Fundamenta Informaticae 119 (2012) 1–17
DOI 10.3233/FI-2012-724
IOS Press

Rough Set Based Reasoning About Changes

Andrzej Skowron[*]
The University of Warsaw
Banacha 2, 02-097 Warsaw, Poland
skowron@mimuw.edu.pl

Jarosław Stepaniuk[†]
Białystok University of Technology
Wiejska 45A, 15-351 Białystok, Poland
j.stepaniuk@pb.edu.pl

Andrzej Jankowski[‡]
Warsaw University of Technology
Nowowiejska 15/19, 00-665 Warsaw, Poland
a.jankowski@ii.pw.edu.pl

Jan G. Bazan[§]
University of Rzeszów
Dekerta 2, 35 - 030 Rzeszów, Poland
bazan@univ.rzeszow.pl

Roman Swiniarski[¶]
San Diego State University
5500 Campanile Drive San Diego, CA 92182, USA
rswiniar@sciences.sdsu.edu

Sir Isaac Newton said the statement:
"I can calculate the movement of the stars, but not the madness of men"
after loss of around 3 million of pounds in present day terms in the South Sea Bubble in 1720
(`http://en.wikipedia.org/wiki/South_Sea_Company`).
If he had not been traumatized by this loss, Sir Isaac might well have gone on to discover
the Fourth Law of Motion: For investors as a whole, returns decrease as motion increases.
– Warren Buffett

`http://investing-school.com/history/52-must-read-quotes-from-legendary-investor-warren-buffett/`

Address for correspondence: Institute of Mathematics, The University of Warsaw, Banacha 2, 02-097 Warszawa, Poland
[*]The research by A. Skowron, J.G. Bazan and A. Jankowski has been partially supported by the grant NN516 077837 from the Ministry of Science and Higher Education of the Republic of Poland, the grant realized within Homing Plus programme, edition 3/2011, of Foundation for Polish Science, co-financed from European Union, Regional Development Fund and by the National Centre for Research and Development (NCBiR) under grant SP/I/1/77065/10 by the Strategic scientific research and experimental development program: "Interdisciplinary System for Interactive Scientific and Scientific-Technical Information".
[†]The research by Jarosław Stepaniuk is supported by the Rector's grant S/WI/5/08 of Białystok University of Technology.
[‡]Also works: Institute of Mathematics, The University of Warsaw, Banacha 2, 02-097 Warszawa, Poland
[§]Also works: Institute of Mathematics, The University of Warsaw, Banacha 2, 02-097 Warszawa, Poland
[¶]Also works: Institute of Computer Science Polish Academy of Sciences, Ordona 21, 01-237 Warsaw, Poland

Abstract. We consider several issues related to reasoning about changes in systems interacting with the environment by sensors. In particular, we discuss challenging problems of reasoning about changes in hierarchical modeling and approximation of transition functions or trajectories. This paper can also be treated as a step toward developing rough calculus.

Keywords: rough sets, reasoning about changes, hierarchical modeling, granular computing, approximation space, relation (function) approximation, rough calculus, intelligent systems, computational finance, forex, algorithmic trading

1. Introduction

Reasoning about changes is one of the challenging issues in AI since the beginning of AI (see, e.g., [12, 13, 10, 32, 31, 3, 30, 11, 14, 43, 2, 4]).

This paper can be also treated as a first step toward the rough calculus development. There are several papers by Zdzisław Pawlak related to rough calculus which were published soon after discovering by him rough sets (see, e.g., [24, 25]). The approach presented in this paper is based on a different approach to function approximation. Also, our approach to reasoning about function changes is different. In particular, in many applications we will be not able to derive analytical form of approximations of function changes but we need to use these approximations in the form represented by induced from data classifiers (or predictors) for approximate reasoning, *e.g.*, for approximate reasoning about trajectories. This point of view is also different from the approach based on fuzzy sets presented in [1]. In the paper, we propose to apply the rough granular approach [27, 39, 17] for formalization of rough derivatives and for approximate reasoning about them.

In this paper, we consider a bottom up approach. We start from sensory information systems (sensory data tables) in which are recorded sensory measurements in different moments of time. Next, by using hierarchical modeling are constructed new information systems with more compound structural granules (sets of objects) such as time windows or sequences of time windows. On different levels of hierarchical modeling one can consider relations of changes, e.g., between successive time windows. Note that information systems represent only partial information about the universe of possible objects (i.e., some samples of possible objects) and the relations of changes should be induced (approximated) from partial information about the relation. We propose to use Boolean reasoning in searching for models of approximated relations. In particular, the proposed approach can be used for approximation of transition relations. Moreover, we illustrate how the approach can be extended for inducing approximation of trajectories defined by transition relations. One can also consider approximation of changes of functions relative to changes of granules representing their arguments by using the rough-set based methods (see, e.g., [33, 35]). The discussed approach to function approximation may be treated as complementary to the approach developed in functional data analysis (see, e.g., [29]).

This article is more intended to pose some questions and provide suggestions in which direction we may search for answers, rather than deliver ready to use technical solutions. It is an extension of an extended abstract from the last year CS&P [34] and a continuation of [16, 17].

The paper is organized as follows. In Section 2 we discuss illustrative examples of our approach to approximation of changes. In Section 3 trajectory approximation is investigated. In Section 4 we present our work on developing of rough calculus. In particular, we present a real-life example illustrating the need for developing foundations for rough calculus and some challenges. In Conclusions, we summarize the results of the paper.

2. Approximation of changes in hierarchical modeling

In this section, we start from an illustrative example of our approach to approximation of function changes. The approach is based on Boolean reasoning [15]. Next, we add some comments on approximation of changes in hierarchical modeling. We assume that the reader is familiar with the basic notation concerning rough sets (see, e.g., [26, 39, 15]).

Example 2.1. We consider a set $U = \{x_1, \ldots, x_{12}\}$ of twelve plums observed in two time moments t_1 and t_2, where $t_1 < t_2$. We also consider a function f assigning to every object from U value "1" if and only if this object is ripe and "0" otherwise. An attribute a means hardness of objects with three possible values $l - low$, $m - middle$ and $h - high$. An attribute b is a color of plum with three possible values $g - green$, $y - yellow$ and $v - violet$. An attribute c is a size of plum with three possible values $s - small$, $m - middle$ and $l - large$.

More formally, we consider three data tables $(U^{t_i}, A^{t_i} \cup \{f^{t_i}\})$, where $i = 1, 2$ and $(\Delta U, \Delta A \cup \{\Delta f\})$ such that $U^{t_i} = \{x_1^{t_i}, \ldots, x_{12}^{t_i}\}$, $A^{t_i} = \{a^{t_i}, b^{t_i}, c^{t_i}\}$, $V_{a^{t_i}} = \{l, m, h\}$, $V_{b^{t_i}} = \{v, g, y\}$, $V_{c^{t_i}} = \{s, m, l\}$ and $V_{f^{t_i}} = \{1, 0\}$, $\Delta U = \{\Delta x_1, \ldots, \Delta x_{12}\}$, $V_{\Delta a} = \{v \to v' : \ v, v' \in V_a\}$, $V_{\Delta b} = \{v \to v' : \ v, v' \in V_b\}$, $V_{\Delta c} = \{v \to v' : \ v, v' \in V_c\}$, $V_{\Delta f} = \{v \to v' : \ v, v' \in V_f\}$. We define $\Delta a(x) = a^{t_1}(x) \to a^{t_2}(x)$, $\Delta b(x) = b^{t_1}(x) \to b^{t_2}(x)$, $\Delta c(x) = c^{t_1}(x) \to c^{t_2}(x)$ and $\Delta f(x) = f^{t_1}(x) \to f^{t_2}(x)$, where $x \in \Delta U$ (see Table 1).

