Rough-Neuro Computing

Lech Polkowski[1], Andrzej Skowron[2]

[1] Polish–Japanese Institute of Information Technology
Koszykowa 86, 02-008 Warsaw, Poland
and
Department of Mathematics and Information Sciences
Warsaw University of Technology
Pl. Politechniki 1,00-650 Warsaw, Poland
e-mail: polkow@pjwstk.waw.pl

[2] Institute of Mathematics
Warsaw University
Banacha 2, 02-097 Warsaw, Poland
e-mail: skowron@mimuw.edu.pl

Abstract. We outline a rough–neuro computing model as a basis for granular computing. Our approach is based on rough sets, rough mereology and information granule calculus.
Keywords: rough sets, neural networks, granular computing, rough mereology

1 Introduction

Rough Mereology [4], [7] is a paradigm allowing for a synthesis of main ideas of two paradigms for reasoning under uncertainty: Fuzzy Set Theory and Rough Set Theory. We present applications of Rough Mereology to the important theoretical idea put forth by Lotfi Zadeh [11], [12], i.e., Granularity of Knowledge by presenting the idea of rough–neuro computing paradigm.

We emphasize an important property of granular computing related to the necessity of lossless compression tuning for complex object constructions. It means that we map a cluster of constructions into one representation. Any construction in the cluster is delivering objects satisfying the specification in satisfactory degree if only objects input to synthesis are sufficiently close to selected standards (prototypes). In rough mereological approach clusters of constructions are represented by the so–called stable schemes (of co–operating agents), i.e., schemes robust to some deviations of parameters of transformed granules. In consequence, the stable schemes are able to return objects satisfying in satisfactory degree the specification not only from standard (prototype) objects but also from objects sufficiently close to them [4], [5]. In this way any stable scheme of complex object construction is a representation of a cluster of similar constructions from clusters of elementary objects.

We extend schemes for synthesis of complex objects (or granules) developed in [7] and [5] by adding one important component. As a result we receive the granule construction schemes which can be treated as a generalization of neural network models. The main idea is that granules sent by one agent to another are not, in general, exactly understandable by the receiving agent because these agents are using different languages and usually there is no meaning–preserving translation of formulas of the language of the sending agent to formulas of the receiving agent. Hence, it is necessary to construct some interfaces which will allow to approximately understand received granules. These interfaces can be, in the simplest case, constructed on the basis of exchanged information about agents stored in the form of decision data tables. From these tables the approximations of concepts can be constructed using rough set approach [10]. In our model we assume that for any agent ag and its operation $o(ag)$ of arity n there are approximation spaces $AS_1(o(ag), in), ..., AS_n(o(ag), in)$ which will filter (approximately) the granules received by the agent for performing the operation $o(ag)$. In turn, the granule sent by the agent after performing the operation is filtered (approximated) by the approximation space $AS(o(ag), out)$. These approximation spaces are parameterized with parameters allowing to optimize the size of neighborhoods in these spaces as well as the inclusion relation [8] using as a criterion for optimization the quality of granule approximation. Approximation spaces attached to an operation correspond to neuron weights in neural networks whereas the operation performed by the agent corresponds to the operation realized on the vector of real numbers by the neuron. The generalized scheme of agents is returning a granule in response to input information granules. It can be for example a cluster of elementary granules. Hence our schemes realize much more general computations then neural networks operating on vectors of real numbers. The question, if such schemes can be efficiently simulated by classical neural networks is open.

We would like to call extended schemes for complex object construction *rough-neuro schemes* (for complex object construction). The stability of such schemes corresponds to the resistance to noise of classical neural networks.

In the paper we present in some details the outlined above rough– neuro computing paradigm.

2 Adaptive Calculus of Granules in Distributed Systems

We now present a conceptual scheme for adaptive calculus of granules aimed at synthesizing solutions to problems posed under uncertainty. This exposition is based on our earlier analyzes presented in [4], [7]. We construct a scheme of agents which communicate by relating their respective granules of knowledge by means of transfer functions induced by rough mereological connectives extracted from their respective information systems. We assume the notation of [7] where the reader will find all the necessary information.

We now define formally the ingredients of our scheme of agents.

2.1 Distributed Systems of Agents

We assume that a pair (Inv, Ag) is given where Inv is an *inventory of elementary objects* and Ag is a set of inteligent computing units called shortly *agents*.

