisdom Assistant. We present the proposed products in Figure 15.

The scope of the program described in this paper should be considered as dynamic and more as a basis for further discussion than a final version of the specific definitions of the projects. This is why the innovative ideas presented and the vision for their implementation do not contain any detailed cost benefits analysis. It will only be possible to specify revenues, costs and cash flow forecasts with any accuracy after the planned scope of work and the role of the WN has stabilized. As there are as yet no final decisions on the scope of operations or role of the WN, this means that at the current stage it is impossible to precisely estimate the planned requirements for human resources. This is why in this document we present only the general human resources requirements and a description of the general mechanisms for acquiring these resources to implement WN.

Each of these products would have their own individual functionality which would result from adapting the wisdom engine to the specific characteristics of their specialist fields. Figure 16 and Figure 17 show the functionality of the individual products.

5 Conclusions

We have discussed the main features of wistech and its importance for further progress in the development of intelligent systems. The proposed approach is based on Rough Granular Computing (RGC).

One of the central problems of science today is to develop methods for approximation of compound vague concepts and approximate reasoning about them [32,81].

Today, we do not have yet satisfactory tools for discovery of relevant patterns for approximation of compound concepts directly from sample objects. However, we have developed methods for compound concept approximation using sample objects and domain knowledge acquired from experts (this is the approach pioneered by Zdzisław Pawlak in [73]). The performed experiments based on approximation of concept ontology (see, e.g., [3,5,6,7,8,22,68,69,70], [78,79,93,94,95,98,99], [100,101,102,105,106]) showed that domain knowledge enables to discover relevant patterns in sample objects for compound concept approximation. Our approach to compound concept approximation and approximate reasoning about compound concepts is based on the rough-granular approach.

One of the RGC challenges is to develop approximate reasoning techniques for reasoning about dynamics of distributed systems of judges. These techniques should be based on systems of evolving local perception logics rather than on a global logic [94,95]. Approximate reasoning about global behavior of judges' system is infeasible without methods for approximation of compound vague concepts and approximate reasoning about them. One can observe here an analogy to phenomena related to the emergent patters in complex adaptive systems [21].

Let us observe that judges can be organized into a hierarchical structure, i.e., one judge can represent a coalition of judges in interaction with other agents existing in the environment [2,56,62]. Such judges representing coalitions play an important role in hierarchical reasoning about behavior of judges' populations. Strategies for coalition formation and cooperation [2,62,64] are of critical importance in designing systems of judges with dynamics satisfying to a satisfactory degree a given specification. Developing strategies for discovery of information granules representing relevant (for the given specification) coalitions and cooperation protocols is another challenge for RGC.

RGC will become more and more important for analysis and synthesis of the discussed compound adaptive processes.The impact of RGC on real-life applications will be determined by techniques based on the rough-granular approach to modeling of relevant computations on compound information granules and methods for approximate reasoning about complex adaptive processes over such information granules. RGC techniques for modeling of complex processes will also have impact on the development of new non-conventional computation models.

Acknowledgments

The research of Andrzej Jankowski was supported by Institute of Decision Process Support. The research of Andrzej Skowron has been supported by the grant 3 T11C 002 26 from Ministry of Scientific Research and Information Technology of the Republic of Poland.

Many thanks to Professors James Peters and Anna Gomolińska for their incisive comments and for suggesting many helpful ways to improve this article.

