Chapter 24

GRANULAR COMPUTING AND ROUGH SETS

An Incremental Development

Tsau Young ('T. Y.') Lin
Department of Computer Science
San Jose State University
San Jose, CA 95192
tylin@cs.sjsu.edu

Churn-Jung Liau
Institute of Information Science
Academia Sinica, Taipei 115, Taiwan
liaucj@iis.sinica.edu.tw

Abstract This chapter gives an overview and refinement of recent works on binary granular computing. For comparison and contrasting, granulation and partition are examined in parallel from the prospect of rough Set theory (RST).The key strength of RST is its capability in representing and processing knowledge in table formats. Even though such capabilities, for general granulation, are not available, this chapter illustrates and refines some such capability for binary granulation. In rough set theory, quotient sets, table representations, and concept hierarchy trees are all set theoretical, while in binary granulation, they are special kind of pretopological spaces, which is equivalent to a binary relation Here a pretopological space means a space that is equipped with a neighborhood system (NS). A NS is similar to the classical NS of a topological space, but without any axioms attached to it[1].

Keywords: Granular computing, rough set, binary relation, equivalence relation

1. Introduction

Though the label, granular computing is relatively recent, the notion of granulation has in fact been appeared, under different names, in many related fields, such as programming, divide and conquer, fuzzy and rough set theories, pretopological spaces, interval computing, quantization, data compression, chunking, cluster analysis, belief functions, machine learning, databases, and many others. In the past few years, we have seen a renewed and fast growing interest in Granular Computing (GrC). Many applications of granular computing have appeared in fields, such as medicine, economics, finance, business, environment, electrical and computer engineering, a number of sciences, software engineering, and information science.

Granulation seems to be a natural problem-solving methodology deeply rooted in human thinking. Many daily "things" have been routinely granulated into sub"things;" human body has been granulated into head, neck, and so forth; geographic features into mountains, planes, and others. The notion is intrinsically fuzzy, vague and imprecise. Mathematicians idealized it into the notion of partitions, and developed it into a fundamental problem-solving methodology; it has played major roles throughout the entire history of mathematics.

Nevertheless, the notion of partitions, which absolutely does not permit any overlapping among its granules, seems to be too restrictive for real world problems. Even in natural science, classification *does* permit small degree of overlapping; there are beings that are both appropriate subjects of zoology and botany. A more general theory is needed.

Based on Zadeh's grand project on granular mathematics, during his sabbatical leave (1996/1997) at Berkeley, Lin focused on a subset of granular mathematics, which he called granular computing (Zadeh, 1998). To stimulate research on granular computing, a special interest group, with T. Y. Lin as its Chair, was formed within BISC (Berkeley Initiative in Soft Computing). Since then, granular computing has evolved into an active research area, generating many articles, books and presentations at conferences, workshops and special sessions. This chapter is devoted to present some of such development over the past few years.

There are two possible approaches: (1) One is starting from fuzzy side and moving down, and (2) the other one is from extreme crisp side and moving up. In this chapter, we take the second approach incrementally. Recall that algebraically a partition is an equivalence relation, so a natural next step is the binary granulation defined by a binary relation. For contrasting, we may call a partition A-granulation and the more general granulation B-granulation.

2. Naive Model for Problem Solving

An obvious approach to a large-scaled computing problem is: (1) To divide the problem into subtasks, might be point by point and level by level. (2) To elevate or abstract the problem into concept/knowledge spaces, could be in multilevels. (3) To integrate the solutions of subtasks and quotient tasks (knowledge spaces) of several levels

2.1 Information Granulations/Partitions

In the first step, we select an appropriate system of granulation/partition so that only the summaries of granules/equivalence classes may enter into the higher level computing. The information in data space is transformed to a concept space, possibly in levels, which may be locally at each point or globally at eh whole universe (Lin, 2003b). Classically, we granulate by partitioning (no overlapping on granules). Such examples are plentiful: in mathematics (quotient groups, quotient rings and etc. (Birkhoff and MacLane, 1977)), in theoretical computer science (divide-and-conquer (Aho *et al.*, 1974)), in software engineering (the structural, object oriented, and component based design and programming (Szyperski, 2002)), in artificial intelligence (Hobbs, 1985; Zhang and Zhang, 1992), in rough set theory (Pawlak, 1991) among others. However, these are all partition based, where no overlapping of granules is permitted. As we have observed, even in biology, classification does allow some overlapping. The focus of this presentation will be on *non-partition theory*, but only in an epsilon step away from partitioning method.

2.2 Knowledge Level Processing and Computing with Words

The information in each granule is summarized and the original problem is re-expressed in terms of symbols, words, predicates or linguistic variables. Such re-expressing is often referred to as knowledge representations. Its processing has been termed computing with symbols (table processing, computing with words, knowledge level processing, even precisiated natural language, depending on the complexity of the representations.

In this chapter, we are computing on the space of granules or "quotient space." in which each granule is represented by a word that carries different degree of semantics. For partition theory, the knowledge representation is in table format (Pawlak, 1991) and its computation is syntactic in nature. For binary granulation, that we have focused here, is semantic oriented. We expand and streamline the previous works (Lin, 1998a; Lin, 1998b; Lin, 2000); the main idea is to transfer the computing with words into computing with symbols.

Loosely speaking computing with symbols or symbolic computing is an "axiomatic" Computing: all rules of computing symbols are determined by the axioms. The computation follows the formal specifications. Such computing occurs only in an ideal situation. In many real world applications, unfortunately, such as non-linear computing, the formal specifications are often unavailable. So computing with words are needed; it can be processed informally. Semantics of words often may not be completely or precisely formalized. Their semantic computing is often carried out in the systems with human helps (the semantics of symbols are not implemented). Human enforced semantic computing are common in data processing environment.

2.3 Information Integration and Approximation Theory

Most applications require the solutions be presented in the same level as input data. So the solutions often need to be integrated from subtasks (solutions in granules) and quotient tasks (solutions in the spaces of granules). For some applications, such as Data Mining and some rough set theory, are aimed at high level information; in such cases this step can be skipped. In general, the integration is not easy. In partition world, many theories have been developed in mathematics; e. g., extension functors. The approximation theory of pretopological spaces and rough set theory can be regarded as in this step.

3. A Geometric Models of Information Granulations

For understanding the general idea, in this section, we recall and refine a previous formalization in (Lin, 1998a). The goal is to formalize Zadeh's informal notion of granulation mathematically.

As original thesis is informal, the best we could do is to present, hopefully, convincing arguments. We believe our formal theory is very close to the informal one. According to Zadeh (1996):

> Information granulation involves partitioning a class of objects(points) into granules, with a granule being a clump of objects (points) which are drawn together by indistinguishability, similarity or functionality.

We will literally take Zadeh's informal words as a formal definition of granulation. We observe that:

1. A granule is a group of objects that are draw together (by indistinguishability, similarity or functionality).

   The phrase "drawn together" implicitly implies certain level of symmetry among the objects in a granule. Namely, if p is drawn towards q, then q is also drawn towards p.

   Such symmetry, we believe, is imposed by imprecise-ness of natural language. To avoid such an implications, we will rephrase it to "drawn towards an object p," so that it is clear the reverse may or may not be true. So we have first revision:

2. A granule is a group $B(p)$ of objects that are draw toward an object p. Here p varies through every object in the universe.