We consider an agent $ag \in Ag$. The agent ag is endowed with tools for reasoning about objects in its scope; these tools are defined by components of the agent label. The *label of the agent ag* is the tuple

$$lab(ag) = (\mathcal{A}(ag), M(ag), L(ag), Link(ag), AP\_O(ag), St(ag),\\ Unc\_rel(ag), H(ag), Unc\_rule(ag), Dec\_rule(ag))$$

where

1. $\mathcal{A}(ag) = (U(ag), A(ag))$ is an information system of the agent ag; we assume as an example that objects (i.e., elements of $U(ag)$) are granules of the form: $(\alpha, [\alpha])$ where α is a conjunction of descriptors (one may have more complex granules as objects).

2. $M(ag) = (U(ag), [0,1], \mu_o(ag))$ is a pre - model of L_{rm} with a pre - rough inclusion $\mu_o(ag)$ on the universe $U(ag)$;

3. $L(ag)$ is a set of unary predicates (properties of objects) in a predicate calculus interpreted in the set $U(ag)$; we may assume that formulae of $L(ag)$ are constructed as conditional formulae of logics L_B where $B \subseteq A(ag)$.

4. $St(ag) = \{st(ag)_1, ..., st(ag)_n\} \subset U(ag)$ is the set of *standard objects* at ag;

5. $Link(ag)$ is a collection of strings of the form $t = ag_1 ag_2 ... ag_k ag$; the intended meaning of a string $ag_1 ag_2 ... ag_k ag$ is that $ag_1, ag_2, .., ag_k$ are children of ag in the sense that ag can assemble complex objects (constructs) from simpler objects sent by $ag_1, ag_2, ..., ag_k$. In general, we may assume that for some agents ag we may have more than one element in $Link(ag)$ which represents the possibility of re - negotiating the synthesis scheme.

We denote by the symbol $Link$ the union of the family $\{Link(ag) : ag \in Ag\}$.

6. $AP\_O(ag)$ consists of pairs of the form:

$$(o(ag,t), ((AS_1(o(ag), in), ..., AS_n(o(ag), in)), AS(o(ag), out))$$

where $o(ag,t) \in O(ag)$, n is the arity of $o(ag,t)$, $t = ag_1 ag_2 ... ag_k ag \in Link$, $AS_i(o(ag,t), in)$ is a parameterized approximation space [10] corresponding to the $i-th$ argument of $o(ag,t)$ and $AS(o(ag,t), out)$ is a parameterized approximation space [10] for the output of $o(ag,t)$.

$O(ag)$ is the set of operations at ag; any $o(ag,t) \in O(ag)$ is a mapping of the Cartesian product $U(ag) \times U(ag) \times ... \times U(ag)$ into the universe $U(ag)$; the meaning of $o(ag,t)$ is that of an operation by means of which the agent ag is able to assemble from objects $x_1 \in U(ag_1), x_2 \in U(ag_2), ..., x_k \in U(ag_k)$ the object $z \in U(ag)$ which is an approximation defined by $AS(o(ag,t), out)$ to $o(ag,t)(y_1, y_2, ..., y_k) \in U(ag)$ where y_i is the approximation to x_i defined by $AS_i(o(ag,t), in)$. One may choose here either a lower or an upper approximation.

7. $Unc\_rel(ag)$ is the set of uncertainty relations unc_rel_i of type

$$(o_i(ag,t), \rho_i(ag), ag_1, ..., ag_k, ag, \mu_o(ag_1), ..., \mu_o(ag_k), \mu_o(ag),\\ st(ag_1)_i, ..., st(ag_k)_i, st(ag)_i)$$

where $ag_1 ag_2 ... ag_k ag \in Link(ag)$, $o_i(ag,t) \in O(ag)$ and ρ_i is such that

$$\rho_i((x_1, \varepsilon_1), (x_2, \varepsilon_2), ., (x_k, \varepsilon_k), (x, \varepsilon))$$

holds for $x_1 \in U(ag_1), x_2 \in U(ag_2), .., x_k \in U(ag_k)$ and $\varepsilon, \varepsilon_1, \varepsilon_2, .., \varepsilon_k \in [0,1]$ iff $\mu_o(x_j, st(ag_j)_i) = \varepsilon_j$ for $j = 1, 2, .., k$ and $\mu_o(x, st(ag)_i) = \varepsilon$ for the collection of standards $st(ag_1)_i, st(ag_2)_i, ..\ .,\ st(ag_k)_i, st(ag)_i$ such that