References

1. G. Antoniou, F. van Harmelen, A Semantic Web Primer (Cooperative Information Systems) The MIT Press, 2004.
2. R. Axelrod, The Complexity of Cooperation, Princeton, NJ: Princeton University Press, 1997.
3. A. Bargiela, W. Pedrycz, Granular Computing: An Introduction, Dordrecht: Kluwer Academic Publishers, 2003.
4. J. Barwise, J. Seligman, Information Flow: The Logic of Distributed Systems, Cambridge University Press, 1997.
5. J. G. Bazan, J. F. Peters, and A. Skowron, Behavioral pattern identification through rough set modelling, in [71], pp. 688–697.
6. J. Bazan, A. Skowron, On-line elimination of non-relevant parts of complex objects in behavioral pattern identification, in [111], pp. 720–725.
7. J. Bazan, A. Skowron, Classifiers based on approximate reasoning schemes, in Monitoring, Security, and Rescue Tasks in Multiagent Systems (MSRAS 2004), B. Dunin-Keplicz, A. Jankowski, A. Skowron, and M. Szczuka, Eds., Advances in Soft Computing, pp. 191-202, Heidelberg: Springer, 2005.
8. J. Bazan, A. Skowron, R. Swiniarski, Rough sets and vague concept approximation: From sample approximation to adaptive learning, Transactions on Rough Sets V: Journal Subline, Lecture Notes in Computer Science, vol. 3100, pp. 39-63, Heidelberg: Springer, 2006.
9. R. Baeza-Yates, B. Ribeiro-Neto, Modern Information Retrieval, Addison Wesley, 1999.
10. M. W. Berry, Survey of Text Mining : Clustering, Classification, and Retrieval, Springer, 2003.
11. R. Brachman, H. Levesque, Knowledge Representation and Reasoning, Morgan Kaufmann, 2004.
12. L. Breiman, Statistical modeling: The two Cultures, Statistical Science 16(3) (2001) 199–231.

13. S. Brin, L. Page, The Anatomy of a Large-Scale Hypertextual Web Search Engine, Stanford University, 1998.
14. C. Carpineto, G. Romano, Concept Data Analysis: Theory and Applications, John Wiley & Sons, 2004.
15. N. L. Cassimatis, A cognitive substrate for achieving human-level intelligence, AI Magazine 27(2) (2006) 45-56.
16. N. L. Cassimatis, E. T. Mueller, P. H. Winston, Achieving human-level intelligence through integrated systems and research, AI Magazine 27(2) (2006) 12-14.
17. S. Chakrabarti, Mining the Web: Analysis of Hypertext and Semi Structured Data, The Morgan Kaufmann Series in Data Management Systems, Morgan Kaufmann, 2002.
18. Mu-Y. Chen, An-P. Chen, Knowledge management performance evaluation: a decade review from 1995 to 2004, Journal of Information Science 32 (1) 2006 17-38.
19. H. W. Chesbrough, Open Innovation: The New Imperative for Creating and Profiting from Technology, Cambridge MA: Harvard Business School Publishing, 2003.
20. J. Coleman, Introducing Speech and Language Processing (Cambridge Introductions to Language and Linguistics), Cambridge University Press, 2005.
21. A. Desai, Adaptive complex enterprises, Communications ACM 48(5) (2005) 32-35.
22. P. Doherty, W. Łukaszewicz, A. Skowron, and A. Szałas, Knowledge Engineering: A Rough Set Approach, Studies in Fuzziness and Soft Computing, vol. 202, Heidelberg: Springer, 2006.
23. R. Dornfest, T. Calishain, Wistech Network Hacks O'Reilly Media, Inc.; 2004.
24. R. Duda, P. Hart, and R. Stork, Pattern Classification, New York, NY: John Wiley & Sons, 2002.
25. B. Dunin-Kęplicz, A. Jankowski, A. Skowron, M. Szczuka, Monitoring, Security, and Rescue Tasks in Multiagent Systems (MSRAS'2004), Series in Soft Computing, Heidelberg: Springer, 2005.
26. A. E. Eiben, J. E. Smith, Introduction to Evolutionary Computing, Natural Computing Series, Springer, 2003.
27. E. Feigenbaum, J. Feldman (Eds.), Computers and Thought, New York: McGraw Hill, 1963.
28. S. Feldman, Why Search is Not Enough (white paper), IDC, 2003.
29. S. Feldman, Enterprise Search Technology: Information Disasters and the High Cost of Not Finding Information (Special IDC Report), IDC, 2004.
30. D. Fensel, Ontologies: A Silver Bullet for Knowledge Management and Electronic Commerce, Springer, 2003.
31. K. D. Forbus, T. R. Hinrisch, Companion cognitive systems: A step toward human-level AI, AI Magazine 27(2) (2006) 83-95.
32. M. Gell-Mann, The Quark and the Jaguar, NY: Freeman and Co., 1994.
33. J. H. Friedman, T. Hastie, R. Tibshirani, The Elements of Statistical Learning: Data Mining, Inference, and Prediction, Heidelberg: Springer, 2001.
34. M. Ghallab, D. Nau, and P. Traverso, Automated Planning: Theory and Practice, CA: Morgan Kaufmann, 2004.
35. S. Ghemawat, H. Gobioff, Shun-Tak Leung, The Wistech Network File System, Wistech Network, 2005.
36. D. E. Goldberg, Genetic Algorithms in Search, Optimization, and Machine Learning, Addison-Wesley Professional, 1989.