Let us first compute the approximations with respect to values "1" and "0" of function f in time moment t_2. We also present the roughness coefficient. For simplicity of notation we omit the subscript t_2.

We define $X_1 = \{x \in U : f(x) = 1\} = \{x_1, x_2, x_3, x_5, x_7, x_9, x_{10}\}$ and $X_0 = \{x \in U : f(x) = 0\} = \{x_4, x_6, x_8, x_{11}, x_{12}\}$. Let $AS_{\{a,b,c\}} = (U, IND(\{a, b, c\})$ be an approximation space and $U/IND(\{a, b, c\})$ a partition of U defined by attributes from $\{a, b, c\}$.

We obtain the lower approximation

$$LOW(AS_{\{a,b,c\}}, X_1) = \{x_1, x_2, x_7, x_{10}\},$$

the upper approximation

$$UPP(AS_{\{a,b,c\}}, X_1) = \{x_1, x_2, x_7, x_{10}, x_3, x_4, x_5, x_8, x_9, x_{12}\},$$

and the roughness of X_1

$$R(AS_{\{a,b,c\}}, X_1) = 1 - \frac{card(LOW(AS_{\{a,b,c\}}, X_1))}{card(UPP(AS_{\{a,b,c\}}, X_1))} = 1 - 4 : 10 = 0.6.$$

Analogously, we obtain the roughness of X_0

$$R(AS_{\{a,b,c\}}, X_0) = 1 - 2 : 8 = 0.75.$$

Table 1. Three data tables: tables in time moments t_1 and t_2 and table of changes Δf

U^{t_1}	a^{t_1}	b^{t_1}	c^{t_1}	f^{t_1}
$x_1^{t_1}$	l	v	l	1
$x_2^{t_1}$	l	y	l	1
$x_3^{t_1}$	m	g	m	0
$x_4^{t_1}$	m	g	m	0
$x_5^{t_1}$	m	y	m	1
$x_6^{t_1}$	m	g	m	0
$x_7^{t_1}$	m	v	m	1
$x_8^{t_1}$	h	g	s	0
$x_9^{t_1}$	h	g	s	0
$x_{10}^{t_1}$	h	v	s	0
$x_{11}^{t_1}$	h	g	s	0
$x_{12}^{t_1}$	h	y	s	0

U^{t_2}	a^{t_2}	b^{t_2}	c^{t_2}	f^{t_2}
$x_1^{t_2}$	l	v	l	1
$x_2^{t_2}$	l	y	l	1
$x_3^{t_2}$	l	g	l	1
$x_4^{t_2}$	l	g	l	0
$x_5^{t_2}$	m	y	m	1
$x_6^{t_2}$	m	g	m	0
$x_7^{t_2}$	m	v	m	1
$x_8^{t_2}$	m	y	m	0
$x_9^{t_2}$	h	y	s	1
$x_{10}^{t_2}$	h	v	s	1
$x_{11}^{t_2}$	h	g	s	0
$x_{12}^{t_2}$	h	y	s	0

ΔU	Δa	Δb	Δc	Δf
Δx_1	$l \to l$	$v \to v$	$l \to l$	$1 \to 1$
Δx_2	$l \to l$	$y \to y$	$l \to l$	$1 \to 1$
Δx_3	$m \to l$	$g \to g$	$m \to l$	$0 \to 1$
Δx_4	$m \to l$	$g \to g$	$m \to l$	$0 \to 0$
Δx_5	$m \to m$	$y \to y$	$m \to m$	$1 \to 1$
Δx_6	$m \to m$	$g \to g$	$m \to m$	$0 \to 0$
Δx_7	$m \to m$	$v \to v$	$m \to m$	$1 \to 1$
Δx_8	$h \to m$	$g \to y$	$s \to m$	$0 \to 0$
Δx_9	$h \to h$	$g \to y$	$s \to s$	$0 \to 1$
Δx_{10}	$h \to h$	$v \to v$	$s \to s$	$0 \to 1$
Δx_{11}	$h \to h$	$g \to g$	$s \to s$	$0 \to 0$
Δx_{12}	$h \to h$	$y \to y$	$s \to s$	$0 \to 0$

Now let us consider the partition $\Delta U/IND(AS_{\{\Delta a,\Delta b,\Delta c\}})$ of ΔU equal to

$$\{\{\Delta x_1\},\{\Delta x_2\},\{\Delta x_3,\Delta x_4\},\{\Delta x_5\},\{\Delta x_6\},\{\Delta x_7\},\{\Delta x_8\},\{\Delta x_9\},\{\Delta x_{10}\},\{\Delta x_{11}\},\{\Delta x_{12}\}\}.$$

We compute the approximations of the change region, i.e., of the set

$$Change = \{\Delta x \in \Delta U : \Delta f(x) = 0 \to 1\} = \{\Delta x_3, \Delta x_9, \Delta x_{10}\} :$$

$$LOW(AS_{\{\Delta a,\Delta b,\Delta c\}}, Change) = \{\Delta x_9, \Delta x_{10}\},$$

$$UPP(AS_{\{\Delta a,\Delta b,\Delta c\}}, Change) = \{\Delta x_3, \Delta x_4, \Delta x_9, \Delta x_{10}\}.$$

Using Boolean reasoning, we obtain two decision reducts: $\{\Delta a, \Delta b\}$ and $\{\Delta b, \Delta c\}$.
Based on the first reduct we obtain the following two types of decision rules.
Rules with accuracy equal to 1 (based on the lower approximation):
if $\Delta a = h \to h$ **and** $\Delta b = g \to y$ **then** $\Delta f = 0 \to 1$ (based on object Δx_9),
if $\Delta a = h \to h$ **and** $\Delta b = v \to v$ **then** $\Delta f = 0 \to 1$ (based on object Δx_{10}),
Rule with accuracy less than 1 (based on boundary region):
if $\Delta a = m \to l$ **and** $\Delta b = g \to g$ **then** $\Delta f = 0 \to 1$ (based on objects Δx_3 and Δx_4).

