

Toward a theory of Granular Computing for human-centred information processing

Andrzej Bargiela, *Member, IEEE*, Witold Pedrycz, *Fellow, IEEE*

Abstract—Human-centred information processing has been pioneered by Zadeh through his introduction of the concept of fuzzy sets in the mid-60s. The insights that were afforded through this formalism have led to the development of the Granular Computing paradigm in the late '90s. Subsequent research has highlighted the fact that many founding principles of Granular Computing have in fact been adopted in other information processing paradigms and indeed in the context of various scientific methodologies. This study expands on our earlier research exploring the foundations of Granular Computing and casting it as a structured combination of algorithmic and non-algorithmic information processing that mimics human, intelligent synthesis of knowledge from information.

Index Terms — Granular Computing, Human-centred information processing, Information Abstraction, Universal Turing Machine.

I. INTRODUCTION

Granular Computing (GrC) is frequently defined in an informal way as a general computation theory for effectively using granules such as classes, clusters, subsets, groups and intervals to build an efficient computational model for complex applications with huge amounts of data, information and knowledge. Though the GrC term is relatively recent, the basic notions and principles of granular computing have appeared under different names in many related fields, such as information hiding in programming, granularity in artificial intelligence, divide and conquer in theoretical computer science, interval computing, cluster analysis, fuzzy and rough set theories, neutrosophic computing, quotient space theory, belief functions, machine learning, databases, and many others. In the past few years, we have witnessed a renewed and fast growing interest in GrC. Granular computing has begun to play important roles in bioinformatics, e-Business, security, machine learning, data mining, high-performance computing and wireless mobile computing in terms of efficiency, effectiveness, robustness

Manuscript received May 31, 2006. This work was supported in part by the Engineering and Physical Sciences (EPSRC, UK) grant EPC51274x

Andrzej Bargiela is with the University of Nottingham, Nottingham, UK (corresponding author: e-mail: Andrzej.Bargiela@cs.nott.ac.uk)

Witold Pedrycz is with the University of Alberta, Edmonton, Canada, email: pedrycz@ualberta.ca

and structured representation of uncertainty.

With the vigorous research interest in the GrC paradigm [3-10, 17, 20-21, 25-28, 33, 36-45] it is natural to see that there are voices calling for clarification of the distinctiveness of GrC from the underpinning constituent disciplines and from the other computational paradigms proposed for large-scale/complex information possessing. Recent contributions by Yao [36-40] attempt to bring together various insights into GrC from a broad spectrum of disciplines and cast the GrC framework as a structured thinking at the philosophical level and structured problem solving at the practical level.

In this study we elaborate on our earlier proposal [11] and look at the roots of Granular Computing in the light of the original insight of Zadeh [41] stating, "... fuzzy information granulation in an intuitive form underlies human problem solving ...".

We suggest that the above insight has strong foundations in axiomatic set theory and theory of computability and it underlies some recent research results linking intelligence to physical computation [1-2]. In fact, re-examining human information processing in this light brings Granular Computing from a domain of computation and philosophy to one of physics and set theory.

The paper is intended to stimulate a broad-ranging discussion in the context of other emerging perspectives on Granular Computing such as the generalized constraint-based computation recently communicated by Zadeh [45].

II. SET THEORETICAL INTERPRETATION OF GRANULATION

The commonly accepted definition of granulation introduced in [17], [21], [38] is:

Definition 1

Information granulation is a grouping of elements based on their indistinguishability, similarity, proximity or functionality.

This definition serves well the purpose of constructive generation of granules but it does little to differentiate granulation from clustering. More importantly however, *Definition 1* implies that the nature of information granules is fully captured by their interpretation as subsets of the original data set within the Intuitive Set Theory of Cantor [13]. Unfortunately, an inevitable consequence of that is that the inconsistencies (paradoxes) associated with intuitive set

theory, such as “cardinality of set of all sets” (Cantor) or “definition of a set that is not a member of itself” (Russel) are imported into the domain of information granulation.

In order to provide a more robust definition of information granulation we follow the approach adopted in the development of axiomatic set theory. The key realization there was that the commonly accepted intuition, that one can form any set one wants, should be questioned. Accepting the departure point of intuitive set theory we can say that, normally, sets are not members of themselves, i.e. normally, $\sim(y \text{ in } y)$. But, the axioms of intuitive set theory do not exclude the existence of “abnormal” sets, which are members of themselves. So, if we consider a set of all “normal” sets: $x = \{y | \sim(y \text{ in } y)\}$ we can axiomatically guarantee the existence of set x :

$$\exists x \forall y (y \in x \leftrightarrow \sim(y \in y))$$

If we then substitute x for y we arrive at a contradiction:

$$\exists x \forall x (x \in x \leftrightarrow \sim(x \in x))$$

So, the unrestricted comprehension axiom of the intuitive set theory leads to contradictions and cannot therefore serve as a foundation of set theory.

A. Zermelo-Fraenkel axiomatization

An early attempt at overcoming the above contradiction was an axiomatic scheme developed by Ernst Zermelo and Abraham Fraenkel [46]. Their idea was to restrict the comprehension axiom schema by adopting only those instances of it, which are necessary for reconstruction of common mathematics. In other words, the standard approach, of using a formula $F(y)$ to collect the set y having the property F , leads to generation of an object that is not a set (otherwise we arrive at a contradiction). So, looking at the problem the other way, they have concluded that the contradiction constitutes a de-facto proof that there are other semantical entities in addition to sets.

The important thesis of this paper is that the semantical transformation of sets through the process of applying some set-forming formula applies also to the process of information granulation and consequently, information granules should be considered as being semantically distinct from the granulated entities. We therefore propose a modified definition of information granulation as follows:

Definition 2

Information granulation is a semantically meaningful grouping of elements based on their indistinguishability, similarity, proximity or functionality.

Continuing with the Zermelo-Fraenkel approach we must legalize some collections of sets that are not sets. Let $F(y, z_1, z_2, \dots, z_n)$ be a formula in the language of set theory (where z_1, z_2, \dots, z_n are optional parameters). We can say that for any values of parameters z_1, z_2, \dots, z_n the formula F defines a “class” A

$$A = \{y \mid F(y, z_1, z_2, \dots, z_n)\}$$

which consists of all y 's possessing the property F . Different values of z_1, z_2, \dots, z_n give rise to different classes. Consequently the axiomatization of set theory involves

formulation of axiom schemas that represent possible instances of axioms for different classes.

The following is a full set of axioms of the Zermelo-Fraenkel set theory:

Z1, Extensionality:

$$\forall x \forall y [\forall z (z \in x \equiv z \in y) \Rightarrow x = y]$$

Asserts that if sets x and y have the same members, the sets are identical.

