Yao, Y.Y., Granular Computing, Computer Science (Ji Suan Ji Ke Xue), Vol. 31, pp. 1-5, 2004
Proceedings of The 4th Chinese National Conference on Rough Sets and Soft Computing

Granular Computing

Y.Y. Yao
Department of Computer Science, University of Regina
Regina, Saskatchewan, Canada S4S 0A2
E-mail: yyao@cs.uregina.ca, http://www.cs.uregina.ca/~yyao

Abstract The basic ideas and principles of granular computing (GrC) have been studied explicitly or implicitly in many fields in isolation. With the recent renewed and fast growing interest, it is time to extract the commonality from a diversity of fields and to study systematically and formally the domain independent principles of granular computing in a unified model. A framework of granular computing can be established by applying its own principles. We examine such a framework from two perspectives, granular computing as structured thinking and structured problem solving. From the philosophical perspective or the conceptual level, granular computing focuses on structured thinking based on multiple levels of granularity. The implementation of such a philosophy in the application level deals with structured problem solving.

Keywords: Granularity, granule, level, hierarchy, structured thinking, structured problem solving

1. Introduction

Human problem solving involves the perception, abstraction, representation and understanding of real world problems, as well as their solutions, at different levels of granularity [4, 6, 23, 28, 32-35]. The consideration of granularity is motivated by the practical needs for simplification, clarity, low cost, approximation, and tolerance of uncertainty [32]. As an emerging field of study, granular computing attempts to formally investigate and model the family of granule-oriented problem solving methods and information processing paradigms [14, 23, 28].

Ever since the introduction of the term of "Granular computing (GrC)" by T.Y. Lin in 1997 [8, 32], we have witnessed a rapid development of and a fast growing interest in the topic [2, 5, 8-10, 13, 14, 16-20, 22-31, 33, 35, 37]. Many models and methods of granular computing have been proposed and studied. From the wide spectrum of current research, one can easily make several observations. There does not exist a general agreement about what is granular computing, nor there is a unified model [36]. Many studies concentrate on concrete models in particular contexts, and hence only capture limited aspects of granular computing. Consequently, the potential applicability and usefulness of granular computing are not well perceived and appreciated.

The studies of concrete models and methods are important for the development of a field in its early stage. It is equally important, if not more, to study a general theory that avoids constraints of a concrete model.

The basic notions and principles of granular computing, though under different names, have in fact been appeared in many related fields, such as programming, artificial intelligence, divide and conquer, interval computing, quantization, data compression, chunking, cluster analysis, rough set theory, quotient space theory, belief functions, machine learning, databases, and many others [8, 23, 28, 32, 33]. However, granular computing has not been fully explored in its own right. It is time to extract the commonality from these diverse fields and to study systematically and formally the domain independent principles of granular computing in a unified and well-formulated framework.

In this paper, we study high level and qualitative characteristics of a theory of granular computing. A general domain independent framework is presented, in which basic issues are examined.

2. Perspectives of Granular Computing

It may be difficult, if not impossible, to give a formal, precise and uncontroversial definition of granular computing. Nevertheless, one can still extract the fundamental elements from the human problem solving experiences and methods. There are basic principles, techniques and methodologies that are commonly used in most types of problem solving. Granular computing, therefore, focuses on problem solving based on the commonsense concepts of granule, granulated view, granularity, and hierarchy. They are interpreted as the abstraction, generalization, clustering, levels of abstraction, levels of detail, and so on in various domains. We view granular computing as a study of a general theory of problem solving based on different levels of granularity and detail [28].

Granular computing can be studied by applying its principles and ideas. It can be investigated in different levels or perspectives by focusing on its philosophical foundations, basic components, fundamental issues, and general principles. The philosophical level concerns structured thinking, and the application level deals with principles of structured problem solving. While structured thinking provides guidelines and leads naturally to structured problem solving, structured problem solving implements the philosophy structured thinking.

The philosophy of thinking in terms of levels of granularity, and its implementation in more concrete models, would result in disciplined procedures that help to avoid errors and to save time for solving a wide range of complex problems.

3. Basic Components of Granular Computing

In modeling granular computing, we focus on three basic components and their interactions.