37. A. Gomez-Perez, O. Corcho, M. Fernandez-Lopez, Ontological Engineering with examples from the areas of Knowledge Management, e-Commerce and the Semantic Web (Advanced Information and Knowledge Processing), Springer, 2004.
38. R. Granger, Engines of the brain: The computational instruction set of human cognition, AI Magazine 27(2) (2006) 15-31.
39. S. Grimes, The Developing Text Mining Market, A white paper prepared for Text Mining Summit 2005, Boston, June 7-8 2005, Alta Plana Corporation, 2005.
40. A. OHagan, C. E. Buck, A. Daneshkhah, J. R. Eiser, P. H. Garthwaite, D. J. Jenkinson, J. E. Oakley, T. Rakow, Uncertain Judgements: Eliciting Expert Probabilities, Wiley, New York, 2006.
41. J. Heaton, Programming Spiders, Bots, and Aggregators in Java, Sybex, 2002.
42. K. Hemenway, T. Calishain, Spidering Hacks O'Reilly Media, Inc.; 2003.
43. M. Henzinger, S. Lawrence, Extracting knowledge from the World Wide Web, Wistech Network, 2004.
44. M. Hirvensalo, Quantum Computing, Springer-Verlag, Heidelberg 2001.
45. M. N. Huhns, M. P. Singh, Readings in Agents, Morgan Kaufmann, 1998.
46. P. Jackson, I. Moulinier, Natural Language Processing for Online Applications: Text Retrieval, Extraction, and Categorization (Natural Language Processing, 5), John Benjamins Publishing Co, 2002.
47. Z. Janiszewski, On needs of mathematics in Poland (O potrzebach matematyki w Polsce) (in Polish), In: Nauka Polska. Jej Potrzeby, Organizacja i Rozwój, Warszawa, 1918; see also reprint in Wiadomości Matematyczne VII (1963) 3-8.
48. A. Jankowski, An alternative characterization of elementary logic, Bull. Acad. Pol. Sci., Ser. Math. Astr. Phys.XXX (1-2) (1982) 9-13.
49. A. Jankowski, Galois structures, Studia Logica 44(2) (1985) 109-124.
50. S. Johnson, Dictionary of the English Language in Which the Words are Deduced from Their Originals, and Illustrated in their Different Significations by Examples from the Best Writers, 2 Volumes. London: F.C. and J. Rivington, 1816.
51. R. M. Jones, R. E. Wray, Comparative analysis of frameworks for knowledge-intensive intelligent agents, AI Magazine 27(2) (2006) 57-70.
52. D. Jurafsky, J. H. Martin, Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics and Speech Recognition, Prentice Hall, 2000.
53. D. Kahneman, P. Slovic, A. Tversky, A. (Eds.). Judgement under Uncertainty: Heuristics and Biases, Cambridge University Press, New York, 1982.
54. R. Keefe, Theories of Vagueness, Cambridge, UK: Cambridge Studies in Philosophy, 2000.
55. W. Kloesgen, J. Żytkow, Handbook of Knowledge Discovery and Data Mining, New York: Oxford University Press, 2002.
56. S. Kraus, Strategic Negotiations in Multiagent Environments, Massachusetts: The MIT Press, 2001.
57. J. Lambek, P. J. Scott, Introduction to Higher-Order Categorical Logic (Cambridge Studies in Advanced Mathematics 7), Cambridge University Press, 1986.
58. M. Lamb, Build Your Own Army of Web Bots Within 24 Hours (Army of Web Bots Series) Authorhouse, 2003.
59. P. Langley, H. A. Simon, G. L. Bradshaw, J. M. Zytkow, Scientific Discovery: Computational Explorations of the Creative Processes, MIT Press, 1987.
60. P. Langley, Cognitive architectures and general intelligent systems, AI Magazine 27(2) (2006) 33-44.
61. G. W. Leibniz, New Essays on Human Understanding, Cambridge UP, 1982.