$$o_i(ag,t)(st(ag_1)_i, st(ag_2)_i, .., st(ag_k)_i) = st(ag)_i.$$

The operation o performed by ag here is more complex then that of [7] as it is composed of three stages: first, approximations to input objects are constructed, next the operation is performed, and finally the approximation to the result is constructed. Relations unc_rel_i provide a global description of this process; in reality, they are composition of analogous relations corresponding to the three stages. As a result, unc_rel_i depend on parameters of approximation spaces. This concerns also other constructs discussed here. It follows that in order to get satisfactory decomposition (similarly, uncertainty and so on) rules one has to search for satisfactory parameters of approximation spaces (this is in analogy to weight tuning in neural computations).

Uncertainty relations express the agents knowledge about relationships among uncertainty coefficients of the agent ag and uncertainty coefficients of its children. The relational character of these dependencies expresses their intensionality.

8. $Unc\_rule(ag)$ is the set of uncertainty rules unc_rule_j of type
$(o_j(ag,t), f_j, ag_1,\ ag_2, .\ .., ag_k,\ ag, st(ag_1), st(ag_2), ..., st(ag_k), st(ag),$
$\mu_o(ag_1), ...\ , \mu_o(ag_k), \mu_o(ag))$
of the agent ag where $ag_1 ag_2 ... ag_k ag \in Link(ag)$ and $f_j : [0,1]^k \longrightarrow [0,1]$ is a function which has the property that
if $o_j(ag,t)(st(ag_1),\ st(ag_2),\ ..., st(ag_k)) = st(ag)$ **and**
$x_1 \in U(ag_1), x_2 \in U(ag_2), ..., x_k \in U(ag_k)$
satisfy the conditions $\mu_o(x_i, st(ag_i)) \geq \varepsilon(ag_i)$ for $i = 1, 2, .., k$

then $\mu_o(o_j(ag,t)(x_1, x_2, ..., x_k), st(ag)) \geq f_j(\varepsilon(ag_1), \varepsilon(ag_2), .., \varepsilon(ag_k))$.

Uncertainty rules provide functional operators (approximate mereological connectives) for propagating uncertainty measure values from the children of an agent to the agent; their application is in negotiation processes where they inform agents about plausible uncertainty bounds.

9. $H(ag)$ is a strategy which produces uncertainty rules from uncertainty relations; to this end, various rigorous formulas as well as various heuristics can be applied among them the algorithm presented in Section 2.8 of [7].

10. $Dec\_rule(ag)$ is a set of decomposition rules dec_rule_i of type

$$(o_i(ag,t), ag_1, ag_2, ..., ag_k, ag)$$

such that $(\Phi(ag_1), \Phi(ag_2), .., \Phi(ag_k), \Phi(ag)) \in dec\_rule_i$ (where $\Phi(ag_1) \in L(ag_1)$, $\Phi(ag_2) \in L(ag_2)$, ..., $\Phi(ag_k) \in L(ag_k)$, $\Phi(ag) \in L(ag)$ and $ag_1 ag_2 ... ag_k ag \in Link(ag)$) and there exists a collection of standards $st(ag_1)$, $st(ag_2)$,..., $st(ag_k)$, $st(ag)$ with the properties that

$$o_j(ag,t)(st(ag_1),st(ag_2),..,st(ag_k))=st(ag),$$

$st(ag_i)$ satisfies $\Phi(ag_i)$ for $i=1,2,..,k$ and $st(ag)$ satisfies $\Phi(ag)$.

Decomposition rules are decomposition schemes in the sense that they describe the standard $st(ag)$ and the standards $st(ag_1),...,st(ag_k)$ from which the standard $st(ag)$ is assembled under o_i in terms of predicates which these standards satisfy.

We may sum up the content of 1 - 10 above by saying that for any agent ag the possible sets of children of this agent are specified and, relative to each team of children, decompositions of standard objects at ag into sets of standard objects at the children, uncertainty relations as well as uncertainty rules, which relate similarity degrees of objects at the children to their respective standards and similarity degree of the object built by ag to the corresponding standard object at ag, are given.