Z2, Null Set:

$$\exists x \sim \exists y (y \in x)$$

Asserts that there is a unique empty set.

Z3, Pair Set:

$$\forall x \forall y \exists z \forall w (w \in z \equiv w = x \vee w = y)$$

Asserts that for any set x and y , there exists a pair set of x and y , i.e., a set that has only x and y as members.

Z4, Unions:

$$\forall x \exists y \forall z (z \in y \equiv \exists w (w \in x \wedge z \in w))$$

Asserts that for any set x there is a set y containing every set that is a member of some member of x .

Z5, Power Set:

$$\forall x \exists y \forall z [z \in y \equiv \forall w (w \in z \Rightarrow w \in x)]$$

Asserts that for any set x , there is a set y which contains as members all those sets whose members are also elements of x , i.e., y contains all of the subsets of x .

Z6, Infinite Set:

$$\exists x [\emptyset \in x \wedge \forall y (y \in x \Rightarrow \cup\{y, \{y\}\} \in x)]$$

Asserts that there is a set x which contains \emptyset as a member and which is such that, whenever y is a member of x , then $y \cup \{y\}$ is a member of x .

Z7, Regularity:

$$\forall x [x \neq \emptyset \Rightarrow \exists y (y \in x \wedge \forall z (z \in x \Rightarrow \sim(z \in y)))]$$

Asserts that every set is “well-founded”, i.e., it rules out the existence of circular chains of sets as well as infinitely descending chains of sets. A member y of a set x with this property is called a “minimal” element.

Z8, Replacement Schema:

$$\forall x \exists y F(x, y) \Rightarrow \forall u \exists v \forall r (r \in v \equiv \exists s (s \in u \wedge F_{x,y}[s, r]))$$

Asserts that given a formula $F(x, y)$ and $F_{x,y}[s, r]$ as a result of substituting s and r for x and y , every instance of the above axiom schema is an axiom. In other words, given a functional formula F and a set u we can form a new set v by collecting all of the sets to which the members of u are uniquely related by F . It is important to note that elements of v need not be elements of u .

Z9, Separation Schema:

$$\forall u \exists v \forall r (r \in v \equiv r \in u \wedge F_x[r])$$

Asserts that there exists a set v which has as members precisely the members of u which satisfy the formula F . Again, every instance of the above axiom schema is an axiom.

Unfortunately the presence of the two axiom schemas, Z6 and Z7, implies infinite axiomatization of the Zermelo-Fraenkel (ZF) set theory. While it is fully acknowledged that the ZF set theory, and its many variants, has advanced our understanding of cardinal and ordinal numbers and has led to the proof of the property of “well-ordering” of sets (with the help of an additional “Axiom of Choice”) (ZFC), the theory seems unduly complex for the purpose of set-theoretical interpretation of information granules.

B. VonNeumann-Bernays-Goedel axiomatization

A different approach to the axiomatization of set theory designed to yield the same results as ZF but with a finite number of axioms (i.e without the reliance on axiom schemas) has been proposed by von Neumann in 1920 and subsequently has been refined by Bernays in 1937 and Goedel in 1940 [15]. The defining aspect of Von Neumann-Bernays-Goedel set theory (NBG) is the introduction of the concept of “proper class” among its objects. NBG and ZFC are very closely related and, in fact, NBG is a conservative extension of ZFC. In NBG, the proper classes are differentiated from sets by the fact that they do not belong to other classes. Thus in NBG we have

$$x \leftrightarrow \exists y(x \in y)$$

which can be phrased as x is a set if it belongs to either a set or a class.

The basic observation that can be made about NBG is that it is essentially a two-sorted theory; it involves sets (denoted here by lower-case letters) and classes (denoted by upper-case letters). Consequently the above statement about membership assumes one of the forms

$$x \in y \quad \text{or} \quad x \in Y$$

and statements about equality are in the form

$$x = y \quad \text{or} \quad X = Y$$

Using this notation the axioms of NBG are as follows:

N1, Class Extensionality:

$$\forall x[x \in A \leftrightarrow x \in B] \Rightarrow A = B$$

Asserts that classes with the same elements are the same.

N2, Set Extensionality:

$$\forall x[x \in a \leftrightarrow x \in b] \Rightarrow a = b$$

Asserts that sets with the same elements are the same.

N3, Pairing:

$$\forall x \forall y \exists z \forall w (w \in z \equiv w = x \vee w = y)$$

Asserts that for any set x and y , there exists a set $\{x, y\}$ that has exactly two elements x and y . It is worth noting that this axiom allows definition of ordered pairs and taken together with the Class Comprehension axiom, it allows implementation of relations on sets as classes.

N4, Union:

$$\forall x \exists y \forall z (z \in y \equiv \exists w (w \in x \wedge z \in w))$$

Asserts that for any set x there exists a set which contains exactly the elements of elements of x .

N5, Power Set:

$$\forall x \exists y \forall z [z \in y \equiv \forall w (w \in z \Rightarrow w \in x)]$$

Asserts that for any set x , there is a set which contains exactly the subsets of x .

N6, Infinite Set:

$$\exists x [\emptyset \in x \wedge \forall y (y \in x \Rightarrow \cup \{y, \{y\}\} \in x)]$$

Asserts there is a set x , which contains an empty set as an element and contains $y \cup \{y\}$ for each of its elements y .

N7, Regularity:

$$\forall x [x \neq \emptyset \Rightarrow \exists y (y \in x \wedge \forall z (z \in x \Rightarrow \sim (z \in y)))]$$

Asserts that each nonempty set is disjointed from one of its elements.

N8, Limitation of size:

$$\sim x \leftrightarrow |x| = |V|$$

Asserts that if the cardinality of x equals to the cardinality of the set theoretic universe V , x is not a set but a proper class. This axiom can be shown to be equivalent to the axioms of Regularity, Replacement and Separation in NBG. Thus the classes that are proper in NBG are in a very clear sense big, while sets are small.

It should be appreciated that the latter has a very profound implication on computation, which processes proper classes. This is because the classes built over countable sets can be uncountable and, as such, do not satisfy the constraints of the formalism of the Universal Turing Machine.

N9, Class Comprehension schema:

Unlike in the ZF axiomatization, this schema consists of a finite set of axioms (thus giving finite axiomatization of NBG).

Axiom of Sets: For any set x , there is a class X such that $x = X$

Axiom of Complement: For any class X , the complement $V - X = \{x \mid x \notin X\}$

Axiom of Intersection: For any class X and Y the intersection $X \cap Y = \{x \mid x \in X \wedge x \in Y\}$ is a class.