62. J. Liu, Autonomous Agents and Multi-Agent Systems: Explorations in Learning, Self-Organization and Adaptive Computation, Singapore: World Scientific Publishing, 2001.
63. J. Liu, L. K. Daneshmend, Spatial Reasoning and Planning: Geometry, Mechanism, and Motion, Springer, 2003, Hardcover.
64. J. Liu, X. Jin, K. Ch. Tsui, Autonomy Oriented Computing: From Problem Solving to Complex Systems Modeling, Heidelberg: Kluwer Academic Publisher/Springer, 2005.
65. S. MacLane, I. Moerdijk, Sheaves in Geometry and Logic: A First Introduction to Topos Theory (Universitext), Springer, 1994.
66. A. Madhavapeddy, N. Ludlam, Ubiquitious Computing needs to catch up with Ubiquitous Media, University of Cambridge Computer Laboratory, Interceptor Communications Ltd., 2005.
67. C. D. Manning, H. Schütze, H., Foundations of Statistical Natural Language Processing, The MIT Press, 1999.
68. S. H. Nguyen, J. Bazan, A. Skowron, and H. S. Nguyen, Layered learning for concept synthesis, Transactions on Rough Sets I: Journal Subline, Lecture Notes in Computer Science, vol. 3100, pp. 187-208, Heidelberg: Springer, 2004.
69. T. T. Nguyen, Eliciting domain knowledge in handwritten digit recognition, in [71] 762-767.
70. S. K. Pal, L. Polkowski, and A. Skowron (Eds.), Rough-Neural Computing: Techniques for Computing with Words, Cognitive Technologies, Heidelberg: Springer-Verlag, 2004.
71. S. K. Pal, S. Bandoyopadhay, and S. Biswas (Eds.), Proceedings of the First International Conference on Pattern Recognition and Machine Intelligence (PReMI'05), December 18-22, 2005, Indian Statistical Institute, Kolkata, Lecture Notes in Computer Science vol. 3776, Heidelberg: Springer, 2005.
72. T. B. Passin, Explorer's Guide to the Semantic Web Mining Publications, 2004.
73. Pawlak, Z.: Classification of objects by means of attributes, Research Report PAS 429, Institute of Computer Science, Polish Academy of Sciences, ISSN 138-0648, January (1981).
74. Z. Pawlak, Rough Sets: Theoretical Aspects of Reasoning about Data, System Theory, Knowledge Engineering and Problem Solving 9, Dordrecht: Kluwer Academic Publishers, 1991.
75. Z. Pawlak, A. Skowron, Rudiments of rough sets. Information Sciences. An International Journal. 177(1) (2007) 3-27.
76. Z. Pawlak, A. Skowron, Rough sets: Some extensions. Information Sciences. An International Journal. 177(1) (2007) 28-40.
77. Z. Pawlak, A. Skowron, Rough sets and Boolean reasoning. Information Sciences. An International Journal. 177(1) (2007) 41-73.
78. J. F.Peters, Rough ethology: Toward a biologically-inspired study of collective behavior in intelligent systems with approximation spaces. Transactions on Rough Sets III: Journal Subline, Lecture Notes in Computer Science, vol. 3400, pp. 153-174, Heidelberg: Springer, 2005.
79. J. F. Peters, C. Henry, C., Reinforcement learning with approximation spaces. Fundamenta Informaticae 71(2-3) (2006) 323-349.
80. S. Plous, The Psychology of Judgement and Decision Making, McGraw-Hill, New York, 1993.
81. T. Poggio, S. Smale, The mathematics of learning: Dealing with data, Notices of the AMS 50(5) (2003) 537-544.