3. Such an association between object p and a granule $B(p)$ induces a map from the object space to power set of object space. This map has been called a binary granulation (BG).

4. Geometric View:

   We may use geometric terminology and refer to the granule as a neighborhood of p, and the collection $\{B(p)\}$ a binary neighborhood system (BNS). It is possible that $B(p)$ is an empty set. In this case we will simply say p has no neighborhood (abuse of language; to be very correct, we should say p has an empty neighborhood). Also it is possible that different points may have the same neighborhood (granule) $B(p) = B(q)$. The set of all q, where $B(q)$ is equal to $B(p)$, is called the centers $C(p)$ of $B(p)$.

5. Algebraic View:

   Consider the set $R = \{(p, u)\}$, where u in $B(p)$ and p in U. It is clear that R is a subset of $U \times U$, hence defines a binary relation (BR), and vice versa.

PROPOSITION 1 *A binary neighborhood system (BNS), A binary granulation (BG), and a binary relation (BR) are equivalent.*

From the analysis given above, we propose the following mathematical model for information granulation.

DEFINITION 1 *By a (single level) information granulation defined on a set U we mean a binary granulation (binary neighborhood system, binary relation) defined on U.*

Let us goes a little bit further. Note that the binary relation is a mathematical expression of Zadeh's "indistinguishability, similarity or functionality." We abstract the three properties into a list of abstract binary relations $\{B_j \mid j$ run through some index set $\}$, where each B_j is a binary relation.

Note that at each point p, each B_j induces a neighborhood $B_j(p)$. Some may be empty, or identical. By removing empty set and duplications, the family have been we re-indexed $N_i(p)$. As in the single level case, we will define directly the granulation

$$N : U \rightarrow 2^{2^U}; p \mapsto \{B_i(p) \mid i \text{ run through some index set }\}.$$

The collection $\{B_i(p)\}$ is called a neighborhood system(NS)or (LNS); the latter one is used to distinguish itself from the neighborhood system (TNS) of a topological space (Lin, 1989a; Lin, 1992).

DEFINITION 2 *By a local multi-level information granulation defined on U, we mean a neighborhood system (NS) is defined on U. By a global multi-level information granulation defined on U, we mean a set of BG is defined on U.*

All notions can be fuzzified. The right way to look at this section is to assume implicitly there is a modifier "crisp/fuzzy" to all notions presented above.

4. Information Granulations/Partitions

Technically, granular computing is actually computing with constraints. Especially in "infinite world", granulation is often given in terms of constraints. In this chapter, we concerns primarily with constraints that are mathematically represented as binary relations

4.1 Equivalence Relations(Partitions)

Partition is a decomposition of the universe into a family of disjoint subsets. They are called equivalence classes, because a partition induces an equivalence relation and vice versa. In this chapter, we will view the equivalence class in a special way. Let $A \subseteq U \times U$ be an equivalence relation (a reflexive, symmetric and transitive binary relation). For each p, let

$$A_p = \{v \in U : pAv\} \tag{24.1}$$

A_p is the equivalence class containing p, and will be called A-granule for the purpose of contrasting with general cases. Elements in A_p are equivalent to each other. Let us summarize the discussions in:

$$A : U \longrightarrow 2^U : p \longmapsto A_p \tag{24.2}$$

PROPOSITION 2 *An equivalence relation on $U \Leftrightarrow$ a partition on U*

In RST, the pair (U, A) is called an approximation space and its topological properties are studied.

4.2 Binary Relation (Granulation) - Topological Partitions

In (Lin, 1998b), we observe that there is a derived partition for each BNS, that is, the map $B : V \longrightarrow 2^U; p \longmapsto B(p)$ induces a partition on V; the equivalence class $C(p) = B^{-1}(B(p))$ is the center of $B(p)$. In the case $V = U$, the $B(p)$ is the neighborhood of $C(p)$, and $C(p)$ consists of all the points that have the same neighborhood. So $B(p) = B((C(p))$. We observe that $\{C(p)\}$ is a partition. Since each $B(p)$ is a neighborhood of the set $C(p)$. The quotient set is a BNS (Lin, 1989a). We will call the collection of $C(p)$ topological partition with the understanding that there is a neighborhood $B(p)$ for each equivalence class $C(p)$. The neighborhoods capture the interaction among equivalence classes (Lin, 2000).

4.3 Fuzzy Binary Granulations (Fuzzy Binary Relations)

In (Lin, 1996), we have discussed various fuzzy sets. In this chapter, a fuzzy set is uniquely defined by its membership function. So a fuzzy set is a w-sofset, if we use the language of the cited paper.

A fuzzy binary relation is a fuzzification of a binary relation. Let I be the unit interval [0, 1]. Let FBR be a fuzzy binary relation, that is, there is a membership function: $FBR : V \times U \longrightarrow I : (p, u) \longmapsto r$. For each $p \in V$, there is a fuzzy set whose membership function $FM_p : U \longrightarrow I$ is defined by $FM_p(u) = FBR(p, u)$, we call FM_p a fuzzy binary neighborhood/set.

Again, we can view the idea geometrically. We assume a fuzzy binary neighborhood system (FBNS) is imposed for V on U. For each object $p \in V$, we associate a fuzzy subset, denoted by $FB(p) \subseteq U$. In other words, we have a map $FB : V \longrightarrow FZ(U) : p \longmapsto FB(p)$, where $FZ(U)$ means all fuzzy subsets on U. $FB(p)$ is called a fuzzy binary neighborhood and FB a fuzzy binary granulation (FBG) and the collection $\{FB(p) | p \in V\}$ a fuzzy binary neighborhood system (FBNS).

It is clear that given a map FB, there is a binary relation FBR such that $FM_p = FB(p)$. So as in crisp cases, from now on we will use algebraic and geometric terms interchangeably. FB, FBNS, FBG, and FBR are synonyms.

5. Non-partition Application - Chinese Wall Security Policy Model

In 1989 IEEE Symposium on Security and Privacy, Brewer and Nash (BN) proposed a very intriguing security model, called Chinese Wall Security Policy (CWSP) model. Intuitively BN's idea was to build a family of impenetrable walls, called Chinese Walls, among the datasets of competing companies so that no datasets that are in conflict can be stored in the same side of Chinese Walls; this is BN's requirements and will be called *Aggressive (Strong) Chinese Wall Security Policy (ACWSP) Model.*

The methods are based on the formal analysis of the binary relations (CIR) of conflict of interests. Roughly, BN granulated the data sets by CIR and assumed the granulation was a partition. CIR is rarely an equivalence relation, for example, a company cannot be self conflicting; so reflexivity can never met by CIR. So a modified model, called an aggressive Chinese Wall Security Policy model (ACWSP) is proposed (Lin, 1989b). However, in that paper, the essential strength of **ACWSP** model had not brought out. With recent development in GrC, **ACWSP** model was refined (Lin, 2003a), and successfully captured the intuitive intention of BN "theory."

CWSP Model is essentially a Discretionary Access Control Model (DAC). The central notion of DAC is that owner of an object has discretionary authority on the access rights of that objects. The owner X of the dataset x may grant the read access of x to a user Y who owns a dataset y. The use Y may make a copy, Copy-of-x, in y. Even in the strict DAC model, this is permissible (Osborn*et al.*, 2000)). We have summarized the above grant access procedure, including making a copy, as a direct information flow (DIF) from X or x to Y or y respectively.