3.1. Granules

A granule may be interpreted as one of the numerous small particles forming a larger unit. Collectively, they provide a representation of the unit with respect to a particular level of granularity. That is, a granule may be considered as a localized view or a specific aspect of a large unit.

Granules are regarded as the primitive notion of granular computing. Its physical meanings become clearer when dealing with more concrete models. For example, in set-theoretic setting, such as rough sets, quotient space theory and cluster analysis, a granule may be interpreted as a subset of a universal set [12, 13, 34, 35]. In planning, a granule can be a sub-plan [6]. In programming, a granule can be a program module [7]. For the conceptual formulation of granular computing, we do not attempt to interpret the notion of granules based on more intuitive, but rather restrictive, concepts. We focus on some fundamental issues based on this weak view of granules.

The size of a granule is considered as a basic property. Intuitively, the size may be interpreted as the degree of abstraction, concreteness, or detail. In the set-theoretic setting, the size of a granule can be the cardinality of the granule.

Connections and relationship between granules can be represented by binary relations. In concrete models, they may be interpreted as dependency, closeness, or overlapping. For example, based on the notion of size, one can define an order relation on granules. Depending on the particular context, the relation may be interpreted as “greater than or equal to”, “more abstract than”, or “coarser than”. The order relation may be reflexive and transitive, but not symmetric. The order relation is particularly useful in studying connections between granules in different levels.

One can define operations on granules so that one can operate on granules, such as combining many granules to form a new granule or decomposing a granule into many granules. The operations on granules must be consistent with the binary relations on the granules. For example, the combined granule should be more abstract than its components. The sizes of granules, the relations between granules, and the operations on granules provide the essential ingredients for developing a theory of granular computing.

3.2. Granulated views and levels

In his work on vision, Marr convincingly made the point that a full understanding of an information processing system involves explanations at various levels [11]. The three levels considered are the computational, algorithmic, and implementational. The computational level describes the information processing problem to be solved by the system. The algorithmic level describes the steps that need to be carried out to solve the problem. The implementational level deals with physical realization of the system. Although there does exist a general agreement on the interpretations and the exact number of levels, it is commonly accepted that the notion of levels is an important one in computer science [3].

Foster critically reviewed and systematically compared various definitions and interpretations of the notion of levels [3]. Three basic issues, namely, definition of levels, number of levels, and relationship between levels, are clarified. Levels are considered simply as descriptions or points of views and often for the purpose of explanation. The number of levels is not fixed, but depends on the context and the purpose of description or explanation. A multi-layered theory of levels captures two senses of abstraction. One is the abstraction in terms of concreteness and is represented by planes along the dimension from top to bottom. The other is the abstraction in terms of the amount of detail and can be modeled along another dimension from less detail to more detail on the same plane.

By viewing a level as a description or a point of view, one can immediately apply it as a basic notion to model granular computing. In order to emphasize the context of granular computing, we also refer to a level as a granulated view. A level consists of entities called granules whose properties characterize and describe the subject matters of study, such as a real world problem, a theory, a design, a plan, a program, or an information processing system. Granules are formed with respect to a particular degree of granularity or detail. Granules in a level are defined and formed within a particular context and are related to granules in other levels.

There are two types of information and knowledge encoded in a level. A granule captures a particular aspect, and collectively, all granules in the level provide a granulated view. The granularity of a level refers to the collective properties of granules in a level with respect to their sizes. The granularity is reflected by the sizes of all granules involved.

3.3. Hierarchies

Granules in different levels are linked by the order relations and operations on granules. The order relation on granules can be extended to granulated views (levels). A level is above another level if each granule in the former level is ordered before a granule in the latter level, and each granule in the latter level is ordered after a granule in the former level, under the order relation. The ordering of levels can be described by the notion of hierarchy.

The theory of hierarchy provides a multi-layered framework based on levels. Mathematically, a hierarchy may be viewed as a partially ordered set [1]. For the study of granular computing, the elements of the ordered set are interpreted as hierarchical levels or granulated views. The ordering of levels in a hierarchy is based on criteria that are related to the order relations on granules. A higher level may provide a constraint to and/or context of a lower level, and may contain and be made of lower levels. Depending on the context, a hierarchy may consist of levels of interpretation, levels of abstraction, levels of organization, levels of observation, and levels of detail. A hierarchy represents relationships between different granulated views, and explicitly shows the structure of granulation.