Axiom of Products: For any classes X and Y , the class $X \times Y = \{(x, y) \mid x \in X \wedge y \in Y\}$ is a class. This axiom provides actually for more than is needed for representing relations on classes. What is actually needed is just that $V \times Y$ is a class.

Axiom of Converses: For any class X , the class $Conv1(X) = \{(y, x) \mid (x, y) \in X\}$ and the class $Conv2(X) = \{(y, (x, z)) \mid (x, (y, z)) \in X\}$ exist.

Axiom of Association: For any class X , the class $Assoc1(X) = \{((x, y), z) \mid (x, (y, z)) \in X\}$ and the class $Assoc2(X) = \{(w, (x, (y, z))) \mid (w, ((x, y), z)) \in X\}$ exist.

Axiom of Ranges: For any class X the class $Rng(X) = \{y \mid (\exists x (x, y) \in X)\}$ exists.

Axiom of Membership: The class $[\in] = \{(x, y) \mid x \in y\}$ exists.

Axiom of Diagonal: The class $[=] = \{(x, y) \mid x = y\}$ exists.

This axiom can be used to build a relation asserting the equality of any two of its arguments and consequently used to handle repeated variables.

With the above finite axiomatization the NBG theory can be adopted as a set theoretical basis for Granular Computing. Such a formal framework prompts a powerful insight into the essence of granulation namely that **the granulation process transforms the semantics of the granulated entities**, mirroring the semantical distinction between sets and classes.

The semantics of granules is derived from the domain that has, in general, higher cardinality than the cardinality of the granulated sets. Although, at first, it might be a bit surprising to see that such a semantical transformation is an essential part of information granulation, in fact, we can point to a common framework of many scientific disciplines which have evolved by abstracting from details inherent to the underpinning scientific discipline and developing a vocabulary of terms (proper classes) that have been verified by the reference to real-life (ultimately to the laws of physics). An example of granulation of detailed information into semantically meaningful granules might be the consideration of cells and organisms in Biology rather than consideration of molecules, atoms or sub-atomic particles when studying the physiology of living organisms.

The operation on classes in NBG is entirely consistent with the operation on sets in the intuitive set theory. The principle of abstraction implies that classes can be formed out of any statement of the predicate calculus, with the membership relation. Notions of equality, pairing and such, are thus matters of definitions (a specific abstraction of a formula) and not of axioms. In NBG a set represents a class if every element of the set is an element of the class. Consequently, there are classes that do not have representations.

We suggest therefore that the advantage of adopting NBG as a set theoretical basis for Granular Computing is that it provides a framework within which one can discuss a hierarchy of different granulations without running the risk of inconsistency. For instance one can denote a “large category” as a category of granules whose collection and collection of morphisms can be represented by a class. A “small category” can be denoted as a category of granules contained in sets. Thus, we can speak of “category of all small categories” (which is a “large category”) without the risk of inconsistency.

A similar framework for a set-theoretical representation of granulation is offered by the theory of types published by Russell in 1937, [29]. The theory assumes a linear hierarchy of types: with type 0 consisting of objects of undecided type and, for each natural number n , type $n+1$ objects are sets of type n objects. The conclusions that can be drawn from this

framework with respect of the nature of granulation are exactly the same as that drawn from the NBG.

C. Mereology

An alternative framework for the formalisation of Granular Computing, that of mereology, has been proposed by other researchers. The roots of mereology can be traced to the work of Edmund Husserl (1901) [16] and to the subsequent work of Polish mathematician, Stanislaw Lesniewski, in the late 1920s [18-19]. Much of this work was motivated by the same concerns about the intuitive set theory that have spurred the development of axiomatic set theories (ZF, NBG and others) [15, 46].

Mereology replaces talk about “sets” with talk about “sums” of objects, objects being no more than the various things that make up wholes. However such a simple replacement results in an “intuitive mereology” that is analogous to “intuitive set theory”. Such “intuitive mereology” suffers from paradoxes analogous to Russel’s paradox (we can ask: If there is an object whose parts are all the objects that are not parts of themselves; is it a part of itself?). So, one has to conclude that the mere introduction of the mereological concept of “partness” and “wholeness” is not sufficient and that mereology requires axiomatic formulation.

Axiomatic formulation of mereology has been proposed as a first-order theory whose universe of discourse consists of *wholes* and their respective *parts*, collectively called objects [31, 34]. A mereological system requires at least one primitive relation e.g. dyadic Parthood, *x is a part of y*, written as Pxy . Parthood is nearly always assumed to partially order the universe. An immediate defined predicate is *x is a proper part of y*, written $PPxy$, which holds if Pxy is true and Pyx is false. An object lacking proper parts is an *atom*. The mereological universe consists of all objects we wish to consider and all of their proper parts. Two other predicates commonly defined in mereology are Overlap and Underlap. These are defined as follows:

- Oxy is an overlap of x and y if there exists an object z such that Pzx and Pzy both hold.
- Uxy is an underlap of x and y if there exists an object z such that x and y are both parts of z (Pxz and Pyz hold).

With the above predicates axiomatic mereology defines the following axioms:

M1, Parthood is Reflexive: Asserts that object is part of itself.

M2, Parthood is Antisymmetric: Asserts that if Pxy and Pyx both hold, then x and y are the same object.

M3, Parthood is Transitive: Asserts that if Pxy and Pyz hold then Pxz hold.

M4, Weak Supplementation: Asserts that if $PPxy$ holds, there exists a z such that Pzy holds but Ozx does not.

M5, Strong Supplementation: Asserts that if Pyx does not hold, there exists a z such that Pzy holds but Ozx does not.

M5a, Atomistic Supplementation: Asserts that if Pxy does not hold, then there exists an atom z such that Pzx holds but Ozy does not.

Top: Asserts that there exists a “universal object”, designated W , such that PxW holds for any x .

Bottom: Asserts that there exists an atomic “null object”, designated N , such that PNx hold for any x .

M6, Sum: Asserts that if Uxy holds, there exists a z , called the “sum of x and y ”, such that the parts of z are just those objects which are parts of either x or y .

M7, Product: Asserts that if Oxy holds, there exists a z , called the Product of x and y ”, such that the parts of z are just those objects which are parts of both x and y .

M8, Unrestricted Fusion: Let f be a first order formula having one free variable. Then the fusion of all objects satisfying f exists.

M9, Atomicity: Asserts that all objects are either atoms or fusions of atoms.

