



## Ontology design with a granular approach



Yong Liu<sup>a</sup>, Xiaoling Zheng<sup>b,c,\*</sup>, Feng Tang<sup>d</sup>, Xiaofei Chen<sup>e</sup>

<sup>a</sup> Department of Control Science and Engineering, Zhejiang University, Hangzhou 310027, China

<sup>b</sup> College of Computer Science, Zhejiang University, Hangzhou 310027, China

<sup>c</sup> Stanford University, Stanford, CA 94305, USA

<sup>d</sup> Hewlett-Packard Labs, 94304 Palo Alto, CA, USA

<sup>e</sup> School of Finance and Economics, Zhejiang University of Finance & Economics, Dongfang College Jiaxing, Hangzhou 314408, China

### ARTICLE INFO

#### Keywords:

Ontology design  
Granular structure  
Approximation  
Inconsistency reduction  
Optimization subset

### ABSTRACT

Ontology design for complex applications is quite a challenge. The quality of an ontology is highly dependent upon the capabilities of designers, and the collaborative design process is hampered by the difficulty of balancing the viewpoints of different designers. In this paper, we present a granular view of ontology: ontologies are granular, ontologies are granular approximations of conceptualizations and conceptualization granules of an ontology are ordered tuples. We then propose a corresponding granular ontology design approach. In our granular ontology design approach, the unified granular cognition level and hierarchies of sub-concepts are initialized before ontological terms are designed in detail, which reduces the subjective effects of the capabilities of designers. Our approach also introduces the idea of optimization to choose an optimal subset, which can best approximate the real concept domain, from the knowledge rule set presented by different domain experts. The optimal subset is chosen on the basis of the principle of granular ontology knowledge structure.

© 2014 Elsevier Ltd. All rights reserved.

### 1. Introduction

Ontology design is an important and essential technique in the creation of knowledge-based applications (Bittner & Smith, 2003; Guarino & Welty, 2002; Liu, Xu, Zhang, & Pan, 2008). Ontology design has been successfully applied in many areas, such as decomposition of information systems (Wand & Weber, 1990), model checking and semantic reasoning (DiPietro, Pagliarecci, & Spalazzi, 2012), and inconsistent detection in complex scene modeling (Liu, Zhang, Jiang, & Zhao, 2012a, 2012b). However, ontology design is still a great challenge for many knowledge-based applications, especially when many complex concepts with fuzzy overlap are involved with the target objects.

Generally speaking, there are two challenges in the designing of an ontology. The first one is the subjective bias in ontology design. It is well known that different designers produce different ontologies for the same target concepts, and the qualities of those ontologies rely greatly on the subjective cognition levels of designers. We call the effect caused by the capabilities of designers *subjective bias*. Currently, few works focus on removing or even reducing subjective bias via objective ontology design approaches. The second

challenge is how to enable experts to design an ontology collaboratively. In many cases of collaborative design, the crucial problem becomes how to balance the viewpoints of different experts. Thus a general ontology approach that can highlight experts' conflicts intuitively will simplify the ontology design greatly.

In this paper, we present three granular viewpoints on ontology design: that an ontology is granular, that an ontology is a granular approximation of a conceptualization, and that the conceptual relationship between granules of an ontology are ordered tuples. Based on the three basic granular viewpoints of ontology, we focus on a general design approach for ontology. Our approach can help address the vagueness, fuzzy and overlapped concepts, and potential need for collaboration between different domain experts that make ontology design a challenge.

In our granular ontology design approach, the unified granular cognition level and hierarchies of sub-concepts are initialized before the detailed designing of ontology terms, which reduces the subjective bias in ontology design. Our approach also introduces the idea of optimization to choose an optimal subset, which can best approximate the real concept domain, from the knowledge rule set presented by different domain experts. We present our approach and demonstrate it with an ontology design process for ancient Chinese architecture (Liu, Jiang, & Huang, 2010; Liu et al., 2012a, 2012b), which contains complex sub-concepts and

\* Corresponding author at: Zhejiang University, Hangzhou 310027, China. Tel.: +86 571 87951453.