Let O be the set of all objects (corporate data),X and Y are typical objects in O. $CIR \subseteq O \times O$ represents the binary relation of conflict of interests. We will consider the following properties:

- CIR-1: CIR is symmetric.
- CIR-2: CIR is anti-reflexive.
- CIR-3: CIR is anti-transitive.

5.1 Simple Chinese Wall Security Policy

In (Brewer and Nash, 1988), Section "Simple Security", p. 207, BN asserted that *"people are only allowed access to information which is not held to conflict with any other information that they already possess."* So if $(X, Y) \notin$ CIR, then X and Y could be assigned to one single agent. So we assume that information in X and Y have been disclosed to each other (since one agent knows both). So outside of CIR-class, there are direct information flows between any two objects.

DEFINITION 3 *Simple CWSP : Direct Information Flow (DIF) may flow between X and Y if and only if $(X, Y) \notin$ CIR,*

Simple CWSP is a requirement on DIF, it does not prevent information flow between X and Y indirectly. So we need composite information flow (CIF). By a CIF, we mean information flow between X and Y via a sequence of DIF's. An information flow from X to Y is called a *malicious Trojan horse*, if Simple CWSP is imposed on X and Y.

DEFINITION 4 *(Strong) ACWSP: CIF may flow between X and Y if and only if $(X, Y) \notin$ CIR,*

Next, let us quote a theorem from (Lin, 2003a).

THEOREM 1 Chinese Wall Security Theorem, *If CIR is symmetric, anti-reflexive and anti-transitive, then Simple CWSP implies (Strong) ACSWP.*

6. Knowledge Representations

At the current states, knowledge representations are mainly in table or tree formats. So the knowledge level processing is basically table processing. The main works, we will present here is the extension of the representation theory of equivalence relations to binary relations.

6.1 Relational Tables and Partitions

(Pawlak, 1982) and (Lee, 1983) observed that: A relational table is a knowledge representation of a universe of entities. Each column induces a partition on the universe; n columns induce n partitions. Here, we will explore the converse. How could we represent a finite set of partitions? The central idea is to assign meaningful name (a summary) to each equivalence class (Lin, 1998a; Lin, 1998b; Lin, 1999b).

We will illustrate the idea by example: Let $U = \{id_1, id_2, \ldots, id_9\}$ be a set of 9 balls with two partitions:

(1) $\{\{id_1, id_2, id_3\}, \{id_4, id_5\}, \{id_6, id_7, id_8, id_9\}\}$

(2) $\{\{id_1, id_2\}, \{id_3\}, \{id_4, id_5\}, \{id_6, id_7, id_8, id_9\}\}$

We name the first partition COLOR, (because it is the best summarization of the given partition from physical inspection).

COLOR = Name($\{\{id_1, id_2, id_3\}, \{id_4, id_5\}, \{id_6, id_7, id_8, id_9\}\}$)

Next, we will name each equivalence class to reflect its characteristic. We name the first equivalence class

Red = Name($\{id_1, id_2, id_3\}$),

because each ball of this group has red color (appears to human). Note that this name reflects human's observation and meaningful to human only; its meaning (such as light spectrum) is not implemented or stored in the system. In AI, the term COLOR or Red are called semantic primitive (Barr and Feigenbaum, 1981). The same intent leads to the following names

Orange = Name($\{id_4, id_5\}$)

Yellow = Name($\{id_6, id_7, id_8, id_9\}$)

Next, we give names to the second partition, again by its characteristics (appear to human):

WEIGHT = Name($\{\{id_1, id_2\}, \{id_3\}, \{id_4, id_5\}, \{id_6, id_7, id_8, id_9\}\}$)

W1 = Name($\{id_1, id_2\}$)

W2 = Name($\{id_3\}$)

W3 = Name($\{id_4, id_5\}$)

W4 = Name($\{id_6, id_7, id_8, id_9\}$)

Base on these names, we have Table 24.1:

Table 24.1. Constructing an Information table by naming each partition and equivalence class

U	COLOR	WEIGHT
id_1	Red	W1
id_2	Red	W1
id_3	Red	W2
id_4	Orange	W3
id_5	Orange	W3
id_6	Yellow	W4
id_7	Yellow	W4
id_8	Yellow	W4
id_9	Yellow	W4

The first tuple can be interpreted as follows: the first ball belongs to the group that is labeled Red, and another group whose weight is labeled W1. We can do the same for rest of the tuples. This table is a classical bag relation.

The goal of this chapter is to generalize this naming methodology to general granulations. The word-representation of partitions is a very clean representation; each name (word) represents an equivalence class uniquely and independently. In next section, we will investigate the representations of binary relations, in which names have overlapping semantics.

6.2 Table Representations of Binary Relations

Real world granulation often cannot be expressed by equivalence relations. For example, the notions of "near","similar", and "conflict" are not equivalence relations. So there are intrinsic needs to generalize the theory of partition (RST) to the theory of more general granulation (granular computing). In this section, we will explain how to represent a finite set of binary granulations (binary relations) into a table format. So we can extend the relational theory from partitions to binary granulations. Most of the results are recall and refinements of the results observed in (Lin, 1998a; Lin, 1998b; Lin, 1999b; Lin, 2000).

The representation of a partition is rested on two properties:

(a) Each object p belongs to an equivalence class (the union of equivalence class covers the whole universe)

(b) No object belongs to two equivalence classes (equivalence class are pairwise disjoint)

The important question is: Does the family of binary granules have the same properties as equivalence classes? Obviously, a granulation does satisfy (a), but not (b), because granules may overlap each other. We need a different way to look at the problem: we restate the two properties into the following form:

- Each object belongs to one and only one equivalence class

If we assign each equivalence class a meaningful name, then each object is associated with a unique name (attribute value). Such an assignment construct one column of the table representation. Each equivalence relation get a column. So n equivalence relations construct a table of n columns.

With these observations, we can state a similar property for the binary granulation. Let B be a binary granulation

- Each object, $p \in V$, is assigned to one and only one B-granule $B_p \in 2^U$; $B : p \mapsto B_p$.

If we assign each B-granule a meaningful name, then each object is associated with a unique name (attribute value).

$$p(\in V) \ {}^{B}\!\!\rightarrow \ B_p(\in 2^U) \ {}^{\text{Name}}\!\!\rightarrow \ \text{Name}(B_p)(\in Dom(B)) \qquad (24.3)$$

$$p \rightarrow \text{Name}(B_p)(\in Dom(B)) \qquad (24.4)$$

Such an association allows us to represent

- a finite set of binary granulations by a “relational table”, called granular table.