A granule in a higher level can be decomposed into many granules in a lower level, and conversely many granules in a lower level can be combined into one granule in a higher level. A granule in a lower level may be a more detailed description of a granule in a higher level with added information. In the other direction, a granule in a higher level is a coarse-grained description of a granule in a lower level by omitting irrelevant details.

3.4. Granular structures

With the introduction of the three components, one can examine three types of structures for modeling their interactions. They are the internal structure of a granule, the collective structure of the all granules (i.e., the internal structure of a granulated view or level), and the overall structure of all levels.

Although a granule is normally considered as a whole instead of many sub-granules at a given level, its internal structure needs to be examined. The internal structure of a granule provides a proper description, interpretation, and characterization of the granule. A granule may have a complex structure itself. For examples, the internal structure of a granule may be a hierarchy consisting of many levels. The internal structure is also useful in establishing linkage among granules in different levels.

All granules in a level may collectively show a certain structure. This is the internal structure of a granulated view. Granules in a level, although may be relatively independent, are somehow related to a certain degree. This stems from the fact that they together form a granulated view. On the other hand, it is expected that in many situations the relationships between different granules are much weaker. The internal structure of a level is only meaningful if all the granules in the level are considered together.

A hierarchy represents the overall structure of all levels. In a hierarchy, both the internal structure of granule and the internal structure of granulated views are reflected, to some degree, by the order relations. In a hierarchy, not any two granulated views can be compared based on the order relation. In the special case, the hierarchy is a tree.

The three structures as a whole is referred to as the granular structure. One can establish more connections between three structures. For example, granules in a higher level may have greater integrity and higher bond strength than those in a lower level. The structures need to be fully explored to establish a basis of granular computing.

3.5. A partition model

The three basic components of granular computing can be easily illustrated by a concrete model known as the partition model of granular computing [28], which is based on rough set theory [12, 13] and quotient space theory [34, 35].

A central notion of the partition model is equivalence relations. In rough set theory, an equivalence relation on a set of objects can be concretely defined in an information table based on their values on a finite set of attributes [12, 31]. Two objects are equivalent if they have exact the same values on a set of attributes.

An equivalence relation divides a universal set into a family of pair-wise disjoint subsets, called the partition of the universe. A granule of a partition model is therefore an equivalence class defined by an equivalence relation. The internal structure of an equivalence class is captured by the same values of some attributes. A granulated view is the partition induced by an equivalence relation, and its structure is defined by the properties of the partition. Different equivalence relations can be ordered based on set inclusion, which leads to a hierarchy of partitions. In an information table, we only consider partitions generated by different subsets of attributes. The overall hierarchical structure is therefore induced by subsets of attributes.

The partition model may be viewed as a special case of cluster analysis. Following the same argument, one can easily find the correspondence between basic components of granular computing and its structures in cluster analysis. In general, given any concrete model of granular computing, we can easily find the corresponding components and structures.

4. Basic Issues of Granular Computing

The discussions of this section summarize and extend the preliminary results reported in [23, 28]. The list of issues discussed should not be viewed as a complete one. It can only be viewed as a set of representatives. Based on the principles of granular computing, these issues may also be studied at different levels of detail.

Granular computing may be studied based on two related issues, i.e., granulation and computation [23, 28]. The former deals with the construction, interpretation, and representation of the three basic components, and the latter deals with the computing and reasoning with granules and granular structures.

Studies of granular computing cover two perspectives, namely, the algorithmic and the semantic [23, 28]. Algorithmic study concerns the procedures for constructing granules and related computation, and the semantic study concerns the interpretation and physical meaningfulness of various algorithms. Studies from both aspects are necessary and important. The results from semantic study may provide not only interpretations and justifications for a particular granular computing model, but also guidelines that prevent possible misuses of the model. The results from algorithmic study may lead to efficient and effective granular computing methods and tools.