E-mail addresses: [xlzheng@zju.edu.cn](mailto:xlzheng@zju.edu.cn), [zhengxl@stanford.edu](mailto:zhengxl@stanford.edu) (X. Zheng).

also requires the design input of multiple experts working collaboratively.

## 2. General ontology design process

The use of “ontology” in a design context was originally introduced by Gruber (Gruber, 1993), who described ontology as an explicit specification of a conceptualization—an abstract, simplified view of the “things” in a designer’s viewpoints (Sure Staab et al., 2004). A general formal definition of an ontology is a quads  $C = \langle D, W, \mathfrak{R}, V \rangle$ , where  $D$  is the conceptual domain;  $V$  is a set of related entities or sub-concepts involved in the ontology conceptualization  $C$ , and  $W$  is a set of the conceptual instances. The ontology contains possible states of affairs that correspond to mutual arrangements of the above entities, and  $\mathfrak{R}$  is a set of conceptual relations (also called a knowledge set). The conceptual relations are established between the entities and a specific domain’s instance in  $W$ , the conceptual relations may be referred as a  $n$ -ary function  $\mathfrak{R}^n : W \rightarrow 2^{V^n}$ . For example, in our architecture modeling case (Liu et al., 2008), the concept of southeast ancient Chinese architecture  $C$  may include four sets:

- (1) the hierarchical domain structure set  $D$  for the target concept;
- (2) the entity set  $V$ , which contains the basic architecture components, such as gate, window, and roof;
- (3) the instance set  $W$ , which contains all the possible instances of the southeast ancient Chinese architecture domain, each instance of which is formed by the components in  $V$ , an example is shown in Fig. 1;
- (4) the knowledge set  $\mathfrak{R}$ , which contains all the “correct” combination and topology relations of the basic components.

Thus the design of an ontology can be summarized as the process by which a group of experts clarifies a set of entities  $V$  and conceptual relations  $\mathfrak{R}$  with respect to a conceptualization  $C$ . An obvious way to clarify the set of conceptual relations is to enumerate all the mappings between the set  $W$  and  $V$ ; however, this is impossible when the  $W$  is infinite, so designers may introduce a rule system  $R^1$  based on first-order logic (FOL) to represent how the basic components  $V$  can constitute the instances in  $W$ . A typical example is the grammar used in procedural modeling of architecture (Liu et al., 2008; Müller, Wonka, Haegler, Ulmer, & Van Gool, 2006), for which grammar rules<sup>2</sup> such as the following present the combination sequence of each component in  $V$ :

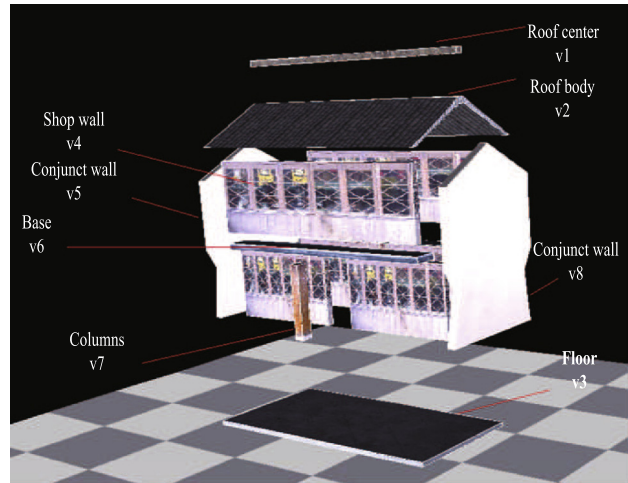
$$r_1(\text{roof} ::= \text{roof\_center}|\text{roof\_body})$$

$$r_2(\text{window\_wall} ::= \text{window\_wall}|\text{shop\_wall}|\text{column})$$

$$r_3(\text{window\_wall} ::= \text{shop\_wall}|\text{base}|\text{shop\_wall}|\text{column})$$

$$r_4(\text{house} ::= \text{house}|\text{roof}|\text{window\_wall}|\text{shop\_wall}|\text{window\_wall}|\text{shop\_wall})$$