Note that we did not use the relationships “$\in$”. Instead, we use the *assignment of neighborhoods (binary granules).*

We will illustrate the idea by modifying the last example. In binary granulation each p is associated with a unique binary neighborhood B_p. The following neighborhoods are given.

$$B_{id_1} = B_{id_2} = B_{id_3} = \{id_1, id_2, id_3, id_4, id_5\}$$

$$B_{id_4} = B_{id_5} = \{id_1, id_2, id_3, id_4, id_5, id_6, id_7, id_8, id_9\}$$

$$B_{id_6} = B_{id_7} = B_{id_8} = B_{id_9} = \{id_4, id_5, id_6, id_7, id_8, id_9\}.$$

By examining the characteristic of each binary neighborhood, we assign their names as follows:

Having-RED =Name(B_{id_1})=Name(B_{id_2})=Name(B_{id_3})

Having-RED+YELLOW =Name(B_{id_4})= Name(B_{id_5})

Having-YELLOW =Name(B_{id_6})=Name(B_{id_7})=Name(B_{id_8})= Name(B_{id_9})

For illustration, let us trace the journey of id_1: It is an object of V, and is moved to a subset, B_{id_1}, then stop at the name, Having-RED, in notation,

$$id_1 \ {}^{B}\!\!\longrightarrow \ B_{id_1} \ {}^{\text{Name}}\!\!\longrightarrow \ \text{Having-RED}.$$

By tracing every object of V, we get the second column of Table 24.2. For the third column, we use the same partition and naming scheme as in the previous section; so the third column is exactly the same as that in Table 24.1. The results are shown in Table 24.2.

Table 24.2. Granular table: Construct granular table by naming each binary granulations and binary granules

BALLs	Granulation 1	Granulation 2
id_1	Having-RED	W1
id_2	Having-RED	W1
id_3	Having-RED	W2
id_4	Having-RED+YELLOW	W3
id_5	Having-RED+YELLOW	W3
id_6	Having-YELLOW	W4
id_7	Having-YELLOW	W4
id_8	Having-YELLOW	W4
id_9	Having-YELLOW	W4

Perhaps, we should stress again that attribute values have overlapping semantics. The constraints among these words have to be properly handled. So, let us examine the "interactions" among attribute values of COLOR. Two attribute values, Having-RED and Having-RED+YELLOW, obviously have overlapping semantics. We need some preparations. We need one more concept, namely, the center

$$C_w = B^{-1}(B_p), \tag{24.5}$$

where w=Name(B_p). Verbally, C_w consists of all objects that have the same B-granule B_p. We use the granule's names to index the centers:

$$\begin{aligned} C_{\text{Having-RED}} &\equiv \text{Center of } B_{id_1} = \text{Center of } B_{id_2} \\ &= \text{Center of } B_{id_3} = \{id_1, id_2, id_3\} \end{aligned}$$

$$\begin{aligned} C_{\text{Having-RED+YELLOW}} &\equiv \text{Center of } B_{id_4} = \text{Center of } B_{id_5} \\ &= \{id_4, id_5\} \end{aligned}$$

$$\begin{aligned} C_{\text{Having-YELLOW}} &\equiv \text{Center of } B_{id_6} = \text{Center of } B_{id_7} \\ &= \text{Center of } B_{id_8} = \text{Center of } B_{id_9} \\ &= \{id_6, id_7, id_8, id_9\} \end{aligned}$$

Now, we will define the binary relation B_{COLOR} in terms of BNS. First we observe that B_{COLOR} is reflexive, so we define the "other" points only. With a slight abuse of notation, we also denote B_{COLOR} by B. Let $w, u \in$ {Having-RED, Having-RED+YELLOW, Having-YELLOW}, then:

$$w \in B_u \Leftrightarrow \forall p \in C_u, B_p \cap C_w \neq \emptyset \Leftrightarrow \exists p \in C_u, B_p \cap C_w \neq \emptyset.$$

Thus, for example, we have: Having-RED+YELLOW $\in B_{\text{Having-RED}}$ since: $B_{id_1} \cap C_{\text{Having-RED+YELLOW}} \neq \emptyset$ and: $id_i \in C_{\text{Having-RED}}$. Analogously, we have: Having-RED $\in B_{\text{Having-RED+YELLOW}}$ etc.

Thus we have defined all B-granules. These B-granules defines a binary relation on the COLOR column, which is displayed in Table 24.3

Table 24.3. A Binary Relation on COLOR

Having-RED	Having-RED
Having-RED	Having-RED+YELLOW
Having-RED+YELLOW	Having-RED
Having-RED+YELLOW	Having-RED+YELLOW
Having-RED+YELLOW	Having-YELLOW
Having-YELLOW	Having-RED+YELLOW
Having-YELLOW	Having-YELLOW

Note that such a binary structure cannot be deduced from the table structure. We are ready to introduce the notion of semantic property.

DEFINITION 5 *A property is said to be semantics if and only if it is not implied by the table structure. A property is said to be syntactic if and only if it is implied by the table structure.*

The binary relation (Table 24.3) is not derived from the table structure (of Table 24.2) so it is a semantic property. This type of tables has been studied in (Lin, 1988; Lin, 1989a) for approximate retrievals; and is called topological relations or tables. Formally,

DEFINITION 6 *A table (e.g. Table 24.2) whose attributes are equipped with binary relations (e.g. Table 24.3 for COLOR attribute) is called a topological relation.*

Table 24.4. Topological Table

BALLs	Granulation 1	Granulation 2
id_1	$C_{\text{Having-RED}}$	W1
id_2	$C_{\text{Having-RED}}$	W1
id_3	$C_{\text{Having-RED}}$	W2
id_4	$C_{\text{Having-RED+YELLOW}}$	W3
id_5	$C_{\text{Having-RED+YELLOW}}$	W3
id_6	$C_{\text{Having-YELLOW}}$	W4
id_7	$C_{\text{Having-YELLOW}}$	W4
id_8	$C_{\text{Having-YELLOW}}$	W4
id_9	$C_{\text{Having-YELLOW}}$	W4

Table 24.5. A Binary Relation on the Centers of COLOR

$C_{\text{Having-RED}}$	$C_{\text{Having-RED}}$
$C_{\text{Having-RED}}$	$C_{\text{Having-RED+YELLOW}}$
$C_{\text{Having-RED+YELLOW}}$	$C_{\text{Having-RED}}$
$C_{\text{Having-RED+YELLOW}}$	$C_{\text{Having-RED+YELLOW}}$
$C_{\text{Having-RED+YELLOW}}$	$C_{\text{Having-YELLOW}}$
$C_{\text{Having-YELLOW}}$	$C_{\text{Having-RED+YELLOW}}$
$C_{\text{Having-YELLOW}}$	$C_{\text{Having-YELLOW}}$

6.3 New representations of topological relations

In (Lin, 2000), the granular table is transformed into topological information table. Here we will give a hew view and a refinement. By replacing the name of binary granule with centers in Table 24.2 and 24.3, we have Table 24.4 and Table 24.5; they are isomorphic. Table 24.5 provides the topology of Table 24.4. Table 24.4 and 24.5 provide a better interpretation than that of Table 24.2 and 24.3.

THEOREM 2 *Given a finite binary relation B, a finite equivalence relation A can be induced. The knowledge representation of B is a topological representation of A.*

7. Topological Concept Hierarchy Lattices/Trees

We will examine a nested sequence of binary granulations; the essential ideas is in (Lin, 1998b; Lin, 2000). Each inner layer is strongly dependent on the immediate next outer layer (Section 8.2).