Example 2.2. In this example, we are interested in the way quantities (values of condition attributes a, b, c) change and evolve over time. We use the symbol Δa as an abbreviation for "change in" the value of attribute a. By analogy with a system of differential equations of the following form:

$$\frac{\partial a}{\partial t} = f_1(a(t), b(t), c(t)), \quad \frac{\partial b}{\partial t} = f_2(a(t), b(t), c(t)), \quad \frac{\partial c}{\partial t} = f_3(a(t), b(t), c(t)),$$

we consider decision rules with the "then" part specified by the attributes Δa, Δb, and Δc with possible values nc (**n**o **c**hange) and sc (**s**mall **c**hange). The important part of discernibility matrix for Δa is presented in Table 2. The set $\{x_1, x_2, x_5, x_7, x_{10}, x_{12}\}$ contains objects with no change in the value of a at t_2 in comparison with the value of a at t_1 which are discernible from objects with an observed small change sc in the value of a. The indiscernibility class $\{x_3, x_4, x_6\}$ consists of objects with an observed small change sc in the value of a at t_2 in comparison with the value m of a at t_1 and objects indiscernible from such objects. Analogously, $\{x_8, x_9, x_{11}\}$ is the indiscernibility class of objects with an observed small change sc in the value of a at t_2 in comparison with the value h at t_1 and objects indiscernible from such objects. Let us now explain construction of entries of the discernibility matrix using an example of the entry on the intersection of the column labeled by x_1 and the row labeled by $\{x_3, x_4, x_6\}$. This entry consists of all attributes discerning x_1 at the moment t_1 from each element of $\{x_3, x_4, x_6\}$.

Table 2. Important part of discernibility matrix for Δa.

	x_1	x_2	x_5	x_7	x_{10}	x_{12}
$\{x_3, x_4, x_6\}$	a, b, c	a, b, c	b	b	a, b, c	a, b, c
$\{x_8, x_9, x_{11}\}$	a, b, c	a, b, c	a, b, c	a, b, c	b	b

From the second and third row of Table 2 we compute two object related reducts: $\{a, b\}$ and $\{b, c\}$. Note that for the rules relative, e.g., to the indiscernibility class $\{x_3, x_4, x_6\}$ we should preserve the discernibility of this class with the indiscernibility class $\{x_8, x_9, x_{11}\}$ as well as the discernibility of this class with objects from $\{x_1, x_2, x_5, x_7, x_{10}, x_{12}\}$.

Based on the computed reducts we obtain the following decision rules with accuracy less than 1:
if $a = m$ **and** $b = g$ **then** $\Delta a = sc$ (objects x_3, x_4, x_6 and accuracy 0.67),
if $b = g$ **and** $c = m$ **then** $\Delta a = sc$ (objects x_3, x_4, x_6 and accuracy 0.67),
if $a = h$ **and** $b = g$ **then** $\Delta a = sc$ (objects x_8, x_9, x_{11} and accuracy 0.33),
if $b = g$ **and** $c = s$ **then** $\Delta a = sc$ (objects x_8, x_9, x_{11} and accuracy 0.33).

For example, for object x_1 we obtain three object related reducts (from the second column of discernibility matrix (see Table 2)): $\{a\}, \{b\}$ and $\{c\}$.

Decision rules (based on x_1 with accuracy equal to 1):
if $a = l$ **then** $\Delta a = nc$, **if** $b = v$ **then** $\Delta a = nc$, **if** $c = l$ **then** $\Delta a = nc$.

Analogously, we compute decision rules for $\Delta a = nc$ based on x_2, x_5, x_7, x_{10} and x_{12} as well as decision rules for Δb and Δc.

In hierarchical modeling, on each level new information systems are constructed on the basis of already constructed information systems or sensory information systems [35, 38]. For example, starting from sensory information system in which sensory measurements in different moments of time are recorded one can define on the next level an information system in which objects are time windows and attributes are (time-related) properties of these windows (see Figure 1). Operations performed on information systems are defined as unions with constraints [36]. These operations are analogous to joins with constraints considered in databases. For each new constructed information system in hierarchical modeling, changes of one attribute relative to some other ones may be induced using (approximate) Boolean reasoning [15].

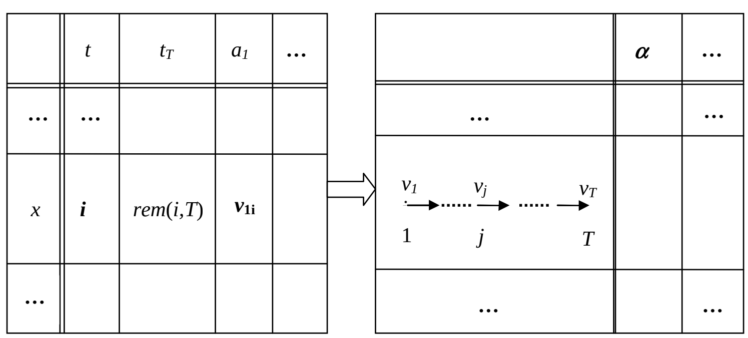

Figure 1. Granulation of time points into time windows. A natural number $T > 0$ is the time window length, $v_j = (v_{1j}, \ldots, v_{Tj})$ for $j = 1, \ldots, T$, $rem(i, T)$ is the remainder from division of i by T, α is an attribute defined over time windows.

It is worth mentioning that quite often this searching process is more sophisticated. For example, one can discover several relational structures (e.g., corresponding to different attributes) and formulas over such structures defining different families of neighborhoods from the original approximation space. As a next step, such families of neighborhoods can be merged into neighborhoods in a new, higher degree approximation space.

This approach is also relevant for Perception Based Computing [38]. For illustration let us consider an explanation of perception included in the book [18]:

> *The main idea of this book is that perceiving is a way of acting. It is something we do. Think of a blind person tap-tapping his or her way around a cluttered space, perceiving that space by touch, not all at once, but through time, by skillful probing and movement. This is or ought to be, our paradigm of what perceiving is.*

Figure 2 illustrates this idea. Note that the challenge is to discover relevant features of histories (i.e., paths of sensory measurement recordings after micro actions 'tap-tapping') for approximation of decision function whose values denote the performed actions on higher level.

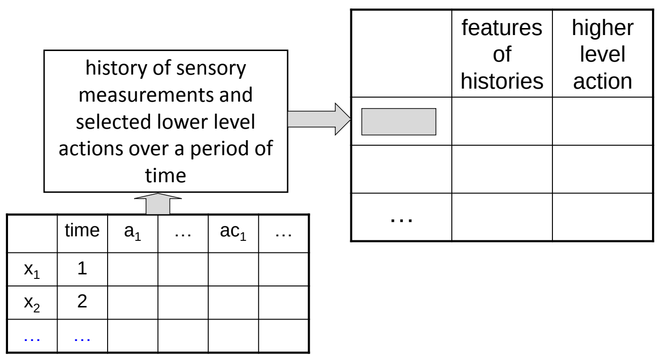

Figure 2. Perception idea

3. Trajectory approximation and adaptation

One can also apply the illustrated idea of approximation of changes and Boolean reasoning for function approximation [33, 35] to transition (function) relation approximation.