Here the terms on the left are internodes and the terms on the right that do not appear on the left are terminal-nodes. An ontology such as this one gives a machine to generate house instances by replacing the internodes with the right parts according to the



**Fig. 1.** An example of the architecture instance, formed by many basic components,  $v_1, \dots, v_8$ , the instance can be represented as  $w = [[v_1]T v_2]T[[v_5]L[[v_4]T v_6]T[v_4 v_7 v_4]D]F[v_8]R[[v_4]T v_6]T[v_4 v_7 v_4]D]B]D$ . Here  $[v_1]T v_2$  means  $v_1$  locates top of  $v_2$ , similarly,  $D$  refers down of,  $L$  refers left of,  $R$  refers right of,  $F$  refers front of,  $B$  refers back of, then the house can be regarded as a sequence of the basic components.

corresponding rules once or multiple times. For example, with the rules above, a combination sequence for an instance of the house might be:

$$\text{house} = \text{roof\_center}, \text{roof\_body}, \text{shop\_wall}, \text{column}, \text{shop\_wall}, \\ \text{shop\_wall}, \text{column}, \text{shop\_wall}, \text{column}, \text{shop\_wall}$$

Unfortunately, the refinement of knowledge rules from complex phenomena is a challenge, especially when the conceptual relations of those “things” that need to be conceptualized are hard to describe in a way that is understandable to humans.

Theoretically, the conceptual relations in an ontology should be complete, correct, clear and concise. However, ontologies are created by humans and bias is inevitably introduced, especially when experts from different domains working on the ontology. The quality of an ontology relies greatly on the experience and skills of the designers, yet to the best of our knowledge, there is not yet a stable data model or objective design pattern for creating an ontology under complex conditions.

In the following article, we present a novel granular ontology design approach, which is based on our three granular views of ontology. In our granular approach, we try to establish a general ontology design framework that is accurate, collaborative, efficient, objective (or at least less subjective than standard ontologies), and appreciable.

## 3. Granular views on ontology

Our granular views are based on the fact that knowledge tends to be vague and the associated data is often incomplete when trying to find new sub-concepts based on data linked to an ontology (Keet, 2010a, 2010b). Our approach employs a rough methodology that considers the interior, exterior and boundaries of the knowledge in an ontology and is similar to the approach of Calegari (Calegari & Ciucci, 2010). According to our methodology, we construct a specific granular view of an ontology.

### 3.1. Ontologies are granular

In information science, an ontology can be regarded as an artifact projection (or representation) of a real-world concept based on

<sup>1</sup> The rules are also referred to as the knowledge set  $R$  in the ontology, although they are an approximation of  $\mathfrak{R}$  in engineering practice.

<sup>2</sup> In a real case, there would also be spatial control terms similar to Fig. 1 among the components.

the designer's cognition. An ontology is a reflection of real conceptualizations, which are naturally hierarchically. As the usual cognitive pattern for a human is “divide and conquer”, the same conceptualization may be represented in different subjective sub-concepts (partitions or granules) due to different designers or different applied fields. For instance, in the ontology for digital 3D architecture models, artist designers might express their ontology entity set  $V$  with several colorful geometries such as point, line, and polygon. Architects might present another  $V'$  with architecture components in semantics such as window, roof, and gate. For the architects, the basic units in  $V$  or  $V'$  will draw together via the objective restriction to form the concept of 3D architecture models. By means of the different partitions of  $V$  and  $V'$ , the ontology is granular, and each basic element in the entity set is a granule. We may here refer to the definition by Zadeh (1998) and Lin (1997, 2009), that each granule is partitioned by an object, and that each granule is drawn together via indistinguishability, similarity, proximity, or functionality. Because the entity set  $V$  can reflect the partitions of the concept to be described, we call it the *granular topology of ontology*. There are various granular topologies for the same concept, and each topology corresponds to a kind of partition for the concept.