7.1 Granular Lattice

Let us continue on the same example: Each ball in U has a B-granule. Balls 1, 2, 3 have the same B-granule; it is labeled H-Red (abbreviation of Having-Red). Similarly, Balls 4, 5 have H-Red+Yellow, and Balls 6, 7 have H-Yellow. The nested sequence (length) is display in Figure 24.1 as a tree:

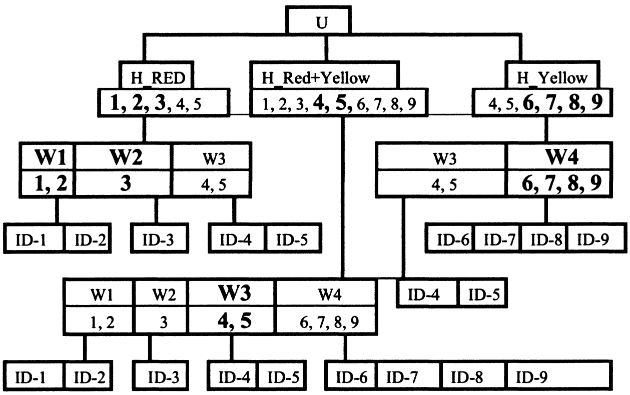

Figure 24.1. In 2nd layer the bold print letters are in the centers.

The first generation children:

1. U is granulated into three distinct children; they are named Having-Red Having-Red+Yellow, Having-Yellow; they are abbreviated to H-Red, H-Red+Yellow, and H-Yellow.

2. The three children are distinct, but *not* independent; their meanings have overlapping. Namely (1) there are interaction between H-Red+Yellow and H-Red+Yellow; (2) between H-Red+Yellow and H-Yellow; (3) there are NO interactions between H-Red and H-Yellow; The interactions are recorded in Table 24.3. This explains how the first level children are produced.

3. Every child has a center: the centers are $C_{\text{H-RED}}$ (abbreviation of $C_{\text{Having-RED}}$), $C_{\text{H-RED+Yellow}}$, $C_{\text{H-Yellow}}$. Centers are pairwise disjoint; they forms a partition.

The second generation children: Since COLOR-granulation strongly depends on WEIGHT-granulation, each COLOR-granule is a union of WEIGHT-granules. Thus one can regard that these WEIGHT-granules forms a granulation of this COLOR granule, so

1. H-Red (a COLOR-granule) is granulated into WEIGHT-granules, W1, W2, W3. Note that within each COLOR-granule the WEIGHT-granules are disjoint, so "granulated" is "partitioned."
2. H-Red+Yellow is granulated into W1, W2, W3, W4,
3. H-Yellow is granulated into W3, W4. This explains how the second level children are produced. We need information about the centers.
4. Since WEIGHT-granulation is a partition, the center is the same as granule.

7.1.1 Some Lattice Paths.

1. $U \rightarrow$ H-Red $\rightarrow W1 \rightarrow id_1$
2. $U \rightarrow$ H-Red $\rightarrow W1 \rightarrow id_2$
3. $U \rightarrow$ H-Red $\rightarrow W2 \rightarrow id_3$
4. $U \rightarrow$ H-Red+YELLOW $\rightarrow W1 \rightarrow id_1$. This path has the same beginning and ending with Item 1; but the two paths are distinct.
5. $U \rightarrow$ H-Red+YELLOW $\rightarrow W1 \rightarrow id_2$;compare with Item 2.
6. $U \rightarrow$ H-Red+YELLOW $\rightarrow W2 \rightarrow id_3$; compare with Item 3.
7. $U \rightarrow$ H-Red+YELLOW $\rightarrow W3 \rightarrow id_4$
8. etc

7.2 Granulated/Quotient Sets

1. The children consists of three (overlapping) subsets, H-Red, H-Red+Yellow, H-Yellow. This collection is more than a classical set; there are interactions among them; It forms a BNS-space; see Table 24.3.
2. The grand children:
   (a) Children of the first child $\{W1, W2, W3\}$ forms a classical set.

(b) Children of the second child $\{W1, W2, W3, W4\}$ forms a classical set.

(c) Children of the third child: $\{W3, W4\}$ forms a classical set.

3. Three distinct classical sets do have non-empty intersections.

Note that since WEIGHT-granulation is a partition, so the grand children under each individual child are disjoint. However, the grand children do overlap. The quotient set (of quotient set)

$$\begin{aligned} & \{\text{H-Red, H-Red+Yellow, H-Yellow}\} \\ = & \{\{\text{W1, W2, W3}\}, \{\text{W2, W3, W4}\}, \{\text{W3, W4}\}\} \\ = & \{\{\{id_1, id_2\}, \{id_3\}, \{id_4, id_5\}\}, \{\{id_3\}, \{id_4, id_5\}, \\ & \{id_6, id_7, id_8, id_9\}\}, \{\{id_4, id_5\}, \{id_6, id_7, id_8, id_9\}\}\} \end{aligned}$$

7.3 Tree of centers

In a granular lattice, children of every generation may overlap. Could we improve the situation? In deed, if we consider the centers only, then lattice becomes a tree (Figure 24.1a; observe the bold prints nodes).

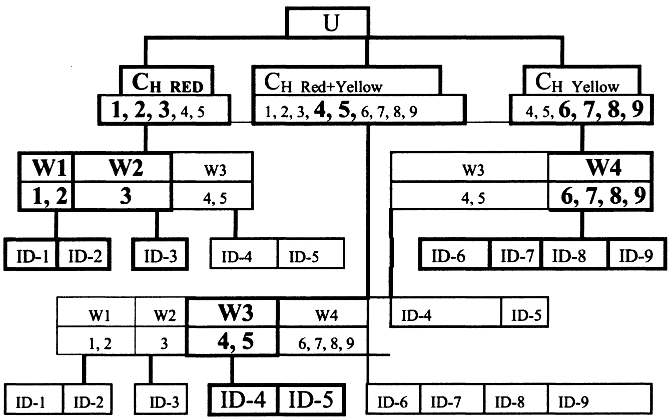

Figure 24.1 (continued). A. Bold print letters are the centers (Wi is its own center).

1. The children consists of three (non-overlapping) subsets:

(a) $C_{H-Red} = \{id_1, id_2, id_3\}$,

(b) $C_{H-Red+Yellow} = \{id_4, id_5\}$,

(c) $C_{H-Yellow} = \{id_6, id_7, id_8.id_9\}$.

They froms a classical set.

2. The grand children:

   (a) Children of the first child: $W1 = \{id_1 id_2\}$, $W2 = \{id_3\}$.

   (b) Children of the second child $W3 = \{id_4, id_5\}$.

   (c) Children of the third child: $W4 = \{id_6, id_7, id_8.id_9\}$.

3. The centers of each layers are disjoints; they forms a honest tree.

7.4 Topological tree

We will combine two trees in Figure 24.1 into one (with no information lost). We will take the tree of centers as the topological tree. Each node of the tree of centers is equipped with a B-granule (neighborhood), which is the corresponding node of the granular tree.