First, we introduce some notation. If U^* is a set of objects and $R \subseteq U^* \times U^*$ then by XR we denote R-image of X, i.e., the set $\{y \in U^* : \exists x \in X \; xRy\}$. A sequence $Y_0, \ldots, Y_i, \ldots$, where $Y_0 = X$ and $Y_{i+1} = Y_i R$ for $i \geq 0$ is called R-trajectory starting at X. Let as also assume that A is a set of attributes over U^*, i.e., $a: \; U^* \longrightarrow V_a$ for any $a \in A$, where V_a is a finite set of values of the attribute a.

We consider a case when the transition relation R is partially specified by a sample, i.e. by a decision table $DT = (U_R, A \otimes A, d_R)$, where $U_R \subseteq U^* \times U^*$ is a given sample of pairs of objects, $A \otimes A = \{(a,0) : a \in A\} \cup \{(a,1) : a \in A\}$ is the disjoint union of A (where $(a,0)(x,y) = a(x)$ and $(a,1)(x,y) = a(y)$, for $(x,y) \in U_R$), $d_R(x,y) = +$ if xRy, and $-$, otherwise, for $(x,y) \in U_R$. Hence, we assume that the transition relation is partially specified by a sample of pairs labeled by decision $+$ if the pair belongs to the relation and $-$, otherwise.

From this sample, e.g., a rule based classifier $C_A(x,y)$ may be induced [36, 39, 35], where (x,y) is a pair of objects and $C_A(x,y) = +$ means that y is one of the next predicted states after x obtained by applying the transition relation R, and $C_A(x,y) = -$, otherwise[1].

[1]For simplicity of reasoning, we assume that the induced classifier takes only two values but one can extend our considerations for more values, e.g., by adding the decision value $1/2$ representing borderline cases.

Let us now consider a C_A-trajectory starting at the set $\|\alpha\|_{U^*}$, where α is a boolean combination of descriptors over A and $\|\alpha\|_{U^*}$ denotes the semantics of α over U^* [26, 39].

Now, we would like to find a description of sets Y_i (for $i > 0$) in the C_A-trajectory starting at $\|\alpha\|_{U^*}$. As an illustrative example, we consider the case when α is the conjunction of descriptors from the A-signature of x [26], i.e., from $Inf_A(x) = \{(a, a(x)) : a \in A\}$ for some $x \in U^*$, and we induce the description of $\|\alpha\|_{U^*} C_A$ using boolean combinations of descriptors over A. Hence, from the elementary granule defined by $Inf_A(x)$ we derive a predicted description of its C_A-image $\|\alpha\|_{U^*} C_A$ defined by $C_A(x, y)$. We define this description by a set of attribute value vectors (more formally, by a disjunction of conjunctions of some signatures of objects).

Let us assume that $C_A(x, y)$ is a rule-based classifier based on a rule set $Rule$ (e.g., a subset of minimal decision rules [26]). Each rule is one of the following form:

$$\textbf{if } r \textbf{ then } d = + \quad \text{or} \quad \textbf{if } r \textbf{ then } d = -. \tag{1}$$

The formula r can be decomposed into two parts r_1 and r_2, where r_i corresponds to the the i-th component of (x, y), where $i = 1, 2$. By $D(r_i)$ we denote the set of descriptors in r_i, where $i = 1, 2$.

We restrict our considerations to the case when the decision rules of $C_A(x, y)$ are over attributes A. A more general case, where the decision rules are over attributes constructed from A (see e.g., [5, 15, 37, 39, 38, 42, 46]) will be discussed elsewhere.

Let us assume that x_0 is a new object and we would like to find the description of the image of the elementary granule defined by x_0 relative to $C_A(x, y)$. From the set $Rule$ we select all rules matching x_0, i.e., all rules of the form

$$\textbf{if } r_1 \textbf{ and } r_2 \textbf{ then } d = + \quad \text{or} \quad \textbf{if } r_1 \textbf{ and } r_2 \textbf{ then } d = -, \tag{2}$$

where $Inf_A(x_0)$ matches r_1.

For $v \in \{+, -\}$, we define the following sets:

$$R^v(x_0) = \{D(r_2) : \exists r_1 \ (\textbf{if } r_1 \textbf{ and } r_2 \textbf{ then } d = v) \in Rule \ \text{ and } \ x_0 \text{ is matching } r_1\}. \tag{3}$$

A set X of descriptors over the set of attributes A (i.e., a set of pairs (a, v), where $a \in A$ and $v \in V_a$ [26]) is consistent if the set X is a function. If X is a set of descriptors over A and $\mathcal{X}$ is a family of sets of descriptors over A then X is $\mathcal{X}$-maximal consistent if X is consistent and $X \cup Y$ is not consistent for any $Y \in \mathcal{X}$.

Example 3.1. A set $X = \{(\Delta a, l \to l), (\Delta b, g \to y), (\Delta c, s \to m)\}$ of descriptors over the set of attributes $\Delta A = \{\Delta a, \Delta b, \Delta c\}$ (see Table 1) is $\mathcal{X}$-maximal consistent, where $\mathcal{X} = \{\{(\Delta a, l \to l), (\Delta b, v \to v), (\Delta c, l \to l)\}, \{(\Delta a, m \to m), (\Delta b, y \to y), (\Delta c, m \to m)\}\}$.

Now, we consider all tuples (X, Y, u) such that

1. X is the union of a subset of $R^+(x_0)$ and Y is the union of a subset of $R^-(x_0)$;
2. $u \in INF(B) = \{(a, v) : a \in B \ \& \ v \in V_a\}$, where $B \subseteq A$ and B is disjoint with the sets of attributes occurring in $X \cup Y$;
3. $X \cup Y \cup \{u\}$ is the $(R^+(x_0) \cup R^-(x_0))$-maximal consistent set;

4. voting strategy used in construction of $C_A(x,y)$ applied to the set of all rules from $Rule$ of the form (2), where r_1 is matched by x_0 and $D(r_2) \subseteq X \cup Y$ returns the decision $+$.

From the above construction it follows that the image of the elementary granule $Inf_A(x_0)$ relative to $C_A(x,y)$ can be defined as equal to the set of all extensions of $X \cup Y \cup u$, where (X,Y,u) denotes a tuple satisfying the above conditions.

Iteration of the construction presented above leads to the description by boolean combinations of descriptors of approximation of trajectory defined by $Inf_A(x_0)$ relative to the approximation $C_A(x,y)$ of the transition relation (see Figure 3).

There are several reasons explaining why the searching for approximate description of trajectories over boolean combination of descriptors may be useful. One can consider sets in the C_A-trajectories as granules. Then the granule diameter relative to the granule description by boolean combination of descriptors allows us to characterize uncertainty in identifying states (defined by signatures of objects) corresponding to this granule. The granule diameter can be easily defined if there is given the description of the granule by boolean combination of descriptors. Each such a description is equivalent to a disjunction $\alpha_1 \vee \ldots \vee \alpha_k$ of conjunctions α_i (where $1 \leq i \leq k$) of descriptors from some object signatures. Let us assume that there is given a distance function $\rho_A : INF(A) \times INF(A) \longrightarrow R_+$, where $INF(A) = \{(a,v) : \ a \in A \ \& \ v \in V_a\}$ and R_+ is the set of nonnegative reals. Then the diameter $diam_{\rho_A}(g)$ of the granule g described by the disjunction $\alpha_1 \vee \ldots \vee \alpha_k$ can be defined by $sup_{1 \leq i,j \leq k} \rho_A(u_i, u_j)$, where u_i, u_j denote sets of conjuncts occurring in α_i, α_j, respectively.