It is clear that if “parthood” in mereology is taken as corresponding to “subset” in set theory, there is some analogy between the above axioms of *classical extensional mereology* and those of standard Zermelo-Fraenkel set theory. However, there are some philosophical and common sense objections to some of the above axioms; e.g. transitivity of Parthood (M3). Also the set of above axioms is not minimal since it is possible to derive Weak Supplementation axiom (M4) from Strong Supplementation axiom (M5).

Axiom M6 implies that if the universe is finite or if *Top* is assumed, then the universe is closed under sum. Universal closure of product and of supplementation relative to W requires *Bottom*. W and N are evidently the mereological equivalents of the universal and the null sets. Because sum and product are binary operations, M6 and M7 admit the sum and product of only a finite number of objects. The *fusion* axiom, M8, enables taking the sum of infinitely many objects. The same holds for product. If M8 holds, then W exists for infinite universes. Hence *Top* needs to be assumed only if the universe is infinite and M8 does not hold. It is somewhat strange that while the *Top* axiom (postulating W) is not controversial, the *Bottom* axiom (postulating N) is. Lesniewski rejected *Bottom* axiom and most mereological systems follow his example. Hence, while the universe is closed under sum, the product of objects that do not overlap is typically undefined. Such defined mereology is equivalent to Boolean algebra lacking a 0. Postulating N generates mereology in which all possible products are definable but it also transforms extensional mereology into a Boolean algebra without a null-element [34].

The full mathematical analysis of the theories of parthood is beyond the intended scope of this paper and the reader is referred to the recent publication by Pontow and Shubert [47] in which the authors prove, by set theoretical means, that there exists a model of general extensional mereology where arbitrary summation of attributes is not possible. However, it is clear from the axiomatization above that the question about the existence of a universal entity containing all other entities and the question about the existence of an empty entity as part of all existing entities are answered very differently by set theory and mereology. In set theory the existence of a universal entity is contradictory and the existence of an empty

set is mandatory while in mereology the existence of a universal set is stipulated by the respective fusion axioms and the existence an empty entity is denied. Also it is worth noting that in mereology there is no straightforward analogue to the set theoretical is-element-of relation [47].

So, taking into account the above we suggest the following answer to the underlying questions of this section: Why granulation is necessary?; and Why the set-theoretical representation of granulation is appropriate?

- The **concept of granulation is necessary** to denote the semantical transformation of granulated entities in a way that is analogous to semantical transformation of sets into classes in axiomatic set theory;
- Granulation interpreted in the context of axiomatic set theory is **very different from clustering**, which is focused on the mere grouping of similar entities; and
- The set-theoretical interpretation of granulation enables **consistent** representation of a hierarchy of information granules.

III. ABSTRACTION AND COMPUTATION

Having established an argument for semantical dimension to granulation, one may ask; how is the meaning (semantics) instilled into real-life information granules? Is the meaning instilled through an algorithmic processing of constituent entities or is it a feature that is independent of algorithmic processing?

The answers to these questions are hinted by the von Neumann’s *limitation of size principle*, mentioned in the previous section, and are more fully informed by Turing’s theoretical model of computation. In his original paper, [35], Turing has defined computation as an automatic version of doing what people do when they manipulate numbers and symbols on paper. He proposed a conceptual model which included: a) an arbitrarily long tape from which one could read as many symbols as needed (from a countable set); b) means to read and write those symbols; c) a countable set of states storing information about the completed processing of symbols; and d) a countable set of rules that governed what should be done for various combinations of input and system state. A physical instantiation of computation, envisaged by Turing, was a human operator (called computer) who was compelled to obey the rules a)-d) above. There are several important implications of Turing’s definition of computation. First, the model implies that computation explores only a subset of capabilities of human information processing. Second, the constraint that the input and output is strictly symbolic (with symbols drawn from a countable set) implies that the computer does not interact directly with the environment. These are critical limitations meaning that Turing’s computer on its own is unable (by definition) to respond to external, physical stimuli. *Consequently, it is not just wrong but essentially meaningless to speculate on the ability of Turing machines to perform human-like intelligent interaction with the real world.*

To phrase it in mathematical terms, the general form of computation, formalized as a Universal Turing Machine

(UTM), is defined as mapping of sets that have at most cardinality \mathcal{N}_0 (infinite, countable) onto sets with cardinality \mathcal{N}_0 . The practical instances of information processing, such as clustering of data, typically involve a finite number of elements both in the input and output sets and represent therefore a more manageable mapping of a finite set with cardinality max_1 onto another finite set with cardinality max_2 . The hierarchy of computable clustering can be therefore represented as in Figure 1.

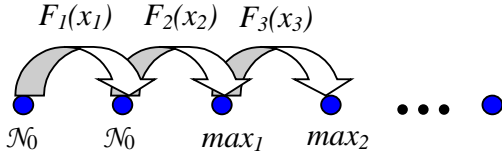


Figure 1. Cardinality of sets in a hierarchy of clusterings implemented on UTM.

The functions $F_1(x_1) \rightarrow x_2, F_2(x_2) \rightarrow x_3, F_3(x_3) \rightarrow x_4$ represent mappings of:

- infinite (countable) input set onto infinite (countable) output set;
- infinite (countable) input set onto finite output set; and
- finite input set onto finite output set, respectively.

The functional mappings, deployed in the process of clustering, reflect the criteria of similarity, proximity or indistinguishability of elements in the input set and, on this basis, grouping them together into a separate entity to be placed in the output set. In other words, the functional mappings generate data abstractions on the basis of pre-defined criteria and consequently represent UTM computation. However, we need to understand how these criteria are selected and how they are decided to be appropriate in any specific circumstance. Clearly, there are many ways of defining similarity, proximity or indistinguishability. Some of these definitions are likely to have good real-world interpretation while others may be difficult to interpret or indeed may lead to physically meaningless results.

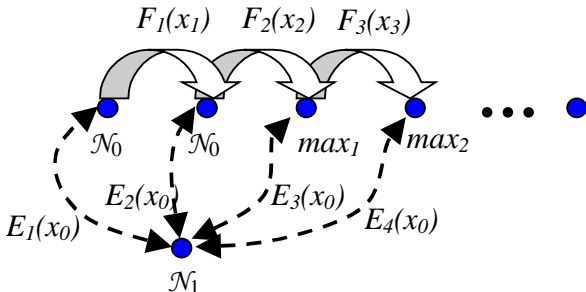


Figure 2. Mapping of abstractions from the real-world domain (cardinality \mathcal{N}_1) onto the sets of clusters.