Here are the COLOR-neighborhoods of the centers of the first generation children:

- The neighborhood of $C_{\text{H-Red}}(= \{1, 2, 3\})$ is H-Red (= {1,2,3,4,5})
- The neighborhood of $C_{\text{H-Red+Yellow}}(= \{4, 5\})$ is H-Red+Yellow (={1,2,3,4,5,6,7, 8,9})
- The neighborhood of $C_{\text{H-Yellow}}(= \{6, 7, 8, 9\})$ is H-Yellow (={4,5,6,7,8,9})

For second generation, the WEIGHT-neighborhoods are:

- The neighborhood of $C_{\text{W1}} = W1 = \{W1, W2, W3\}$
- The neighborhood of = $C_{\text{W2}} = W2 = \{W1, W2, W3\}$
- The neighborhood of = C_{W3}= {W1, W2, W3, W4}
- The neighborhood of = C_{W4}= {W3, W4}

7.5 Table Representation of Fuzzy Binary Relations

We will use a very common example to illustrate the idea. Let the universe be $V = \{0.1, 0.2, \ldots, 0.8, 0.9\}$. It contains 9 ordinary real numbers. Each number is associated with a special fuzzy set, called a fuzzy number (Zimmerman, 1991). For example, in Figure 24.2 the numbers, 01, 02, 03, and 0.4 are respectively associated with fuzzy numbers $N1, N2, N3$ and $N4$.

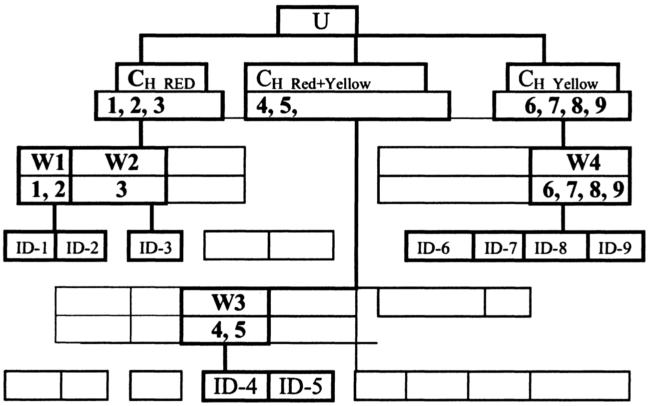

Figure 24.1 (continued). B. The tree of centers.

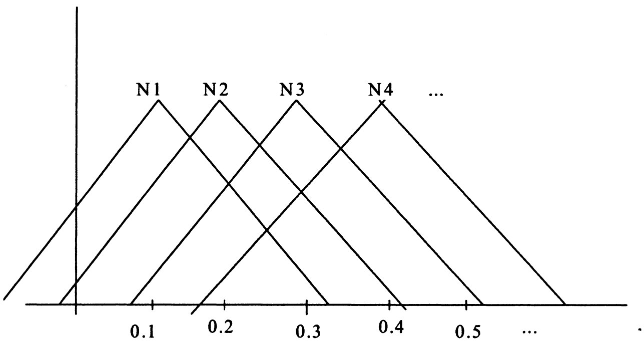

Figure 24.2. Illustration of Fuzzy Numbers Association.

8. Knowledge Processing

Pawlak (Pawlak, 1991) interprets equivalent relations as knowledge and develop a theory. In this section, we will explain how to extend his view to binary relations(Lin, 1996; Lin, 1998a; Lin, 1998b; Lin, 1999a; Lin, 1999b; Lin, 2000; Lin and Hadjimichael, 1996; Lin *et al.*, 1998). To explain these concepts, we are tempted to use the same knowledge-oriented terminology. However, our results are not completely the same; after all, binary relations are not necessarily equivalence relations. We need to distinguish the differences,

Table 24.6. Fuzzy Numbers

Points x	FB-granule	Name
0.1	$N1$	Fuzzy number 0.1=Name($N1$)
0.2	$N2$	Fuzzy number 0.2=Name($N2$)
0.3	$N3$	Fuzzy number 0.3=Name($N3$)
0.4	$N4$	Fuzzy number 0.4=Name($N4$)
...	...	...
0.9	$N9$	Fuzzy number 0.9=Name($N9$)

so mathematical terminology is used. Unless the intuitive support is needed, knowledge-oriented terms will not be employed.

8.1 The Notion of Knowledge

Pawlak views partitions (classification) as knowledge, and calls a finite set of equivalence relations on a given universe a knowledge base (Pawlak, 1991). He interprets refinements of equivalence relations as knowledge dependencies. We will take a stronger view: we regard the interpretations as the integral part of the knowledge. Here an interpretation means the naming of the mathematical structures based on real world characterization; the name is a summarization. Pawlak regards two isomorphic tables possess same knowledge (since they have the same knowledge base), however, we regard them as distinct knowledge. Let us summarize the discussions in a bullet:

- knowledge includes the knowledge representation (human interpretation) of a mathematical structure; it is a semantic notion.

For convenience, let us recall the notion of binary granular structures (Lin, 2000; Lin, 1998a; Lin, 1998b). It consists of 4-tuple

$$(V, U, B, C)$$

where V is called the object space, U the data space (V and U could be the same set), B is a set of finitely many crisp/fuzzy binary granulations, and C is the concept space which consists of all the names of B-granulations and granules. For us a piece of knowledge is a 4-tuple, while Pawlak only looks at the first three items (his definition of knowledge base).

8.2 Strong, Weak and Knowledge Dependence

Let B, P and Q be binary relations (binary granulations) for V on U (e.g. $B \subseteq V \times U$). Then we have the following:

DEFINITION 7

1. *A subset $X \subseteq U$ is B-definable, if X is a union of B-granules B_p's. If the granulation is a partition, then a B-definable subset is definable in the sense of RST.*

2. *Q is strongly dependent on P, denoted by $P \Rightarrow Q$ if and only if every Q-granule is P-definable.*

3. *Q is weakly depends on P, denoted by $P \rightarrow Q$ if and only if every Q-granule contains some P-granule.*

We will adopt the language of partition theory to granulation. For $P \Rightarrow Q$, we will say P is finer than Q or Q is coarser than P. Write $Y_p = \text{Name}(Q_p)$ and $X_{p_i} = \text{Name}(P_{p_i})$. Since $Q_p = \cup_i P_{pi}$ for suitable choices of $p_i \in V$, we write informally

$$Y_p = X_{p_1} \vee X_{p_2} \vee \cdots$$

Note that Y_p and X_{p_i} are words and $\vee$ is the "logical" disjunction. So, this is a "formula" of informal logic. Formally, we have the following proposition.

PROPOSITION 3 *If $P \Rightarrow Q$, then there is a map from the concept space of P to that of Q. The map f can be expressed by $Y_p = f(X_{p_1}, X_{p_2}, \ldots) = X_{p_1} \vee X_{p_2} \vee \cdots$; f will be termed knowledge dependence.*

This proposition is significant, since Name(P_p) is semantically interrelated. It implies that the semantic constraints among these words Name(P_p)'s are carried over to those words, Name(Q_p)'s consistently. Such semantic consistency among columns of granular tables allows us to extend the operations of classical information tables to granular tables.

8.3 Knowledge Views of Binary Granulations

DEFINITION 8

1. *Knowledge P and Q are equivalent, denoted by $P \equiv Q$, if and only if $P \Rightarrow Q$ and $Q \Rightarrow P$*

2. *The intersection of P and Q, $P \wedge Q$, is a binary relation defined by*

$$(v, u) \in P \wedge Q \ \text{ if and only if } \ (v, u) \in P \ \text{ and } \ (v, u) \in Q$$

3. *Let $C = \{C_1, C_2, \ldots, C_m\}$ and $D = \{D_1, D_2, \ldots, D_n\}$ be two collections of binary relations. We write $C \Rightarrow D$, if and only if $C_1 \wedge C_2 \wedge \cdots \wedge C_m \Rightarrow D_1 \vee D_2 \vee \cdots \vee D_n$. By mimicking ((Pawlak,*

1991), chapter 3), we write $IND(C) = C_1 \wedge C_2 \wedge \cdots \wedge C_m$; note that, all of them are binary relations, not necessarily equivalence relations.