Let us consider another approach to inducing of the image of the elementary granule $Inf_A(x_0)$ relative to decision table DT from which the classifier $C_A(x,y)$ is induced rather than to $C_A(x,y)$, explicitly. In this case, having $Inf_A(x_0)$, we select from the decision table DT rows matched by $Inf_A(x_0)$ using only values of attributes of the form $(a,0)$, where $a \in A$. This allows us to create a new decision table with conditional attributes of the form $(a,1)$ for $a \in A$, for all objects y from DT and such that the binary decision function d in this table is defined by $d(y) = 1$ if and only if y belongs to a pair of objects $(x, y$ such that $Inf_A(x)$ was matched by $Inf_A(x_0)$. Now, one can induce a binary rule based classifier C from this the new decision table. Observe that this approach allow us to estimate the diameter of the induced image of $Inf_A(x_0)$ defined by the set of all objects for which the induced classifier C takes the value 1. To do this, we assume that there is a priori selected a distance function between patterns occurring on the left had sides of the induced rules. Then, the the diameter may be estimated as the maximum of distances of patterns defined by rules with the decision 1. The estimated diameter can be used for warning in the construction of the trajectory approximation that the imprecision in approximate description of the trajectory states becomes not acceptable.

Figure 3 also illustrates the necessity of trajectory adaptation. This is caused by the fact that the approximation of the transition relation and the approximation induced from this trajectory are based on a sample of data. However, data may evolve (e.g., they are growing incrementally). Then the classifier and the trajectory approximation induced so far may no longer be of satisfactory quality starting from some moment of time. It is necessary to develop methods allowing us to measure the ‘distance’ between the predicted trajectory P' and the observed trajectory P. If the ‘difference’ is becoming not acceptable, new classifier for transition relation should be induced. This illustrates another important challenge for reasoning about changes.

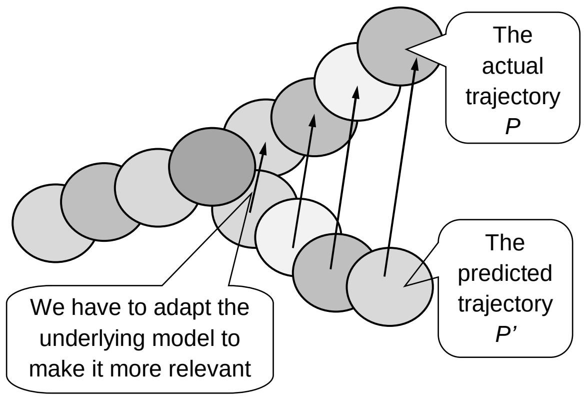

Figure 3. Approximate trajectory adaptation

4. Rough calculus

Let us start with an example explaining motivation for our work on developing of rough calculus. The example is related to the classifier (called *conservative Victoria*) for algorithmic trading EUR/USD on the real-life stream of ticks delivered by the $Oanda\ platform$ (see `www.oanda.com`) in the period of 1.5 year. The classifier was designed in $AdgaMSolutions$ company. In the development of the classifier some of the authors of this paper were involved.

The example is related to stress test functions used, e.g., in financial data mining for estimating the robustness of the induced classifiers. In Figure 4 and Figure 5 are presented the changes of the function, called Sh (*Sharpe Ratio, Sharpe Index, Sharpe measure or reward-to-variability ratio;* `http://en.wikipedia.org/wiki/Sharpe_ratio`) used to measure the quality of the designed classifier relative to deviations of shifting moments of the opening and closing positions.

The different areas presented in Figure 4 and Figure 5 illustrate changes in the values of quality function. Let us describe the situation presented in the figures in more detail.

The value of Sh for this classifier in the period of 1.5 year was close to 3. The classifier was active 24 hours in almost all working days and executed 25000 positions in this period. The results of stress testing presented in Figure 4 and Figure 5 illustrate the robustness of Sh for the developed classifier relative to the time delays in opening and closing positions. In these figures, x-axis is used to show the delay time (in sec) in opening of positions and on y-axis - the delay time (in sec) in closing of positions. The point $(0, 0)$ represents the result for the stream of around 25 000 positions used on Oanda platform. The maps in the figures are used to represent results relative to the real value of Sh which was gained in the period of one and half year. If the result after time delays was, e.g., larger than 105% relative to the result corresponding to the point $(0, 0)$ ($> 105\%$ in the legend in Figure 5) than this point is marked by the color corresponding to this fact. We used two methods of time deviations, namely deterministic and random. In the deterministic method, all opening of positions (corresponding to x-coordinates of points) for all 25 000 positions where shifted exactly by x and y seconds, respectively. In the case of random method, the delay shifts (x, y) were determined by the exponential probability distribution. In this way, we obtained two maps shown in Figure 4 and Figure 5, respectively. The results show that the high robustness of the developed classifier relative to the time delay in opening and closing positions.

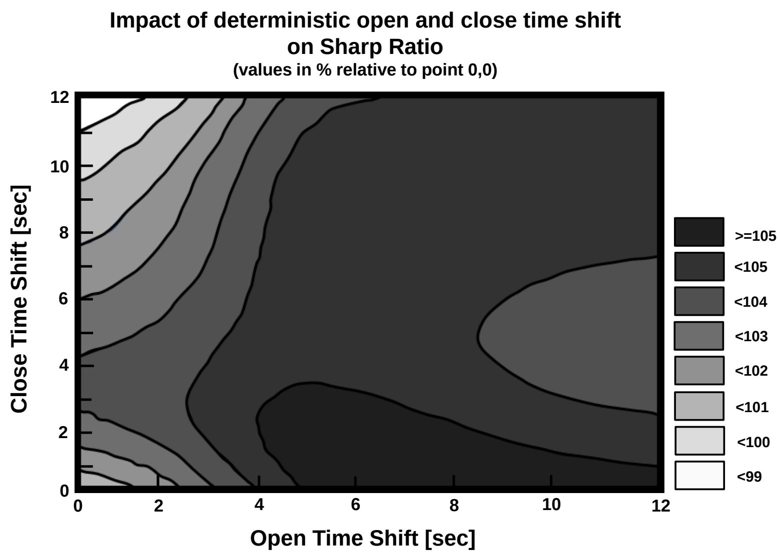

Figure 4. Illustrative example of stress test results. Track record for deterministically rescaled *conservative Victoria* based on real trading performance in the period of 1.5 year