We suggest that the process of instilling the real-world interpretation into data structures generated by functional

mappings $F_1(x_1) \rightarrow x_2, F_2(x_2) \rightarrow x_3, F_3(x_3) \rightarrow x_4$, involves reference to the real-world, as illustrated in Figure 2. This is represented as execution of “experimentation” functions $E_i(x_0)$. These functions map the real-world domain x_0 , which has cardinality \mathcal{N}_1 (infinite, continuum), onto sets x_1, x_2, x_3, x_4 , respectively.

At this point, it is important to underline that the experimentation functions $E_1(x_0) \rightarrow x_1, E_2(x_0) \rightarrow x_2, E_3(x_0) \rightarrow x_3, E_4(x_0) \rightarrow x_4$, are not computational, in UTM sense, because their domain have cardinality \mathcal{N}_1 . So, the process of defining the criteria for data clustering, and implicitly instilling the meaning into information granules, relies on the laws of physics and not on the mathematical model of computation. Furthermore the results of experimentation do not depend on whether the experimenter understands or is even aware of the laws of physics. It is precisely because of that we consider the experimentation functions as providing objective evidence.

IV. EXPERIMENTATION AS A PHYSICAL COMPUTATION

Recent research [30] has demonstrated that analogue computation, in the form of recurrent analogue neural networks (RANN) can exceed the abilities of a UTM if the weights in such neural networks are allowed to take continuous rather than discrete weights. While this result is significant in itself, it relies on the assumptions about the continuity of parameters that are difficult to verify. So, although the brain looks remarkably like a RANN, drawing any conclusions about the hyper-computational abilities of the brain, purely on the grounds of structural similarities, leads to the same questions about the validity of the assumptions about continuity of weights. Of course, this is not to say that these assumptions are not valid, they may well be valid, but we just highlight that this has not been demonstrated yet in a conclusive way.

A pragmatic approach to bridging the gap between the theoretical model of hyper-computation, as offered by RANN, and the human, intelligent information processing (which by definition is hyper-computational) has been proposed by Bains [1, 2]. Her suggestion was to reverse the original question about hyper-computational ability of systems and to ask: *if the behaviour of physical systems cannot be replicated using Turing machines, how can they be replicated?* The answer to this question is surprisingly simple: *we can use inherent computational ability of physical phenomena in conjunction with the numerical information processing ability of UTM.* In other words, the readiness to refine numerical computations in the light of objective evidence coming from a real-life experiment, instills the ability to overcome limitations of the Turing machine. We have advocated this approach in our earlier work, [6], and have argued that the hypercomputational power of granular computing is equivalent to “keeping open mind” in intelligent, human information processing.

In what follows we describe the model of physical computation, as proposed in [1], and cast it in the framework of Granular Computing.

A. A Model of Physical Computation

We define a system under consideration as an identifiable collection of connected elements. A system is said to be *embodied* if it occupies a definable volume and has a collective contiguous boundary. In particular, a UTM with its collection of input/output data, states and collection of rules, implementing some information processing algorithm, can be considered to be a system G whose physical instantiations may refer to specific I/O, processing and storage devices as well as specific energy states. The matter, space and energy outside the boundaries of the embodied system are collectively called the *physical environment* and will be denoted here by P .

A *sensor* is any part of the system that can be changed by physical influences from the environment. Any forces, fields, energy, matter, etc. that may be impinging on the system, are collectively called the *sensor input* ($i \in X$), even where no explicitly-defined sensors exist.

An *actuator* is any part of the system that can change the environment. Physical changes to the embodied system that manifest themselves externally (e.g. emission of energy, change of position, etc.) are collectively called the *actuator output* ($h \in Y$) of G . A coupled pair of sensor input i_t and actuator output h_t represents an instance of experimentation at time t and is denoted here as E_t .

Since the system G , considered in this study, is a computational system (modeled by UTM) and since the objective of the evolution of this system is to mimic human intelligent information processing we will define G_t as the *computational intelligence function* performed by the embodied system G . Function G_t maps the input to the output at specific time instances t resolved with arbitrarily small accuracy $\delta t > 0$, so as not to preclude the possibility of a continuous physical time. We can thus formally define the computational intelligence function as:

$$G_t(i_t) \rightarrow h_{t+\delta t}$$

In the proposed physical model of computation we stipulate that G_t causes only an immediate output in response to an immediate input. This stipulation does not prevent one from implementing some plan over time but it implies that a controller that would be necessary to implement such plan is part of the intelligence function. The adaptation of G_t in response to evolving input i_t can be described by the *computational learning function*, $L_G: L_G(G_t, i_t) \rightarrow G_{t+\delta t}$.

Considering now the impact of the system behaviour on the environment we can define the *environment reaction function* mapping system output h (environment input) to environment output i (system input):

$$P_t(h_t) \rightarrow i_{t+\delta t}$$

The adaptation of the environment P over time can be described by the *environment learning function*, $L_P: L_P(P_t, h_t) \rightarrow P_{t+\delta t}$.

B. Physical Interaction between P and G

The interaction between the system G and its physical environment P may be considered to fall into one of two classes: *real interaction* and *virtual interaction*. Real interaction is a pure physical process in which the output from

the environment P is in its entirety forwarded as an input to the system G and conversely the output from G is fully utilized as input to P .

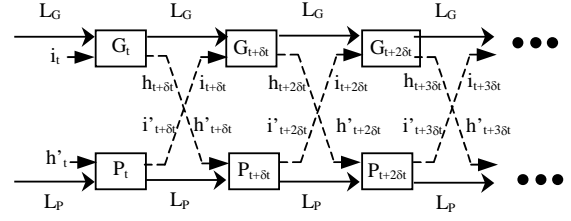


Figure 3. Evolution of a system in an experiment with physical interaction.

Referring to the notation in Figure 3, real interaction is one in which $h_t = h'_t$ and $i_t = i'_t$ for all time instances t . Unfortunately, this type of interaction does not accept the limitations of the UTM, namely the processing of only a pre-defined set of symbols rather than a full spectrum of responses from the environment. Consequently, this type of interaction places too high demands on the information processing capabilities of G and, in practical terms, is limited to interaction of physical objects as governed by the laws of physics. In other words the intelligence function and its implementation are one and the same.

C. Virtual Interaction

An alternative mode of interaction is *virtual interaction*, which is mediated by symbolic representation of information. Here we use the term *symbol* as it is defined in the context of UTM: a letter or sign taken from a finite alphabet to allow distinguishability.