4. *C_j is dispensable in C if $IND(C) = IND(C - \{C_j\})$; otherwise C_j is indispensable.*

5. *C is independent if each $C_j \in C$ is indispensable; otherwise C is dependent.*

6. *S is a reduct of C if S is an independent subset of C such that $IND(S) = IND(C)$.*

7. *The set of all indispensable relations in C is called a core, and denoted by $CORE(C)$.*

8. *$CORE(C) = \cap RED(C)$, where RED is the set of all reducts in C.*

COROLLARY 1 *$P \wedge Q \Rightarrow P$ and $P \wedge Q \Rightarrow Q$.*

The fundamental procedures in table processing are to find cores and reducts of decision table. We hope readers are convinced that we have developed enough notions to extend these operations to granular tables.

9. Information Integration

Many applications would want the solutions be in the same level as input data. So this section is actually quite rich. There are many theories dedicated to this portion in mathematics. For example, suppose we know a normal subgroup and the quotient group of an unknown group, there is a theory to find this unknown group. For Data Mining and part of RST, the interests are on the high level information, so this step can be skipped. For RST, approximations are the only relevant part. In this section, we focus only on the approximation theory of granulations.

9.1 Extensions

Let $Z_4 = \{[0], [1], [2], [3]\}$ be the set of integers mod 4 and we will consider it as a commutative group (Birkhoff and MacLane, 1977). Next we consider a subgroup $\{[0],[2]\}$ which is equivalent (isomorphic) to integer mod 2, Z_2, and its quotient group that consists of two elements, $\{[0],[2]\}$ and $\{[1],[3]\}$ and is also isomorphic to integer mod 2. The question is if we know the subgroup (subtasks) and the quotient group (quotient tasks), can we found the original universe. The answer is we have two universe, one is Z_4 and another is the Cartesian product of Z_2 by Z_2. So integration is not-trivial and is, outside of mathematics, unexplored teritory.

9.2 Approximations in Rough Set Theory (RST)

Let A be an equivalence relation on U. The pair (U, A) is called an approximation space.

1. $C(X) = \{x : A_x \cap X \neq \emptyset\}$ = Closure.
2. $I(X) = \{x : A_x \subseteq X\}$ = Interior,
3. $\overline{A}(X) = \cup\{A_x : A_x \cap X \neq \emptyset\}$ = Upper approximation.
4. $\underline{A}(X) = \cup\{A_x : A_x \subseteq X\}$ = Lower approximation.
5. $U(X) = \overline{A}(X)$ on (U, A)
6. $L(X) = \underline{A}(X)$ on (U, A)

DEFINITION 9 *The pair* $(\overline{A}(X), \underline{A}(X))$ *is called a rough set.*

We should caution the readers that this is a technical definition of rough sets given by Pawlak (Pawlak, 1991). However, rough set theoreticians often use "rough set" as any subset X in the approximation space, where $\overline{A}(X)$ and $\underline{A}(X)$ are defined.

9.3 Binary Neighborhood System Spaces

We will be interested in the case $V = U$. Let B be a granulation. We will call (U, B) a NS-space(Section 3), which is a generalization of the RST and topological spaces. A subset X of U is open if for every object $p \in X$, there is a neighborhood $B(p) \subseteq X$. A subset X is closed if its complement is open. A BNS is open if every neighborhood is open. A BNS is topological, if BNS open and (U, B) is a usual topological space (Sierpenski and Krieger, 1956). So BNS-space is a generalization of topological space. Let X be a subset of U.

$I[X] = \{p : B(p) \subseteq X\}$ = Interior

$C[X] = \{p : X \cap B(p) \neq \emptyset\}$ = Closure

These are common notions in topological space; they were introduced to rough set community in (Lin, 1992), Subsequently re-defined and studied by (Yao, 1998; Grzymala-Busse, 2004). We should point out that $C[X]$ may not be closed; the closure in the sense of topology is transfinite C operations; see the notion of derived sets below. By porting the rough set style definitions to BNS-space, we have:

- $L[X] = \cup\{B(p) : B(p) \subseteq X\}$ = Lower approximation

- $H[X] = \cup\{B(p) : X \cap B(p) \neq \emptyset\}$ = Upper approximation

For BNS-space, these two definitions make sense. In fact, $H(X)$ is the neighborhood of a subset, that was used in (Lin, 1992) for defining the quotient set. In non-partition cases, upper and lower approximations do not equal to interior and closure. For NS-spaces (multilevel granulation), $H(X)$ defines a NS of subset X. The topological meaning of $L(X)$ is not clear. But we have used it in (Lin, 1998b) to compute belief functions, if all granules(neighborhoods) have basic probability assignments.

Note that in BNS, each object p has a unique neighborhood $B(p)$. In general neighborhood system (NS), each object is associated with a set of neighborhoods. In such NS, we have:

- An object p is a limit point of a set X, if every neighborhoods of p contains a point of X other than p. The set of all limit points of X is call derived set $D[X]$.
- Note that $C[X] = X \cup D[X]$ may not be closed. Some authors (e.g. (Sierpenski and Krieger, 1956)) define the closure as X together with repeated (transfinite) derived set. For such a closure it is a closed set.

10. Conclusions

Information granulation is a natural problem solving strategy since ancient time. Partition, the idealized form, has played a central role in the history of mathematics. Pawlak rough set theory has shown that the partition is also powerful notion in computer science; see (Pawlak, 1991) and a more recent survey in (Yao, 2004). Granulation, we believe, will play a similar role in real world problems. Some of its success has been demonstrated in fuzzy systems (Zadeh, 1973). Many ideas have been explored (Lin, 1988; Lin, 1989a; Chu and Chen, 1992; Raghavan, 1995; Miyamoto, 2004; Liu, 2004; Grzymala-Busse, 2004; Wang, 2004; Yao, 2004; Yao, 2004).

There are many strong applications in database, Data Mining, and security (Lin, 2004), (Lin, 2000) (Hu, 2004). The application to security may worth mention; it is a non-partition theory. It shares some light on the difficult problem of controlling of Trojan horses.

Notes

1. This is an expansion of the article (Lin, 2005) in IEEE connections, the news letter of the IEEE Computational Intelligence Society

References

Aho, A., Hopcroft, J., and Ullman, J. (1974). *The Design and Analysis of Computer Algorithms*. Addison-Wesley.

Barr, A. and Feigenbaum, E. (1981). *The Handbook of Artificial Intelligence.* Addison-Wesley.

Birkhoff, G. and MacLane, S. (1977). *A Survey of Modern Algebra.* Macmillan.

David D. C. Brewer and Michael J. Nash: "The Chinese Wall Security Policy" IEEE Symposium on Security and Privacy, Oakland, May, 1988, pp 206-214,

Chu W. and Chen Q. (1992), Neighborhood and associative query answering, *Journal of Intelligent Information Systems*, **1**, 355-382, 1992.

Grzymala-Busse, J. W. (2004) Data with missing attribute values: Generalization of idiscernibility relation and rule induction. Transactions on Rough Sets, Lecture Notes in Computer Science Journal Subline, Springer-Verlag, vol. 1 (2004) 78-95.