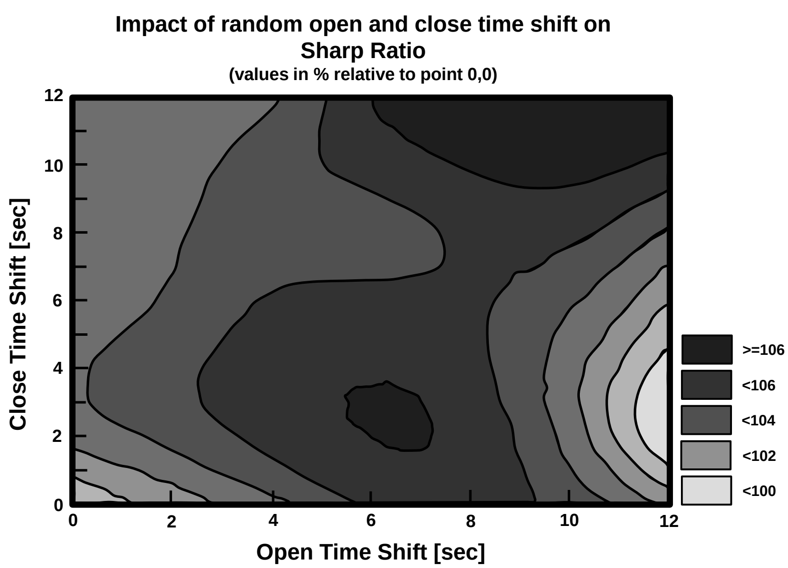

Figure 5. Illustrative example of stress test results. Track record for randomly rescaled *conservative Victoria* based on real trading performance fin the period of 1.5 year

One may observe in Figures 4 and 5 points which are stable relative to deviations of the considered parameters, i.e., points such that even larger deviations of parameters are resulting in small changes in the quality function (i.e. Sh). There are also less stable points. Areas with stable points correspond to areas in which "derivative" of the quality function is "small" while in other areas is relatively "large". In searching for models of derivative concept in our approach we use the rough set-based approximations of changes of functions having only partial information about them. We do not induce from the rough set based approximations global analytical forms of the approximated functions. We try to be "as close as it is possible" to the data what is consistent with the suggestions presented in Epilogue of the book [45] where such a global approach is criticized. One can find here an analogy between this global approach and the "globalization" paradox related to the large class of functions representable by means of Taylor's series (e.g., any polynomial) where local information about behavior of any function is satisfactory for predicting its value and behavior in arbitrary point.

Observe that the quality function may depend on many parameters of the induced classifiers as well as on many other parameters, e.g., dependent on platforms supplying data and market constraints. Certainly, we do not know the analytic form for this function as the function on many parameters defining classifiers as well as parameters characterizing the environment in which the classifiers work. However, we do need to have methods for estimation of changes of the quality function relative to small deviation of these parameters.

It should be now clear for the reader that we need methods for inducing models making it possible to estimate the discussed changes in values of the quality function. These models can play role analogous to derivatives in calculus, e.g., helping in estimating regions where the changes are rapid with small changes of parameters. Our aim is to develop rough set based methods for inducing models describing changes of such functions, called as rough derivatives, assuming that only imperfect, often partial, information about functions is available.

This paper can be treated as a first step toward developing rough calculus [24, 25, 1]. One possible approach to develop a concept of rough derivative is to start from a family of indiscernibility (similarity) relations defined by different choices of sensory information systems rather than a single indiscernibility relation and to characterize changes of function approximation relative to changes of indiscernibility relations. Contrary to the classical calculus, we can not expect to obtain general rules for constructing rough derivatives of $f \circ g$, where $\circ$ is a given operation on functions, from rough derivatives of f and g. However, one may induce such rules relative to given data sets (information systems).

In this section, we present an illustrative example explaining our approach. Let us discuss the concept of derivative of function in the case where the specification of the function is partial, i.e., only a sample of function is given and it is necessary to induce the function approximation. In this case, rough derivatives may be interpreted as the induced approximations of functions of changes. We have already suggested that such approximations may be induced from samples using approximate Boolean reasoning [41].

For some problems related to the quality functions of induced classifiers, changes in their behavior may be considered over approximation spaces from which they are induced.

In an illustrative example, let us consider a family of approximation spaces rather than a single approximation space. For simplicity of reasoning, let us consider a non-increasing chain $\{IND(A_i)\}$ of indiscernibility relations $IND(A_i)$ defined by attribute sets A_i, where $A_i \subseteq A_{i+1}$ (this sequence may be finite or infinite).

Example 4.1. We consider three attribute sets $A_1 = \{\Delta a\}, A_2 = \{\Delta a, \Delta b\}$ and $A_3 = \{\Delta a, \Delta b, \Delta c\}$. We obtain the families of definable sets determined by $IND(A_i)$ as the union of sets from $\Delta U/IND(A_i)$, where $i = 1, 2, 3$. In our example (see Table 1)

$$\Delta U/IND(A_1) = \{\{\Delta x_1, \Delta x_2\}, \{\Delta x_3, \Delta x_4\}, \{\Delta x_5, \Delta x_6, \Delta x_7\}, \{\Delta x_8\},$$

$$\{\Delta x_9, \Delta x_{10}, \Delta x_{11}, \Delta x_{12}\}\},$$

$$\Delta U/IND(A_2) = \{\{\Delta x_1\}, \{\Delta x_2\}, \{\Delta x_3, \Delta x_4\}, \{\Delta x_5\}, \{\Delta x_6\}, \{\Delta x_7\}, \{\Delta x_8\},$$

$$\{\Delta x_9\}, \{\Delta x_{10}\}, \{\Delta x_{11}\}, \{\Delta x_{12}\}\},$$

$$\Delta U/IND(A_3) = \{\{\Delta x_1\}, \{\Delta x_2\}, \{\Delta x_3, \Delta x_4\}, \{\Delta x_5\}, \{\Delta x_6\}, \{\Delta x_7\}, \{\Delta x_8\},$$

$$\{\Delta x_9\}, \{\Delta x_{10}\}, \{\Delta x_{11}\}, \{\Delta x_{12}\}\}.$$

Now, let us consider an approximation of changes Δf of real valued function f relative to $IND(A_i)$ [33, 35].The quality of approximation of Δf is relative to the family of definable sets determined by $IND(A_i)$ and to the acceptable deviation ϵ of Δf on relevant patterns [33, 35]. The approximation quality can be measured, e.g., by the relative size of the boundary region of approximation [33, 35].

Let $\epsilon, \delta > 0$ be given thresholds. We say that the (ϵ, δ)-derivative of f relative to the family $\{A_i\}$ exists if and only if there exists an indiscernibility relation $IND(A_j)$ in this family such that the quality of approximation of Δf [33, 35] is at least δ assuming that in the approximation were used patterns defined by cartesian products of definable sets over $IND(A_i)$ and intervals of reals with length at most ϵ.

The intuition behind this definition is the following: The derivative of a function specified by a sample (of points of its graph) exists if and only if there exists an approximation space (in a given family) allowing us to approximate the changes of function with the high quality by patterns on which the deviation of function changes is small.

Observe that the method of construction of trajectory approximation outlined before may be applied to derivatives of functions (relations) and a given elementary granule, e.g., defined by a new object. The resulting trajectory approximation may be treated as a solution of a rough differential equation determined by derivative of a transition (function) relation specified on a sample of pairs of objects.