We define V_t as the *virtual computational intelligence function*, analogous to G_t in terms of information processing, and V'_t as the *complementary computational intelligence function*, analogous to G_t in terms of communication with the physical environment. With the above definitions we can lift some major constraints of physical interactions, with important consequences. The complementary function V'_t can implement an interface to the environment, filtering real-life information input from the environment and facilitating transfer of actuator output, while the virtual intelligence function V_t can implement UTM processing of the filtered information. This means that i_t does not need to be equal to i'_t and h_t does not need to be equal to h'_t . In other words, inputs and outputs may be considered selectively rather than in their totality. The implication being that many physically distinguishable states may have the same symbolic representation at the virtual computational intelligence function level. The relationship between the two components of the computational intelligence is illustrated in Figure 4.

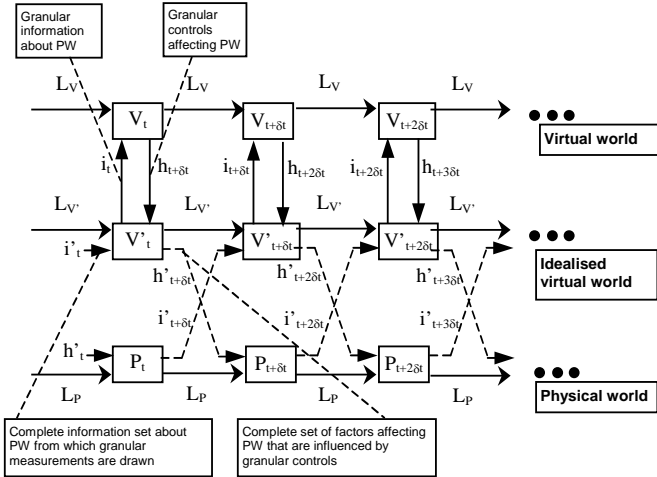


Figure 4. Evolution of a system in an experiment with virtual interaction.

It should be pointed out that typically we think of the complementary function V'_t as some mechanical or electronic device (utilizing the laws of physics in its interaction with the environment) but a broader interpretation that includes human perception, as discussed by Zadeh in [41], is entirely consistent with the above model. In this broader context the UTM implementing the virtual computational intelligence function can be referred to as *computing with perceptions* or *computing with words* (see Figure 5).

Another important implication of the virtual interaction model is that V and P need not have any kind of conserved relationship. This is because only range/modality of subsets of i'_t and h'_t attach to V and these subsets are defined by the choice of sensor/actuator modalities. So, we can focus on the choice of modalities, within the complementary computational intelligence function, as a mechanism through which one can exercise the assignment of semantics to both inputs and outputs of the virtual intelligence function. To put it informally, the complementary function is a facility for defining a “language” in which we chose to communicate with the real world.

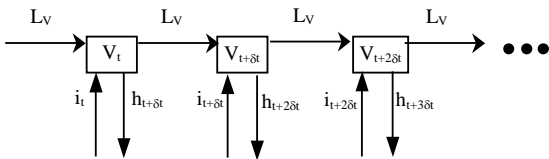


Figure 5. The paradigm of computing with perceptions within the framework of virtual interaction.

Of course, to make the optimal choice (one that allows undistorted perception and interaction with the physical environment), it would be necessary to have a complete knowledge of the physical environment. So, in its very nature the process of defining the semantics of inputs and outputs of the virtual intelligence function is iterative and involves evolution of our understanding of the physical environment.

V. GRANULAR COMPUTATION

An important conclusion from the discussion above is that the discovery of semantics of information abstraction, referred to sometimes as structured thinking, or a philosophical dimension of granular computing, can be reduced to physical experimentation. This is a very welcome development as it gives a good basis for the formalization of the granular computing paradigm.

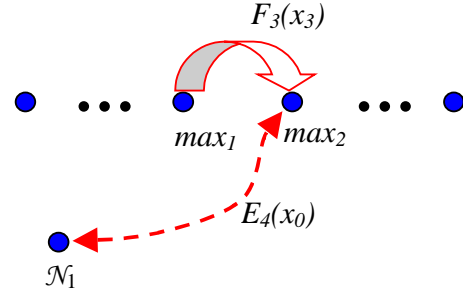


Figure 6. An instance of Granular Computing involving two essential components: algorithmic clustering and empirical evaluation of granules.

We propose that Granular Computing is defined as a **structured combination** of algorithmic abstraction of data and non-algorithmic, empirical verification of the semantics of these abstractions. This definition is general in that it neither prescribes the mechanism of algorithmic abstraction nor it elaborates on the techniques of experimental verification. Instead, it highlights the essence of combining computational and non-computational information processing. Such a definition has several advantages:

- it emphasizes the complementarity of the two constituent functional mappings;
- it justifies the hyper-computational nature of GrC;
- it places physics alongside set theory as the theoretical foundations of GrC;
- it helps to avoid confusion between GrC and purely algorithmic data processing while taking full advantage of the advances in algorithmic data processing.

A specific instance of Granular Computing is illustrated in Figure 6.

VI. PRACTICAL EXAMPLE OF GRANULAR COMPUTATION

We illustrate here the application of the granular computation, cast in the formalism of set theory, to a practical problem of analyzing traffic queues. A 3-way intersection from the SCOOT - UTC system in Mansfield, Nottinghamshire, UK, is represented in Figure 7. The three inductive loops, labeled here as “east”, “west” and “south” provide counts of vehicles passing over them. The counts are then integrated to yield a measure of traffic queues on the corresponding approaches to the junction. A representative sample of the resulting 3-dimensional time series of traffic queues is illustrated in Figure 8.

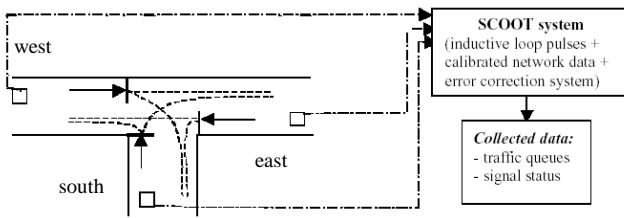


Figure 7. A 3-way intersection with measured traffic queues.

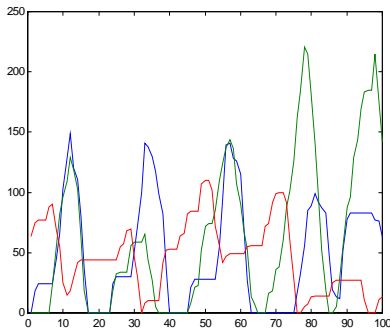


Figure 8. A subset of 100 readings from the time series of traffic queues data.