We take rough inclusions of agents as measures of uncertainty in their respective universes. We would like to observe that the mereological relation of being a part is not transitive globally over the whole synthesis scheme because distinct agents use distinct mereological languages.

2.2 Approximate Synthesis of Complex Objects

The process of synthesis of a complex object (signal, action) by the above defined scheme of agents consists in our approach of the two communication stages viz. the top - down communication/negotiation process and the bottom - up communication/assembling process. We outline the two stages here in the language of approximate formulae.

Approximate logic of synthesis For simplicity of exposition and to avoid unnecessarily tedious notation, we assume that the relation $ag' \leq ag$, which holds for agents $ag', ag \in Ag$ iff there exists a string $ag_1 ag_2 ... ag_k ag \in Link(ag)$ with $ag' = ag_i$ for some $i \leq k$, orders the set Ag into a tree. We also assume that $O(ag) = \{o(ag,t)\}$ for $ag \in Ag$ i.e. each agent has a unique assembling operation for a unique t.

The process of synthesis of a complex object (signal, action) by the above defined scheme of agents consists in our approach of the two communication stages viz. the top - down communication/negotiation process and the bottom - up communication process. We outline the two stages here in the language of approximate formulae. To this end we build a logic $L(Ag)$ (cf. [7]) in which we can express global properties of the synthesis process. We recall our assumption that the set Ag is ordered into a tree by the relation $ag' \leq ag$.

Elementary formulae of $L(Ag)$ are of the form $\langle st(ag), \Phi(ag), \varepsilon(ag)\rangle$ where $st(ag) \in St(ag), \Phi(ag) \in L(ag), \varepsilon(ag) \in [0,1]$ for any $ag \in Ag$. Formulae of $L(ag)$ form the smallest extension of the set of elementary formulae closed under propositional connectives $\vee$, $\wedge$, $\neg$ and under the modal operators [], <> .

To introduce a semantics for the logic $L(ag)$, we first specify the meaning of satisfaction for elementary formulae. The meaning of a formula $\Phi(ag)$ is defined classically as the set $[\Phi(ag)] = \{u \in U(ag) : u$ has the property $\Phi(ag)\}$; we will denote the fact that $u \in [\Phi(ag)]$ by the symbol $u \models \Phi(ag)$. We extend now the satisfiability predicate $\models$ to approximate formulae: for $x \in U(ag)$, we say that x *satifies* an elementary formula $\langle st(ag), \Phi(ag), \varepsilon(ag)\rangle$, in symbols: $x \models < st(ag), \Phi(ag), \varepsilon(ag) >$, iff (i) $st(ag) \models \Phi(ag)$ and (ii) $\mu_o(ag)(x, st(ag)) \geq \varepsilon(ag)$.

We let

(iii) $x \models \neg\langle st(ag), \Phi(ag), \varepsilon(ag)\rangle$ iff it is not true that $x \models \langle st(ag), \Phi(ag), \varepsilon(ag)\rangle$;
(iv) $x \models \langle st(ag)_1, \Phi(ag)_1, \varepsilon(ag)_1\rangle \vee \langle st(ag)_2, \Phi(ag)_2, \varepsilon(ag)_2\rangle$ iff
$$x \models \langle st(ag)_1, \Phi(ag)_1, \varepsilon(ag)_1\rangle \text{ or } x \models \langle st(ag)_2, \Phi(ag)_2, \varepsilon(ag)_2\rangle.$$

In order to extend the semantics over modalities, we first introduce the notion of a selection: by a *selection* over Ag we mean a function sel which assigns to each agent ag an object $sel(ag) \in U(ag)$.

For two selections sel, sel' we say that *sel induces sel'*, in symbols $sel \to_{Ag} sel'$ when $sel(ag) = sel'(ag)$ for any $ag \in Leaf(Ag)$ and
$$sel'(ag) = o(ag,t)(sel'(ag_1), sel'(ag_2), ..., sel'(ag_k))$$
for any $ag_1 ag_2 ... ag_k ag \in Link$.

We extend the satisfiability predicate $\models$ to selections: for an elementary formula $\langle st(ag),\ \Phi(ag),\ \varepsilon(ag)\rangle$, we let $sel \models \langle st(ag), \Phi(ag), \varepsilon(ag)\rangle$ iff $sel(ag) \models \langle st(ag), \Phi(ag), \varepsilon(ag)\rangle$.