4.1. Granulation

Granulation involves the construction of the three basic components, granules, granulated views and hierarchies. Two basic operations are the top-down decomposition of large granules to smaller granules, or the bottom-up combination of smaller granules into larger granules.

The notion of granulation can be studied in many different contexts. The granulation of a problem, a theory, or a universe, particularly the semantics of granulation, is domain and application dependent. Nevertheless, one can still identify some domain independent issues. For clarity, some of these issues are discussed in the set-theoretic setting.

In the set-theoretic setting, a granule may be viewed as a subset of the universe, which may be either fuzzy or crisp. A family of granules containing every object in the universe is called a granulated view of the universe. A granulated view may consist of a family of either disjoint or overlapping granules. There are many granulated views of the same universe. Different views of the universe can be linked together, and a hierarchy of granulated views can be established.

Granulation criteria. A granulation criterion deals with the semantic issues and addresses the question of why two objects are put into the same granule. It is domain specific and relies on the available knowledge. In many situations, objects are usually grouped together based on their relationships, such as indistinguishability, similarity, proximity, or functionality [32]. One needs to build models to provide both semantical and operational interpretations of these notions. They enable us to formally and precisely define various notions involved, and to systematically study the meanings and rationale of a granulation criterion.

Granulation methods. From the algorithmic aspect, a granulation method addresses the problem of how to put two objects into the same granule. It is necessary to develop algorithms for constructing granules and granulated views efficiently based on a granulation criterion.

Representation/description. The next issue is the interpretation of the results of a granulation method, i.e., the granular structures. Once constructed, it is necessary to describe, to name and to label granules using certain languages. One may assign a name to a granule such that an element in the granule is an instance of the named category. One may also provide a formal description of objects in the same granule. By pooling the representations of granules, one can obtain the overall representation of a granulated view.

Qualitative and quantitative characterization. One can associate quantitative measures to the three components, granules, granulated views, and hierarchies. The measures should reflect and be consistent with the three structures, the internal structure of a granule, the collective structure of a granulated view, and the overall structure of a hierarchy.

4.2. Computing with granules

Computing and reasoning with granules explore the three types of structures. They can be similarly studied from both the semantic and algorithmic perspectives. One needs to design and interpret various methods based on the interpretation of granules and relationships between granules, as well as to define and interpret operations of granular computing.

Mappings. The connections between different levels of granulations can be described by mappings. At each level of the hierarchy, a problem is represented with respect to the granularity of the level. The mapping links different representations of the same problem at different levels of detail. In general, one can classify and study different types of granulations by focusing on the properties of the mappings.

Granularity conversion. A basic task of granular computing is to change views with respect to different levels of granularity. As we move from one level of detail to another, we need to convert the representation of a problem accordingly. A move to a more detailed view may reveal information that otherwise cannot be seen, and a move to a simpler view can improve the high level understanding by omitting irrelevant details of the problem.

Operators. Operators can precisely define the conversion of granularity in different levels. They serve as the basic building blocks of granular computing. There are at least two types of operators that can be defined. One type deals with the shift from a fine granularity to a coarse granularity. A characteristic of such an operator is that it will discard certain details, which makes distinct objects no longer differentiable. Depending on the context, many interpretations and definitions are available, such as abstraction, simplification, generalization, coarsening, zooming-out, and so on. The other type deals with the change from a coarse granularity to a fine granularity. A characteristic of such an operator is that it will provide more details, so that a group of objects can be further classified. They can be defined and interpreted differently, such as articulation, specification, expanding, refining, zooming-in, and so on.

Property preservation. Granulation allows different representations of the same problem in different levels of detail. It is naturally expected that the same problem must be consistently represented. Granulation and its related computing methods are meaningful only if they preserve certain desired properties. For example, Zhang and Zhang studied the "false-preserving" property, which states that if a coarse-grained space has no solution for a problem then the original fine-grained space has no solution [34, 35]. Such a property can be explored to improve the efficiency of problem solving by eliminating a more detailed study in a coarse-grained space. One may require that the structure of a solution in a coarse-grained space is similar to the solution in a fine-grained space. Such a property is used in top-down problem solving techniques. More specifically, one starts with a sketched solution and successively refines it into a full solution. In the context of hierarchical planning, one may impose similar properties, such as upward solution property, downward solution property, monotonicity, etc. [6].