It is quite clear that, on its own, data depicted in Figure 8 reflects primarily the signaling stages of the junction. This, initial impression is reinforced if we plot the traffic queues on a 2-dimensional plane and apply some clustering technique (such as FCM) to identify prototypes that are the best (in terms of the given optimality criterion) representation of data. The prototypes, denoted as small circles in Figure 9, indicate that the typical operation of the junction involves simultaneously increasing and simultaneously decreasing queues on the “east” and “west” junction. This of course corresponds to “red” and “green” signaling stages. It is worth emphasizing here that the above prototypes can be considered as a simple subset of the original numerical data since the nature of the prototypes is entirely consistent with that of the original data.

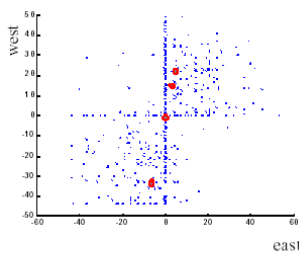


Figure 9. FCM prototypes as subset of the original measurements of traffic queues

Unfortunately, within this framework, the interpretation of the prototype indicating “zero” queue in both “east” and “west” direction is not very informative. In order to uncover the meaning of this prototype we resort to a granulated view of data. Figure 10 represents traffic queue data that has been granulated based on maximization of information density measure discussed in [12]. The semantics of the original

readings is now changed from point representation of queues into interval (hyperbox) representation of queues. In terms of set theory, we are dealing here with a class of hyperboxes, which is semantically distinct from point data.

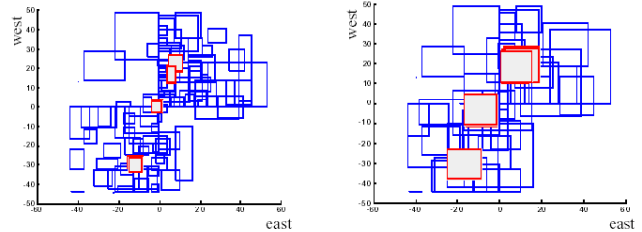


Figure 10. Granular FCM prototypes representing a class that is semantically distinct from the original point data.

Applying FCM clustering to granular data results in granular prototypes denoted on Figure 10 as rectangles with bold boundaries overlaid on the granulated data. In order to ascertain that the granulation does not distort the essential features of the data, different granulation parameters have been investigated and a representative sample of two granulations is depicted in Figure 10. The three FCM prototypes lying in the areas of simultaneous increase and simultaneous decrease of traffic queues have identical interpretation as the corresponding prototypes on Figure 9. However, the central prototype highlights a physical property of traffic that was not captured by the numerical prototype.

Figure 11 illustrates the richer interpretation of the central prototype. It is clear that the traffic queues on the western approach are increasing while the traffic queues on the eastern approach are decreasing. This is caused by the “right turning” traffic being blocked by the oncoming traffic from the eastern junction. It is worth noting that the prototype is unambiguous about the absence of a symmetrical situation where an increase of traffic queues on the eastern junction would occur simultaneously with the decrease of the queues on the western junction. The fact that this is a 3-way junction with no right turn for the traffic from the eastern junction has been captured purely from the granular interpretation of data. Note, that the same cannot be said about the numerical data illustrated in Figure 9.

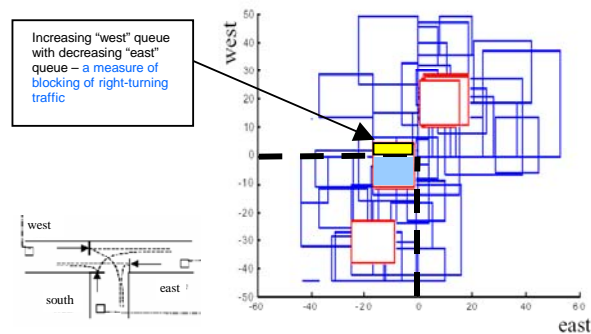


Figure 11. Granular prototype capturing traffic delays for “right turning” traffic.

As we have argued in this paper, the essential component of Granular Computing is the experimental validation of the semantics of the information granules. We have conducted a planned experiment in which we placed counting devices on the entrance to the “south” link. The proportion of the vehicles entering the “south” junction (during the green stage of the “east-west” junction) to the count of vehicles on the stop line on the “west” approach represents a measure of the right turning traffic. The ratio of these numerical counts was 0.1428. A similar measurements derived from two different granulations depicted in Figure 10 was 0.1437 and 0.1498. We conclude therefore that the granulated data captured the essential characteristics of the right turning traffic and that the granulation parameters do not affect the result to a significant degree.

A more extensive experimentation could involve verification of the granular measurement of right turning traffic for drivers that have different driving styles in terms of acceleration and gap acceptance. Although we do not make any specific claim in this respect, it is possible that the granulation of traffic queue data would need to be parameterized with the dynamic driver behaviour data. Such data could be derived by differentiating the traffic queues (measurement of the speed of change of queues) and granulating the resulting six-dimensional data.

ACKNOWLEDGEMENTS

The authors would like to acknowledge Prof. Zadeh’s comments on the ideas reported in this paper during the ECMS 2006 conference. Also, we acknowledge many constructive comments from the anonymous referees.

Financial support from the EPSRC (UK) for both authors is gratefully acknowledged.