We now let $sel \models <>< st(ag), \Phi(ag), \varepsilon(ag) >$ when there exists a selection sel' satisfying the conditions:

(i) $sel \to_{Ag} sel'$; (ii) $sel' \models \langle st(ag), \Phi(ag), \varepsilon(ag)\rangle$.

In terms of logic $L(Ag)$ it is posible to express the problem of synthesis of an approximate solution to the problem posed to the team Ag. We denote by $head(Ag)$ the root of the tree $(Ag, \leq)$.

In the process of top - down communication, a requirement Ψ received by the scheme from an external source (which may be called a *customer*) is decomposed into approximate specifications of the form $\langle st(ag), \Phi(ag), \varepsilon(ag)\rangle$ for any agent ag of the scheme. The decomposition process is initiated at the agent $head(Ag)$ and propagated down the tree.

We are able now to formulate the synthesis problem.

Synthesis problem

Given a formula $\alpha : \langle st(head(Ag)), \Phi(head(Ag)), \varepsilon(head(Ag))\rangle$ *find a selection sel over the tree* $(Ag, \leq)$ *with the property sel* $\models <> \alpha$.

A solution to the synthesis problem with a given formula

$$\langle st(head(Ag)), \Phi(head(Ag)),\ \varepsilon(head(Ag))\rangle$$

is found by negotiations among the agents; negotiations are based on uncertainty rules of agents and their succesful result can be expressed by a top-down recursion in the tree $(Ag, \leq)$ as follows: given a local team $ag_1 ag_2 ... ag_k ag$ with the formula $\langle st(ag), \Phi(ag), \varepsilon(ag)\rangle$ already chosen in negotiations on a higher tree level, it is sufficient that each agent ag_i choose a standard $st(ag_i) \in U(ag_i)$, a formula $\Phi(ag_i) \in L(ag_i)$ and a coefficient $\varepsilon(ag_i) \in [0,1]$ such that

(v) $(\Phi(ag_1), \Phi(ag_2), .., \Phi(ag_k), \Phi(ag)) \in Dec\_rule(ag)$ with standards $st(ag)$, $st(ag_1), . .., st(ag_k)$;

(vi) $f(\varepsilon(ag_1), .., \varepsilon(ag_k)) \geq \varepsilon(ag)$ where f satisfies $unc\_rule(ag)$ with $st(ag)$, $st(ag_1)$, ..., $st(ag_k)$ and $\varepsilon(ag_1)$, .., $\varepsilon(ag_k)$, $\varepsilon(ag)$.

For a formula α : $\langle st(head(Ag)), \Phi(head(Ag)),\ \varepsilon(head(Ag))\rangle$, we call an α - *scheme* an assignment of a formula $\alpha(ag)$: $\langle st(ag), \Phi(ag), \varepsilon(ag)\rangle$ to each $ag \in Ag$ in such manner that (v), (vi) above are satisfied and $\alpha(head(Ag))$ is $\langle st(head(Ag)), \Phi(head(Ag)),\ \varepsilon(head(Ag))\rangle$; we denote this scheme with the symbol

$$sch(\langle st(head(Ag)), \Phi(head(Ag)),\ \varepsilon(head(Ag))\rangle).$$

We say that a selection *sel* is *compatible* with a scheme

$$sch(\langle st(head(Ag)), \Phi(head(Ag)),\ \varepsilon(head(Ag))\rangle)$$

when $\mu_o(ag,t)(sel(ag), st(ag)) \geq \varepsilon(ag)$ for each leaf agent $ag \in Ag$ where $\langle st(ag), \Phi(ag), \varepsilon(ag)\rangle$ is the value of the scheme at ag for any leaf $ag \in Ag$.

Any leaf agent realizes its approximate specification by choosing in the subset $Inv \cap U(ag)$ of the inventory of primitive constructs a construct satisfying the specification.

The goal of negotiations can be summarized now as follows.

Proposition 3.1

For a given a requirement $\langle st(head(Ag)), \Phi(head(Ag)), \varepsilon(head(Ag))\rangle$ we have:

if a selection *sel* is compatible with a scheme

$$sch(\langle st(head(Ag)), \Phi(head(Ag)),\ \varepsilon(head(Ag))\rangle)$$

then $sel \models <> \langle st(head(Ag)), \Phi(head(Ag)),\ \varepsilon(head(Ag))\rangle$.