An ontology should be either a concept that can be partitioned or the partitioned granule of a concept. For example, an ontology of architecture may be partitioned into the combination of the entities of window, wall, roof, etc., and each of those entities may also be defined as an ontology of corresponding conceptualization.

Here the “granular ontology” does not simply refer to an inventory of entities in reality that all belong to a granular partition (Bittner & Smith, 2003), but also to content the crisp or fuzzy conditions of the granules, the topology structure of the granules, the assembly relations among the granules, etc.

### 3.2. Ontology are granular approximations of conceptualizations

It is natural to consider ontologies only as approximations of desired conceptualizations in information science (Sure Staab et al., 2004). The instances deduced from the knowledge rules in  $R$  may contain exceptions that do not belong to the corresponding domain. For example, if the concept of “bird” is defined with the two rules “it is an animal” and “it can fly”, there may be exceptions such as dragonfly. So the rule set may be fuzzy or rough with respect to the domain. And the whole rule set  $R$  of an ontology can be divided into three categories:

- The rules that present the essential features of the desired domain.
- The rules that present the relevant non-essential features of the desired domain.
- The rules that present the irrelevant features of the desired domain<sup>3</sup>.

Similarly, each rule in  $R$  can also be viewed as one granule in a granule set that can approximate the real knowledge relation  $\mathfrak{R}$ , Fig. 2 demonstrates the granular knowledge structure for the conceptual relations in ontology. We have implemented this structure in our previous work (Liu et al., 2008). Because the essential features are necessary for the corresponding domain, from the viewpoint of granular ontology, the essential features are the core of the knowledge structure of the domain. These features, therefore, make up the internal granule set of  $\mathfrak{R}$ . The relevant non-essential features are the external granules of  $\mathfrak{R}$ , and the irrelevant features

are unrelated granules of  $\mathfrak{R}$ . As shown in Fig. 2, internal granules are located absolutely within the boundary of  $\mathfrak{R}$ . External granules cross the irregular boundary of  $\mathfrak{R}$ , which means they identify both instances that are belong to the desired domain and instances that do not belong to the desired domain. Unrelated granules are located entirely outside the boundary of  $\mathfrak{R}$ .

A typical example of a relation set is shown in Fig. 2(b). The concept of “bird” consists of two rules, which are both external granules of the domain, and those two rules can approximate a concept domain of “bird”, although the granules overflow the real conceptual relation set  $\mathfrak{R}$ . The granularity of ontology also means that the same concept may be expressed by different ontologies with different granular knowledge structures. Fig. 2(c) also depicts the relation set of “bird”, but with different rule granules, such as the basic component of a bird: beak, head, wing, etc. The concept of “bird” in both Fig. 2(b) and (c) is the same—they have the same conceptual relation mapping set  $\mathfrak{R}$ —but the real ontology domain may be vary due to their different granular knowledge structures.

Furthermore, the granular structure of the knowledge set in ontology is determined by its granular topology, i.e., the entity set  $V$  of ontology. So the entity set of ontology is the key point of the granular level in ontology, the basic components in granular topology may reflect the ontology's cognition level, which is obviously granular and hierarchical.

As shown in Fig. 2(b), the ontology rule set  $R$  may include only external granules. In this case, the desired concept is approximated by several related rules. Fig. 2(c) represents the boundary of the real concept more accurately than Fig. 2(b).<sup>4</sup> We can conclude that the size of the granule (or granular level) in cognition affects the accuracy of ontology representation; normally, the smaller the granules, the higher the accuracy.

Note also that in Zadeh's definition, granules may be natural crisp or fuzzy. Because the external granules in a rule set may lead to instances that do not belong to the specific target domain, they are also called fuzzy granules. Both the internal granules and unrelated granules are called crisp granules.