82. L. Polkowski, A. Skowron, Rough mereology: A new paradigm for approximate reasoning, International Journal of Approximate Reasoning 15 (1996) 333-365.
83. L. Polkowski, S. Tsumoto, T. Y. Lin (Eds.), Rough Set Methods and Applications: New Developments in Knowledge Discovery in Information Systems, Studies in Fuzziness and Soft Computing vol. 56, Physica-Verlag Heidelberg, 2000.
84. G. Pólya, How to Solve It, 2nd ed., Princeton University Press, 1957; see also http://en.wikipedia.org/wiki/How_to_Solve_It.
85. H. Rasiowa, W. Marek, On reaching consensus by groups of intelligent agents, In: Z. W. Ras (Ed.), Methodologies for Intelligent Systems, North-Holland, Amsterdam, 1989, 234-243.
86. H. Rasiowa, Algebraic Models of Logics, Warsaw University, 2001.
87. H. Rasiowa, R. Sikorski, The Mathematics of Metamathematics, Monografie Matematyczne vol. 41, PWN Warsaw, 1963.
88. S. Read, Thinking about Logic. An Introduction to the Philosophy of Logic, Oxford, New York: Oxford University Press, 1995.
89. P. Saint-Dizier, E. Viegas, B. Boguraev, S. Bird, D. Hindle, M. Kay, D. McDonald, H. Uszkoreit, Y. Wilks, Computational Lexical Semantics (Studies in Natural Language Processing), Cambridge University Press, 2005.
90. C. Schlenoff, J. Albus, E. Messina, A. J. Barbera, R. Madhavan, S. Balakirsky, Using 4D/RCS to address AI knowledge integration, AI Magazine 27(2) (2006) 71-81.
91. Scientific Datalink, The Scientific DataLink index to artificial intelligence research, 1954-1984, Scientific DataLink, 1985.
92. Scientific Datalink, The Scientific DataLink index to artificial intelligence research, 1985 Supplement, Scientific DataLink, 1985.
93. A. Skowron, Approximate reasoning in distributed environments, in N. Zhong, J. Liu (Eds.), Intelligent Technologies for Information Analysis, Heidelberg: Springer, pp. 433-474.
94. A. Skowron, Perception logic in intelligent systems (keynote talk), In: S. Blair et al (Eds.), Proceedings of the 8th Joint Conference on Information Sciences (JCIS 2005), July 21-26, 2005, Salt Lake City, Utah, USA, X-CD Technologies: A Conference & Management Company, ISBN 0-9707890-3-3, 15 Coldwater Road, Toronto, Ontario, M3B 1Y8, 2005, pp. 1-5.
95. A. Skowron, Rough sets in perception-based computing (keynote talk), in [71], pp. 21-29.
96. A. Skowron, R. Agrawal, M. Luck, T. Yamaguchi, O. Morizet-Mahoudeaux, J. Liu, N. Zhong (Eds.), Proceedings of the 2005 IEEE/WIC/ACM International Conference on WEB Intelligence, Compiegne, France, September 19-22, 2005, IEEE Computer Society Press, Los Alamitos, CA, 2005, pp. 1-819.
97. A. Skowron, J.-P. Barthes, L. Jain, R. Sun, P. Morizet-Mahoudeaux, J. Liu, N. Zhong (Eds.), Proceedings of the 2005 IEEE/WIC/ACM International Conference on Intelligent Agent Technology, Compiegne, France, September 19-22, 2005, IEEE Computer Society Press, Los Alamitos, CA, 2005, pp. 1-766.
98. A. Skowron, J. Stepaniuk, Tolerance approximation spaces, Fundamenta Informaticae 27 (1996) 245-253.
99. A. Skowron, J. Stepaniuk, Information granules: Towards foundations of granular computing, International Journal of Intelligent Systems 16(1) (2001) 57-86.
100. A. Skowron, J. Stepaniuk, Information granules and rough-neural computing, in [70], pp. 43-84.
101. A. Skowron, P. Synak, Complex patterns, Fundamenta Informaticae 60(1-4) (2004) 351-366.