REFERENCES

- [1] S. Bains, “Intelligence as Physical Computation”, AISBJ 1 (3), 2003.
- [2] S. Bains, *Physical computation and embodied artificial intelligence*, Ph.D. thesis, The Open University, January 2005.
- [3] A. Bargiela, W. Pedrycz, *Granular Computing – An Introduction*, Kluwer Academic Publishers, 2002.
- [4] A. Bargiela, W. Pedrycz, “From numbers to information granules: A study of unsupervised learning and feature analysis”, In: H. Bunke, A. Kandel (eds.) *Hybrid Methods in Pattern Recognition*, World Scientific, 2002.
- [5] A. Bargiela, W. Pedrycz, “Recursive information granulation: Aggregation and interpretation issues”, *IEEE Trans. on Systems Man and Cybernetics SMC-B*, 33, 1, 96-112, 2003.
- [6] A. Bargiela, “Hypercomputational characteristics of Granular Computing”, *1st Warsaw Int. Seminar on Intelligent Systems - WISIS 2004*, Invited lectures, Warsaw, May 2004.
- [7] A. Bargiela, W. Pedrycz, M. Tanaka, “An inclusion/exclusion fuzzy hyperbox classifier”, *Int. J. Knowledge-based and Intelligent Engineering Systems*, 8, 2, 91-98, 2004.
- [8] A. Bargiela, W. Pedrycz, Hirota K., “Granular prototyping in fuzzy clustering”, *IEEE Transactions on Fuzzy Systems*, 12, 5, 697-709, 2004.
- [9] A. Bargiela, W. Pedrycz, “A model of granular data: a design problem with the Tchebyshev FCM”, *Soft Computing*, 9, 3, 155-163, 2005.
- [10] A. Bargiela, W. Pedrycz, “Granular mappings”, *IEEE Trans. on Systems Man and Cybernetics SMC-A*, vol 35, 2, 288-301, 2005.
- [11] A. Bargiela, W. Pedrycz, “The roots of Granular Computing”, *Proc. IEEE Granular Computing Conf., Atlanta*, May 2006, 741-744.
- [12] A. Bargiela, I. Kosonen, M. Pursula, E. Peytchev, “Granular analysis of traffic data for turning movements estimation”, *Int.J. of Enterprise Information Systems*, vol.2, 2, 2006, 13-27.
- [13] Cantor. G., “Über einen Satz aus der Theorie der stetigen Mannigfaltigkeiten”, *Göttinger Nachr.* 1879, 127-135
- [14] D. Dubois, H. Prade, R. Yager (Eds.), *Fuzzy Information Engineering*, Wiley, New York, 1997.
- [15] K. Goedel, *The Consistency of the Axiom of Choice and of the Generalized Continuum Hypothesis with the Axioms of Set Theory*. Princeton University Press, Princeton, NJ. , 1940.
- [16] E. Husserl, *Logische Untersuchungen. Phänomenologie und Theorie der Erkenntnis*, in *Logical Investigations*, 1970.
- [17] M. Inuiguchi, S. Hirano, S. Tsumoto (Eds.), *Rough Set Theory and Granular Computing*, Springer, Berlin, 2003.
- [18] S. Lesniewski, Über Funktionen, deren Felder Gruppen mit Rücksicht auf diese Funktionen sind, *Fundamenta Mathematicae XIII*: 1929, 319-332.
- [19] S. Lesniewski, Grundzuge eines neuen Systems der Grundlagen der Mathematik, *Fundamenta Mathematicae XIV*, 1929, 1-81.
- [20] T.Y. Lin, “Granular computing on binary relations” in L. Polkowski and A. Skowron (Eds.), *Rough sets in knowledge discovery: Methodology and applications*, Physica-Verlag, Heidelberg, 286-318, 1998.
- [21] T.Y. Lin, Y.Y. Yao, L.A. Zadeh (Eds.), *Data mining, Rough sets and Granular Computing*, Physica-Verlag, 2002.
- [22] M. Matsuyama, *How to summarize and analyze small data sets*, Kawaboku Publishers, Tokyo, 1937.
- [23] Z. Pawlak, *Rough sets: Theoretical aspects of reasoning about data*, Kluwer Academic Publishers, Dordrecht, 1991.
- [24] Z. Pawlak, “Granularity of knowledge, indiscernibility and rough sets”, *Proc. IEEE Conf. Fuzzy Systems*, 106-110, 1998.
- [25] W. Pedrycz, *Fuzzy control and fuzzy systems*, Wiley, New York, 1989.
- [26] W. Pedrycz, F. Gomide, *An Introduction to Fuzzy Sets*, Cambridge, MIT Press, Cambridge, MA, 1998.
- [27] W. Pedrycz, M.H. Smith, A. Bargiela, Granular clustering: A granular signature of data, *Proc. 19th Int. (IEEE) Conf. NAFIPS’2000*, Atlanta, 69-73, 2000.
- [28] W. Pedrycz, A. Bargiela, “Granular clustering: A granular signature of data”, *IEEE Trans. on Systems Man and Cybernetics*, 32, 2, 212-224, 2002.
- [29] B. Russel, “New foundations for mathematical logic”, *American Mathematical Monthly*, 2, 1937.
- [30] H. Siegelmann, “Neural networks and analogue computation: Beyond the Turing limit”, *Birkhauser*, Boston, MA, 1999.
- [31] P. Simons, *Parts: A Study in Ontology*, Oxford Univ. Press, 1987.
- [32] A. Skowron, “Toward intelligent systems: calculi of information granules”, *Bulletin of International Rough Sets Society*, 5, 9-30, 2001.
- [33] A. Skowron, J. Stepaniuk, “Information granules: towards foundations of granular computing,” *International Journal of Intelligent Systems*, 16, 57-85, 2001.
- [34] A. Tarski, Foundations of the geometry of solids, in *Logic, Semantics, Metamathematics: Papers 1923-38*, Hackett, 1984 (1956).
- [35] A. Turing, “On computable numbers, with an application to the entscheidungsproblem”, *Proceedings of the London Mathematical Society*, 42, 230-265, 1936.
- [36] Y.Y. Yao, “Granular computing using neighbourhood systems” in *Advances in Soft computing; Engineering Design and Manufacturing*, Roy, R., Furuhashi, T., Chawdry PK. (Eds.), Springer Verlag, 539-553, 1999.
- [37] Y.Y. Yao, J.T. Yao, “Granular computing as a basis for consistent classification problems”, *Proceedings PAKDD’02 Workshop on Foundations of Data Mining*, 101-106, 2002.
- [38] Y.Y. Yao, “Granular computing”, *Computer Science*, Vol. 31, 1-5, *Proc. 4th Chinese National Conf. On Rough Sets and Soft Computing*, 2004.
- [39] Y.Y. Yao, “A partition model of Granular Computing”, *LNCS Transactions on Rough Sets*, vol. 1, 232-253, 2004.
- [40] Y.Y. Yao, “Perspectives on Granular Computing”, *Proc. IEEE Conf. On Granular Computing*, vol. 1, 85-90, 2005.
- [41] L.A. Zadeh, “Toward a theory of fuzzy information granulation and its centrality in human reasoning and fuzzy logic”, *Fuzzy Sets and Systems*, 90, 111-127, 1997.
- [42] L.A. Zadeh, “Fuzzy sets and information granularity”, in N. Gupta, R. Ragade, R. Yager (Eds.) *Advances in fuzzy set theory and applications*, North-Holland, 1979.

- [43] L.A. Zadeh, "Fuzzy sets", *Inform and Control*, 8, 338-353, 1965.
- [44] L.A. Zadeh, "From computing with numbers to computing with words — from manipulation of measurements to manipulation of perceptions" *Int. J. of Applied Math and Computer Science*, 307-324, 12, 3, 2002.
- [45] L.A. Zadeh, "Granular Computing – The Concept of Generalized Constraint-Based Computation", (*private communication*)
- [46] E. Zermelo, "Untersuchungen ueber die Grundlagen der Mengenlehre", *Math. Annalen*, 1908, 65, 261-281.
- [47] C.Pontow, R. Schubert, "A mathematical analysis of parthood", *Data and Knowledge Engineering*, 59, 1, 2006, pp. 107-138.