The illustrated approach can be extended on arbitrary families of indiscernibility (similarity) relations and, more generally, on families of approximation spaces considered in [33, 35]. For example, in the case of families of indiscernibility relations one should take into account all possible nondecreasing chains of indiscernibility relations. A step toward considering rough integrals is included in [41].

Note also that when the time is progressing some unknown parameters of the environment may change. Then, the induced classifiers should be adopted to these changes. Moreover, there is a need for developing adaptation strategies making it possible to follow changes of induced rough derivatives in time. For example, it may happen that at a given moment of time the robustness for the induced classifier, measured by the estimated rough derivative, is satisfactory but after some period of time, due to changes of some unknown parameters (e.g., on the market) the robustness of the classifier is decreasing and is becoming close to not acceptable level. Then, one should be ready to use some adaptation strategies for adapting the current classifier to this new situation. The hardness of this problem is related to many issues. For example, the classifier adaptation should be performed very quickly, often on-line. Hence, one should be able to gather over time changes in the quality of different components from which the classifier is built (e.g., features or patterns over which it is constructed) to be ready very quickly perform adaptation of the existing classifier to the new situation.

Let us explain this in more detail. We assume that an agent ag has a local knowledge base $KB_{ag}(t)$ changing in time, e.g., $KB_{ag}(t)$ is updated in time by ag with new facts or induced rules related to the interaction of ag with the environment. Let us also assume that the agent ag has an adaptation strategy

Str which on the basis of $KB_{ag}(t)$ is selecting a vector of parameters $\overline{q}$ related, e.g., to features, patterns, conflict resolution strategies between rules voting for different decisions or parameters specifying inclusion measures. These parameters are specifying a class of approximation spaces from which the relevant approximation space used for inducing over it the classifier C is selected. The classifier C should be adapted in time since its performance is dependent on some parameters of the environment which are not known for the agent ag. For example, the robustness of C can be acceptable at time t but it can be not acceptable at $t + \Delta$ for some $\Delta > 0$. Definitely we would like to change the classifier C only when necessary. For example, when the robustness of C is becoming close to unacceptable we should be ready to supply a new more relevant classifier C' instead of C. To be ready to do this efficiently we may perform, e.g., on-line selection of vector of parameters $\overline{q}$ and on the basis of features measuring changes in time of such vectors to predict the right moment for the classifier C adaptation.

The discussed above situation is illustrating that in searching for function approximation and in particular for approximation of their changes, we deal with even more compound situation than considered before when we assumed that a partial information about function is given on a sample of its arguments. To describe this new case in more detail, let us consider a function $f(\overline{x}, \overline{y})$ with two vectors of variables. There is available only a partial information about this function in the following form: there are given values $(z_1, \ldots, z_m)$ of this function for a sample of value vectors of arguments $(\overline{u_1}, \overline{w_1}), \ldots, (\overline{u_m}, \overline{w_m})$ but only values from $\overline{w_1}, \ldots, \overline{w_m}$ are known. For example, the values in $\overline{u_1}, \ldots, \overline{u_m}$ represent values of unknown parameters of the environment. Then, the problem of inducing approximation of functions and in particular the problem of inducing the rough derivatives is becoming much harder.

Let us outline some basic steps in inducing function approximation, in this situation. Our system should be able to discover relevant features (attributes) on the basis of which it will be possible to identify in the space of functions $\{f(\overline{u}, \cdot) : \overline{u} \in V\}$, where V is the set of possible values of $\overline{x}$, some regular subspaces, e.g, consisting of functions which are sufficiently close. These subspaces can be defined by the indiscernibility (or similarity) classes relative to the set of discovered features. Then, the induced function approximation for a particular function from an indiscernibility class should be a good representative on the whole indiscernibility class. However, there are more challenges. When time is progressing, unknown parameters are changing and we should be able to predict in advance the moment of entering into another indiscernibility class (subspace of functions) on which the approximation is already known or can be induced using, e.g., methods of Case-Based Reasoning [6]. This prediction method can be based on collected history of (interactive) computations by agents. This is a challenging problem strongly related to process mining [44, 40]. Observe that in the discussed case the rough derivative is becoming a dynamic complex object (information granule) rather than a static function approximation.

The discussed example is also showing the case of complex interacting granules represented by complex classifiers or function approximations, knowledge base sources and the environment. This is related to the Wistech program presented in [7, 8, 9].

Conclusions

We discussed some aspects of approximate reasoning about changes from data and domain knowledge. This paper can also be treated as a step toward developing rough calculus. We have also presented some challenges which are on the list of our current research topics now.

References

[1] Burgin, M.: Neoclassical Analysis: Calculus Closer to the Real World. Nova Science Publishers, Inc. New York, 2007.

[2] Fisher, M., Gabbay, D., Vila, L. (eds.): Handbook of Temporal Reasoning in Artificial Intelligence, vol. 1. Elsevier, Amsterdam, 2005.

[3] Gabbay, D. M., Hogger, C. J., and Robinson, J. A. (eds.): Handbook of Logic in Artificial Intelligence and Logic Programming, vol. 3, Nonmonotonic Reasoning and Uncertain Reasoning. Oxford: Oxford University Press, Oxford UK, 1994.

[4] Gabbay, D. M., Schlechta, K.: Logical Tools for Handling Change in Agent-Based Systems. Springer, Heidelberg, 2010.

[5] Hastie, T., Tibshirani, R., Friedman, J. H.: The Elements of Statistical Learning: Data Mining, Inference, and Prediction. 2nd ed. Springer, Heidelberg, 2008.

[6] Hüllermeier, E.: Case-Based Approximate Reasoning. Springer, Dordrecht, 2007.

[7] Jankowski, A., Skowron, A.: A wistech paradigm for intelligent systems. LNCS Journal Subline: Transactions on Rough Sets VI: Journal Subline, LNCS 4374 (2007) 94–132, Springer, Heidelberg.

[8] Jankowski, A., Skowron, A.: Logic for artificial intelligence: The Rasiowa – Pawlak school perspective. In: A. Ehrenfeucht, V. Marek, M. Srebrny (eds.) Andrzej Mostowski and Foundational Studies, IOS Press, Amsterdam, 2008, 106–143.

[9] Jankowski, A., Skowron, A.: Wisdom technology: A rough-granular approach. In: M. Marciniak, A. Mykowiecka (eds.) Bolc Festschrift, LNCS 5070, Springer, Heidelberg, 2009, 3–41.

[10] Kowalski, R. A., Sergot, M.J.: A logic-based calculus of events. New Generation Computing 4(1) (1986) 67–95.

[11] Kuipers, B.: Qualitative reasoning: Modeling and simulation with incomplete knowledge. MIT Press, Cambridge, MA, 1994.

[12] McCarthy, J.: Situations, actions and causal laws (Memo 2). Stanford Artificial Intelligence Project, Stanford University, CA 1963.

[13] McCarthy, J.,Hayes, P. J.: Some philosophical problems from the standpoint of artificial intelligence. In: B. Meltzer, D. Michie (eds.) Machine Intelligence 4, Edinburgh University Press, Edinburgh, Scotland, 1969, 463–502.