102. A. Skowron, R. Swiniarski, and P. Synak, Approximation spaces and information granulation, Transactions on Rough Sets III: Journal Subline, Lecture Notes in Computer Science, vol. 3400, pp. 175-189, Heidelberg: Springer, 2005.
103. J. F. Sowa, Knowledge Representation: Logical, Philosophical, and Computational Foundations, Course Technology, 1999.
104. S. Staab, R. Studer, Handbook on Ontologies, in International Handbooks on Information Systems, Heidelberg: Springer 2004.
105. J. Stepaniuk, J. Bazan, and A. Skowron, Modelling complex patterns by information systems, Fundamenta Informaticae 67 (1-3) (2005) 203-217.
106. J. Stepaniuk, A. Skowron, J. Peters, and R. Swiniarski, Calculi of approximation spaces, Fundamenta Informaticae 72 (1-3) (2006) 363-378.
107. P. Stone, Layered Learning in Multi-Agent Systems: A Winning Approach to Robotic Soccer, Cambridge, MA: The MIT Press, 2000.
108. W. Swartout, J. Gratch, R. W. Hill, E. Hovy, S. Marsella, J. Rickel, D. Traum, Towards virtual humans, AI Magazine 27(2) (2006) 96-108.
109. R. Sun (Ed.), Cognition and Multi-Agent Interaction. From Cognitive Modeling to Social Simulation. New York, NY: Cambridge University Press, 2006.
110. K. Sycara, Multiagent systems, in AI Magazine, Summer 1998, 79-92.
111. D. Ślęzak, J. T. Yao, J. F. Peters, W. Ziarko, and X. Hu (Eds.), Proceedings of the 10th International Conference on Rough Sets, Fuzzy Sets, Data Mining, and Granular Computing (RSFDGrC'2005), Regina, Canada, August 31-September 3, 2005, Part II, Lecture Notes in Artificial Intelligence, vol. 3642, Heidelberg: Springer, 2005.
112. A. S. Troelstra, H. Schwichtenberg, Basic Proof Theory, Cambridge University Press, 2000.
113. A. S. Troelstra, D. Van Dalen, Constructivism in Mathematics: An Introduction, Studies in Logic and the Foundations of Mathematics Vol. 1 & 2, Elsevier Science Publishing Company, 1988.
114. A. Turing, Computing machinery and intelligence, Mind LIX(236) (October 1950) 433-460.
115. W. Van Wezel, R. Jorna, and A. Meystel, Planning in Intelligent Systems: Aspects, Motivations, and Methods. Hoboken, New Jersey: John Wiley & Sons, 2006.
116. V. Vapnik, Statistical Learning Theory, New York: John Wiley & Sons, 1998.
117. S. Weiss, N. Indurkhya, T. Zhang, F. Damerau, Text Mining: Predictive Methods for Analyzing Unstructured Information, Springer, 2004.
118. I. H. Witten, E. Frank, Data Mining: Practical Machine Learning Tools and Techniques, Second Edition, Morgan Kaufmann Series in Data Management Systems, Morgan Kaufmann, 2005.
119. I. H. Witten, A. Moffat, T. C. Bell, Managing Gigabytes: Compressing and Indexing Documents and Images, The Morgan Kaufmann Series in Multimedia and Information Systems, Morgan Kaufmann, 1999.
120. L. A. Zadeh, Fuzzy sets, Information and Control 8 (1965) 333-353.
121. L. A. Zadeh, From computing with numbers to computing with words - from manipulation of measurements to manipulation of perceptions, IEEE Transactions on Circuits and Systems - I: Fundamental Theory and Applications 45(1) (1999) 105-119.
122. L. A. Zadeh, A new direction in AI: Toward a computational theory of perceptions, AI Magazine 22 (1) (2001) 73-84.
123. W. Ziarko, Variable precision rough set model, Journal of Computer and System Sciences 46 (1993) 39-59.