The bottom-up communication consists of agents sending to their parents the chosen constructs. The root agent $root(Ag)$ assembles the final construct.

3 Conclusions

We have outlined a general scheme for rough neuro–computation based on ideas of knowledge granulation by rough mereological tools. An important practical problem is a construction of such schemes (networks) for rough–neuro computing and of algorithms for parameter tuning. We now foresee two possible approaches: the one in which we would rely on new, original decomposition, synthesis and tuning methods in analogy to [7] but in the presence of approximation spaces; the second, in which a rough–neuro computing scheme would be encoded by a neural network in such a way that optimalization of weights in the neural net leads to satisfactory solutions for the rough–neuro computing scheme (cf. [3] for a first attempt in this direction).

Acknowledgments: The research of Lech Polkowski has been supported by the grants: No 8T11C 024 17 from the State Committee for Scientific Research (KBN) of the Republic of Poland and the European ESPRIT Program in CRIT 2 Research Project No 20288. The research of Andrzej Skowron has been supported by the grants of National Committee for Scientific Research, European ESPRIT Program in CRIT 2 Research Project No 20288, and the Wallenberg Foundation.

References

1. Lin, T.Y. Granular Computing on Binary Relations I. Data Mining and Neighborhood Systems (1998), In: [6] **1** pp. 107–121.
2. Z. Pawlak (1991), *Rough Sets – Theoretical Aspects of Reasoning about Data*, Kluwer Academic Publishers, Dordrecht.
3. L. Han, J.F. Peters, S. Ramanna, and R. Zhai (1999), Classifying faults in high voltage power systems: a rough–fuzzy neural computational approach, *Proceedings RSFDGrC'99: 7th International Workshop on Rough Sets, Fuzzy Sets, Data Mining and Granular Soft Computing*, Lecture Notes in Artificial Intelligence **1711**, Springer Verlag, Berlin, pp. 47–54.
4. L. Polkowski, A. Skowron (1996), Rough mereology: A new paradigm for approximate reasoning, *International Journal of Approximate Reasoning* **15/4**, pp. 333–365.
5. L. Polkowski, A. Skowron (1998), Rough mereological foundations for design, analysis, synthesis, and control in distributed systems, *Information Sciences An International Journal* **104/1-2**, Elsevier Science, New York, pp. 129–156.
6. L. Polkowski, A. Skowron (Eds.) (1998), *Rough Sets in Knowledge Discovery* **1-2** Physica-Verlag, Heidelberg 1998.
7. L. Polkowski, A. Skowron (1999), Towards adaptive calculus of granules, In: [13], **1**, pp. 201-227.
8. L. Polkowski, A. Skowron (2000). "Rough Mereology in Information Systems. A Case Study: Qualitative Spatial Reasoning", In: L. Polkowski, T.Y. Lin, S. Tsumoto (Eds.), *Rough sets: New developments*, Studies in Fuzziness and Soft Computing, Physica-Verlag / Springer-Verlag, Heidelberg, 2000 (in print).
9. Skowron, A.; Stepaniuk, J. (1999), Towards Discovery of Information Granules, 3rd European Conference of Principles and Practice of Knowledge Discovery in Databases, September 15–18, 1999, Prague, Czech Republic, Lecture Notes in Artificial Intelligence **1704**, Springer-Verlag, Berlin , pp. 542–547.
10. A. Skowron, J. Stepaniuk, S. Tsumoto (1999), Information Granules for Spatial Reasoning, *Bulletin of the International Rough Set Society* **3/4**, pp. 147-154.
11. L.A. Zadeh (1996), Fuzzy logic = computing with words, *IEEE Trans. on Fuzzy Systems* **4**, pp. 103-111.
12. L.A. Zadeh (1997), Toward a theory of fuzzy information granulation and its certainty in human reasoning and fuzzy logic, *Fuzzy Sets and Systems* **90**, pp. 111-127.
13. Zadeh, L.A., Kacprzyk, J. (eds.): *Computing with Words in Information/Intelligent Systems* vol.1-2, Physica-Verlag, Heidelberg, 1999.