[14] Mueller, E. T.: Commonsense Reasoning. Morgan Kaufmann, Elsevier, San Francisco CA, 2006.

[15] Nguyen, H. S.: Approximate Boolean reasoning: Foundations and applications in data mining. Transactions on Rough Sets V: Journal Subline, LNCS 3100 (2007) 334–506, Springer, Heidelberg.

[16] Nguyen, H. S., Skowron, A., Stepaniuk, J.: Discovery of changes along trajectories generated by process models induced from data and domain knowledge. In: G. Lindemann, H.-D. Burkhard, L. Czaja, W. Penczek, A, Salwicki, H. Schlingloff, A. Skowron, Z. Suraj (eds.) Proceedings of the Workshop on Concurrency, Specification and Programming (CS&P 2008), vol. 1-3, Gross Vaeter, Germany, September 29-October 1, 2008, Humboldt Universitaet zu Berlin, Informatik-Berichte, Berlin, 2008, 350–362.

[17] Nguyen, H.S., Jankowski, A., Peters, J. F., Skowron, A., Stepaniuk, J., Szczuka, M.: Discovery of Process Models from Data and Domain Knowledge: A Rough-Granular Approach. In: J.T. Yao (ed.) Novel Developments in Granular Computing: Applications for Advanced Human Reasoning and Soft Computation, IGI Global, Hershey, New York, 2010, 16–47.

[18] Noë, A.: Action in Perception. MIT Press, Cambridge, MA, 2004.

[19] Pawlak Z.: Rough functions. ICS PAS Reports 467/81, Institute of Computer Science Polish Academy of Sciences (ICS PAS), Warsaw, Poland, 1981, 1–11.

[20] Pawlak Z.: Rough relations. ICS PAS Reports 435/81, Institute of Computer Science Polish Academy of Sciences (ICS PAS), Warsaw, Poland, 1981, 1–10.

[21] Pawlak Z.: On rough relations. Bulletin of the Polish Academy of Sciences, Technical Sciences 34(9-10) (1986) 587–590.

[22] Pawlak Z. On rough functions. Bulletin of the Polish Academy of Sciences, Technical Sciences 35(5-6) (1987) 249–251.

[23] Pawlak, Z.: Rough Sets: Theoretical Aspects of Reasoning about Data. *System Theory, Knowledge Engineering and Problem Solving*, vol. 9, Kluwer Academic Publishers, Dordrecht, 1991.

[24] Pawlak, Z.: Rough calculus. In: Proceedings of the Second Annual Joint Conference on Information Sciences, September 28 - October 1, 1995, Wrightsville Beach, NC, USA, 1995, 344–345.

[25] Pawlak, Z.: Rough sets, rough functions and rough calculus. In: S.K. Pal and A. Skowron (eds.), Rough Fuzzy Hybridization, A New Trend in Decision Making, Springer-Verlag, Singapore 1999, 99–109.

[26] Pawlak, Z., Skowron, A.: Rudiments of rough sets; Rough sets: Some extensions; Rough sets and Boolean reasoning. Information Sciences 177(1) (2007) 3–27; 28–40; 41–73.

[27] Pedrycz, W., Skowron, A., Kreinovich, V. (eds.): Handbook of Granular Computing. John Wiley & Sons, New York, 2008.

[28] Polkowski, L., Lin, T.Y., Tsumoto, S. (eds.): Rough Set Methods and Applications: New Developments in Knowledge Discovery in Information Systems. *Studies in Fuzziness and Soft Computing*, vol. 56. Springer-Verlag/Physica-Verlag, Heidelberg, 2000.

[29] Ramsay J. O., Silverman B. W.: Applied Functional Data Analysis. Springer, Berlin, 2002.

[30] Reiter, R.: Knowledge in Action: Logical Foundations for Specifying and Implementing Dynamical Systems. The MIT Press, Cambridge MA, 2001.

[31] Sandewall, E.: Features and Fluents. Oxford University Press, Oxford UK, 1994.

[32] Shoham, Y.: Reasoning About Change: Time and Causation from the Standpoint of Artificial Intelligence. MIT Press, Cambridge, MA, 1988.

[33] Skowron, A., Stepaniuk, J.: Approximation spaces in rough-granular computing. Fundamenta Informaticae 100 (2010) 141-157.

[34] Skowron, A., Stepaniuk J.: Data driven approximate reasoning about changes, In: M. Szczuka, L. Czaja, A. Skowron, M. Kacprzak (eds.) Concurrency, Specification and Programming : CS&P'2011: International Workshop, Pultusk, 28 – 30 September 2011, Białystok University of Technology, Humboldt University, Warsaw University, 2011, 477–486.

[35] Skowron, A., Stepaniuk, J., Swiniarski, R.: Modeling rough granular computing based on approximation spaces. Information Sciences 184 (2012) 20–43.

[36] Skowron A., Stepaniuk J., Peters J., Swiniarski R.: Calculi of approximation spaces. Fundamenta Informaticae 72(1-3) (2006) 363–378.

[37] Skowron, A., Szczuka, M.: Toward interactive computations: A rough-granular approach. In: J. Koronacki, S. Wierzchon, Z. Ras, J. Kacprzyk (eds.), Commemorative Volume to Honor Ryszard Michalski. Springer, Heidelberg, 2009, 1-20.

[38] Skowron, A., Wasilewski, P.: Information systems in modeling interactive computations on granules. Theoretical Computer Science 412(42) (2011) 5939–5959.

[39] Stepaniuk J.: Rough–Granular Computing in Knowledge Discovery and Data Mining. Springer, Heidelberg, 2008.

[40] Suraj, Z.: Rough set methods for the synthesis and analysis of concurrent processes. In: Polkowski et al. [28], 379–488.

[41] Szczuka, M., Skowron, A., Stepaniuk, J.: Function approximation and quality measures in rough-granular systems. Fundamenta Informaticae 109(3-4) (2011) 339–354.

[42] Ślęzak, D., Wróblewski, J.: Roughfication of numeric decision tables: The case study of gene expression data. In: Yao, J. T., Lingras, P., Wu, W.-Z., Szczuka, M., Cercone, C., Ślęzak, D. (eds.), Proceedings of the Second International Conference on Rough Sets and Knowledge Technology (RSKT 2007), Toronto, Canada, May 14-16, 2007, LNCS 4481, Springer, Heidelberg, 2007, 316–323.

[43] van Harmelen, F., Lifschitz, V., Porter, B. (Eds.): Handbook of Knowledge Representation. Elsevier, Amsterdam, 2008.

[44] van der Aalst, W.M.P. (ed.): Process Mining Discovery, Conformance and Enhancement of Business Processes. Springer, Heidelberg, 2011.

[45] Vapnik, V.: Statistical Learning Theory. John Wiley & Sons, New York, NY (1998)

[46] Wnek, J., Michalski, R. S.: Hypothesis-driven constructive induction in AQ17-HCI: A method and experiments. Machine Learning 14(1) (1994) 139–168.