4.3. The rough set model

As an illustration, we discuss the basic issues of granular computing based on the results from the rough set theory. Many applications of the rough set theory are based on the exploration of those issues.

Granulation. The granulation criterion is an equivalence relation on a set of objects, which is concretely defined in an information table based on the values of a set of attributes. The granulation method is simply the collection of equivalent objects. One associates a formula to each equivalence class, which provides a formal description of the equivalence class. One also associates quantitative measures to equivalence classes and the partition induced by the equivalence relation.

Computing with granules. Many of the applications of rough set theory can be viewed as concrete examples of computing with granules. With respect to an information table, mappings between different granulated views are in fact defined by different subsets of attributes. The conversion of granularity is achieved by adding or deleting attributes. The rough set approximation operators are granularity conversion operators.

An important application of rough set theory is to learn classification rules [12, 21]. One of the important steps is to find a reduct of attributes, i.e., a set of individually necessary and collectively sufficient attributes that provide the correct classification [12, 21]. Conceptually, this can be easily modeled as searching the partition hierarchy defined by all subsets of attributes. Even in this simple search process, we have to deal with the issues discussed earlier. The mappings between levels direct the search direction; granularity conversion and property preserving principles govern the quality of the searched granulated views, the operators can be used to define the quality of each decision rule.

5. Conclusion

By explicitly introducing an umbrella term of granular computing, one can explore, organize and unify the divergent concepts, theories, and applications into a well-formulated and unified theory of problem solving. It is time to move from studies of particular methods and concrete models of granular computing to a more abstract level. One needs to study its basic philosophy and principles, and to build a more general framework. This paper may be viewed as a step toward this goal.

Although this paper does not cover all aspects of a complete model of granular computing, the results are useful in building a concrete model in which one can examine specific techniques and issues of granular computing in the context of particular applications.

The notions of granules, granulated views (levels) and hierarchies are sufficient for us to discuss the basic issues of granular computing. The sizes of granules, the granular structures, and the operations on granules provide the essential ingredients for the development of a theory of granular computing.