Hobbs, J. (1985). Granularity. In *Proceedings of the Ninth Internation Joint Conference on Artificial Intelligence*, pages 432–435.

Hu X., Lin T.Y., Han J.,(2004) A New Rough Set Model Based on Database Systems, Journal of Fundamental Informatics, Vol. 59, Number 2,3,135-152

Lee, T. (1983). Algebraic theory of relational databases. *The Bell System Technical Journal*, 62(10):3159–3204.

Lin, T.Y. (1988). Neighborhood systems and relational database. In *Proceedings of CSC'88*, page 725.

Lin, T.Y. (1989). Neighborhood systems and approximation in database and knowledge base systems. In *Proceedings of the Fourth International Symposium on Methodologies of Intelligent Systems (Poster Session)*, pages 75–86.

Lin, T. Y. (1989), "Chinese Wall Security Policy–An Aggressive Model", Proceedings of the Fifth Aerospace Computer Security Application Conference, December 4-8, 1989, pp. 286-293.

Lin, T. Y.(1992) "Topological and Fuzzy Rough Sets," in: Decision Support by Experience - Application of the Rough Sets Theory, R. Slowinski (ed.), Kluwer Academic Publishers, 1992, 287-304.

Lin, T.Y. and Hadjimichael, M. (1996). Non-classificatory generalization in Data Mining. In *Proceedings of the 4th Workshop on Rough Sets, Fuzzy Sets, and Machine Discovery*, pages 404–412.

Lin, T.Y. (1996). A set theory for soft computing. In *Proceedings of 1996 IEEE International Conference on Fuzzy Systems*, pages 1140–1146.

Lin, T.Y. (1998a). Granular computing on binary relations i: Data Mining and neighborhood systems. In Skoworn, A. and Polkowski, L., editors, *Rough Sets In Knowledge Discovery*, pages 107–121. Physica-Verlag.

Lin, T.Y. (1998b). Rough set representations and belief functions ii. In Skoworn, A. and Polkowski, L., editors, *Rough Sets In Knowledge Discovery*, pages 121–140. Physica-Verlag.

Lin, T.Y., Zhong, N., Duong, J., and Ohsuga, S. (1998). Frameworks for mining binary relations in data. In Skoworn, A. and Polkowski, L., editors, *Rough sets and Current Trends in Computing*, LNCS 1424, pages 387–393. Springer-Verlag.

Lin, T.Y. (1999a). Data Mining: Granular computing approach. In *Methodologies for Knowledge Discovery and Data Mining: Proceedings of the 3rd Pacific-Asia Conference*, LNCS 1574, pages 24–33. Springer-Verlag.

Lin, T.Y. (1999b). Granular computing: Fuzzy logic and rough sets. In Zadeh, L. and Kacprzyk, J., editors, *Computing with Words in Information/Intelligent Systems*, pages 183–200. Physica-Verlag.

Lin, T.Y. (2000). Data Mining and machine oriented modeling: A granular computing approach. *Journal of Applied Intelligence*, 13(2):113–124.

Lin, T.Y. (2003a), "Chinese Wall Security Policy Models: Information Flows and Confining Trojan Horses." In: Data and Applications Security XVII: Status and Prospects,S. Vimercati, I. Ray & I. Ray 9eds) 2004, Kluwer Academic Publishers, 275-297 (Post conference proceedings of IFIP11.3 Working Conference on Database and Application Security, Aug 4-6, 2003, Estes Park, Co, USA

Lin, T.Y. (2003b), "Granular Computing: Structures, Representations, Applications and Future Directions." In: the Proceedings of 9th International Conference, RSFDGrC 2003, Chongqing, China, May 2003, Lecture Notes on Artificial Intelligence LNAI 2639, Springer-Verlag, 16-24.

Lin, T.Y. (2004), "A Theory of Derived Attributes and Attribute Completion," Proceedings of IEEE International Conference on Data Mining, Maebashi, Japan, Dec 9-12, 2002.

Lin, T.Y. (2005), Granular Computing - Rough Set Perspective, IEEE connections, The newsletter of the IEEE Computational Inelligence Society, Vol 2 Number 4, ISSN 1543-4281.

Liu, Q. (2004) Granular Language and Its Applications in Problem Solving, LNAI 3066,By Springer,127-132.

Miyamoto, S. (2004) Generalizations of multisets and rough approximations, International Journal of Intelligent Systems Volume 19, Issue 7, 639-652

Osborn S., Sanghu R. and Munawer Q.,"Configuring RoleBased Access Control to Enforce Mandatory and Discretionary Access Control Policies," ACM Transaction on Information and Systems Security, Vol 3, No 2, May 2002, Pages 85-106.

Pawlak, Z. (1982). Rough sets. *International Journal of Information and Computer Science*, 11(15):341–356.

Pawlak, Z. (1991). *Rough Sets–Theoretical Aspects of Reasoning about Data.* Kluwer Academic Publishers.

Raghavan, V. V.,Sever, H.,Deogun, J. S. (1995, August), Exploiting Upper Approximations in the Rough Set Model, Proceedings of the First International

Conference on Knowledge Discovery and Data Mining (KDD'95), Sponsored by AAAI in cooperation with IJCAI, Montreal, Quebec, Canada, August, 1995, pp. 69-74.

Sierpenski, W. and Krieger, C. (1956). *General Topology*. University of Toronto Press.

Szyperski, C. (2002). *Component Software: Beyond Object-Oriented Programming*. Addison-Wesley.

Wang, D. W.,Liau, C. J.,Hsu, T.-S. (2004), Medical privacy protection based on granular computing, Artificial Intelligence in Medicine, 32(2), 137-149

Yao, Y. Y.: Relational interpretations of neighborhood operators and rough set approximation operators. *Information Sciences* **111** (1998) 239–259.

Yao, Y. Y. (2004) A partition model of granular computing to appear in LNCS Transactions on Rough Sets.

Yao Y.Y. , Zhao Y., Yao J.T., Level Construction of Decision Trees in a Partition-based Framework for Classification, Proceedings of the 16th International Conference on Software Engineering and Knowledge Engineering (SEKE'04), Banff, Alberta, Canada, June 20-24, 2004, pp199-204.

Zadeh. L. A. (1973) Outline of a New Approach to the Analysis of Complex Systems and Decision Process. IEEE Trans. Syst. Man.

Zadeh, L.A. (1979). Fuzzy sets and information granularity. In Gupta, N., Ragade, R., and Yager, R., editors, *Advances in Fuzzy Set Theory and Applications*, pages 3–18. North-Holland.

Zadeh, L.A. (1996). Fuzzy logic = computing with words. *IEEE Transactions on Fuzzy Systems*, 4(2):103–111.

Zadeh, L.A. (1997). Towards a theory of fuzzy information granulation and its centrality in human reasoning and fuzzy logic. *Fuzzy Sets and Systems*, 19:111–127.

Zadeh, L.A. (1998) Some reflections on soft computing, granular computing and their roles in the conception, design and utilization of information/ intelligent systems, Soft Computing, 2, 23-25.

Zhang, B. and Zhang, L. (1992). *Theory and Applications of Problem Solving*. North-Holland.

Zimmerman, H. (1991). *Fuzzy Set Theory –and its Applications*. Kluwer Acdamic Publisher.

V

SUPPORTING METHODS