References

1. Ahl, V. and Allen, T.F.H. (1996) *Hierarchy Theory, a Vision, Vocabulary and Epistemology*, Columbia University Press.
2. Bargiela, A. and Pedrycz W. (2002) *Granular Computing: an Introduction,* Kluwer Academic Publishers, Boston.
3. Foster, C.L. (1992) *Algorithms, Abstraction and Implementation: Levels of Detail in Cognitive Science*, Academic Press, London.
4. Hobbs, J.R. (1985) Granularity, *Proceedings of the 9th International Joint Conference on Artificial Intelligence*, 432-435.
5. Inuiguchi, M., Hirano, S. and Tsumoto, S. (Eds.) (2003) *Rough Set Theory and Granular Computing,* Springer, Berlin.
6. Knoblock, C.A. (1993) *Generating Abstraction Hierarchies: an Automated Approach to Reducing Search in Planning*, Kluwer Academic Publishers, Boston.
7. Ledgard, H.F., Gueras, J.F. and Nagin, P.A. (1979) *PASCAL with Style: Programming Proverbs*, Hayden Book Company, Inc., Rechelle Park, New Jersey.
8. Lin, T.Y. (1997) Granular computing, announcement of the BISC Special Interest Group on Granular Computing.
9. Lin, T.Y. (2003) Granular computing, LNCS 2639, Springer, Berlin, 16-24.
10. Lin, T.Y., Yao, Y.Y. and Zadeh, L.A. (Eds.) (2002) *Data Mining, Rough Sets and Granular Computing*, Physica-Verlag, Heidelberg.
11. Marr, D. (1982) *Vision: A Computational Investigation into the Human Representation and Processing of Visual Information*, W.H. Freeman and Company, New York.
12. Pawlak, Z. (1991) Rough Sets, *Theoretical Aspects of Reasoning about Data*, Kluwer Academic Publishers, Dordrecht.
13. Pawlak, Z. (1998) Granularity of knowledge, indiscernibility and rough sets, *Proceedings of 1998 IEEE International Conference on Fuzzy Systems*, 106-110.
14. Pedrycz, W. (Ed.), 2001, *Granular Computing: an Emerging Paradigm*, Physica-Verlag, Heidelberg.
15. Peikoff, L. (1981) *Objectivism: the Philosophy of Ayn Rand*, Dutton, New York.
16. Peters, J.F., Pawlak, Z. and Skowron, A. (2002) A rough set approach to measuring information granules, *Proceedings of COMPSAC 2002*, 1135-1139.
17. Polkowski, L. and Skowron, A. (1998) Towards adaptive calculus of granules, *Proceedings of 1998 IEEE International Conference on Fuzzy Systems*, 111-116.
18. Skowron, A. (2001) Toward intelligent systems: calculi of information granules, *Bulletin of International Rough Set Society*, **5**, 9-30.
19. Skowron, A. and Stepaniuk, J. (2001) Information granules: towards foundations of granular computing, *International Journal of Intelligent Systems*, **16**, 57-85.
20. Wang, G., Liu, Q., Yao, Y.Y. and Skowron, A. (Eds.) (2003) *Rough Sets, Fuzzy Sets, Data Mining, and Granular Computing*, LNCS 2639, Springer, Berlin.
21. Wang, J. (2002), Rough sets and their applications in data mining, in: *Fuzzy Logic and Soft Computing*, Chen, G., Ying, M. and Cai, K.-Y. (Eds.), Kluwer Academic Publishers, Boston.
22. Yao, Y.Y., (1999) Granular computing using neighborhood systems, in: *Advances in Soft Computing: Engineering Design and Manufacturing,* Roy, R., Furuhashi, T., and Chawdhry, P.K. (Eds), Springer-Verlag, London, 539-553.
23. Yao, Y.Y. (2000) Granular computing: basic issues and possible solutions, *Proceedings of the 5th Joint Conference on Information Sciences*, 186-189.
24. Yao, Y.Y. (2001) Information granulation and rough set approximation, *International Journal of Intelligent Systems*, **16**, 87-104.
25. Yao, Y.Y. (2001) Modeling data mining with granular computing, *Proceedings of COMPSAC 2001*, 638-643.
26. Yao, Y.Y. (2003) Information granulation and approximation in a decision-theoretical model of rough sets, in: *Rough-Neural Computing: Techniques for Computing with Words*, Pal, S.K., Polkowski, L., and Skowron, A. (Eds), Springer, Berlin, 491-518.
27. Yao, Y.Y. (2003) Granular computing for the design of information retrieval support systems, in: *Information Retrieval and Clustering*, Wu, W., Xiong, H. and Shekhar, S. (Eds.), Kluwer Academic Publishers 299.
28. Yao, Y.Y. (2004) A partition model of granular computing, *LNCS Transactions on Rough Sets,* to appear.
29. Yao, Y.Y. and Liau, C.-J. (2002) A generalized decision logic language for granular computing, *Proceedings of FUZZ-IEEE'02*, 1092-1097.
30. Yao, Y.Y., Liau, C.-J. and Zhong, N. (2003) Granular computing based on rough sets, quotient space theory, and belief functions, *Proceedings of ISMIS'03*, 152-159.
31. Yao, Y.Y. and Zhong, N. (2002) Granular computing using information tables, in: *Data Mining, Rough Sets and Granular Computing*, Lin, T.Y., Yao, Y.Y. and Zadeh, L.A. (Eds.), Physica-Verlag, Heidelberg, 102-124.
32. 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.
33. 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.
34. Zhang, B. and Zhang, L. (1992) *Theory and Applications of Problem Solving*, North-Holland, Amsterdam.
35. Zhang, L. and Zhang, B. (2003) The quotient space theory of problem solving, LNCS 2639, 11-15.
36. Zhao, M. (2004) *Data Description based on Ruduct Theory*, Ph.D. Dissertation, Institute of Automation, Chinese Academy of Sciences.
37. Zhong, N., Skowron, A. and Ohsuga S. (Eds.) (1999) *New Directions in Rough Sets, Data Mining, and Granular-Soft Computing*, LNAI 1711, Springer,

Berlin.