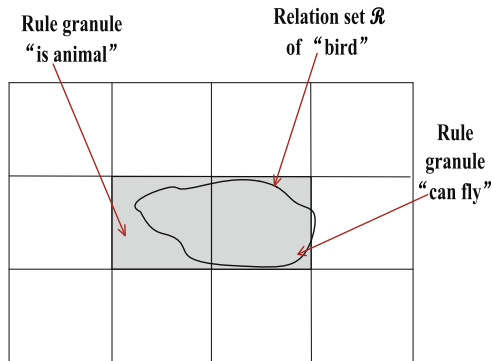
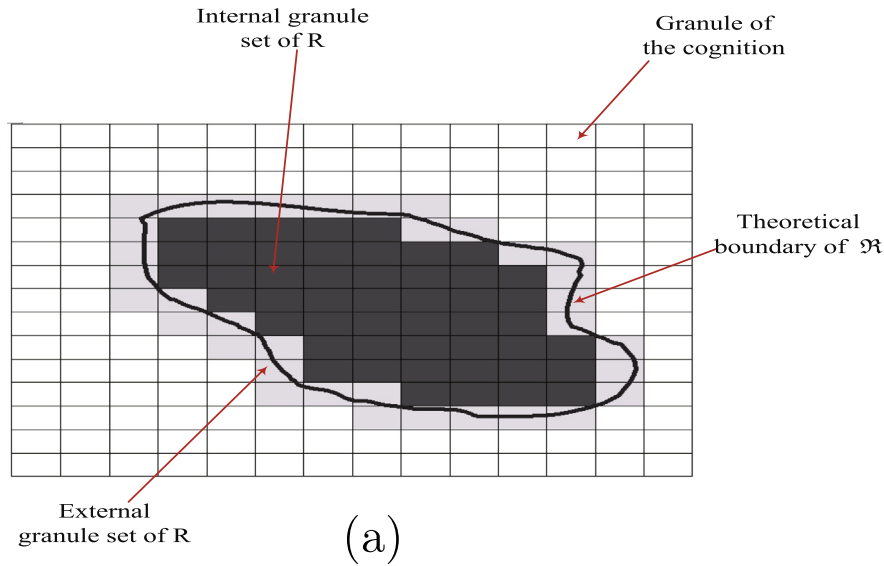
### 3.3. Rule granules are tuples

Because rule granules represent relations among the entities in a granular topology and form target concepts, they should not be treated individually. The capability of a knowledge rule set should be evaluated as a whole, instead of each rule evaluated one by one. This is particularly true when the desired concepts are complex enough to implement with a description logic (DL) language. A typical sample is the ontology for 3D architecture models in Fig. 1—the spatial and combination sequences among the entities are quite difficult to represent with DL language alone, so FOL-based rules—e.g.,  $r_1 - r_4$  in Section 2 – are required. Thus rule granules are sequence-sensitive and are also tuples (i.e., ordered sets).

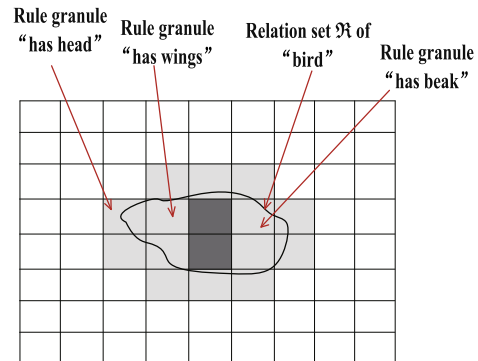
In the architecture ontology, although  $r_2$  and  $r_3$  are quite similar, if we use two different rule sets  $R_1 = \{r_1, r_2, r_4\}$  and  $R_2 = \{r_1, r_2, r_3, r_4\}$ , the corresponding instance spaces will be quite different. Thus they may lead to two approximate ontologies with respect to the desired conceptualization. In practice, the result of approximating the digital architecture domain of  $R_1$  may be better than  $R_2$ . So we can conclude that not all external rule granules are necessary for a desired ontology, only the proper tuple (sequenced rule set) of those rules will best fit the desired concept. This is why we need to optimize the rule set in our following granular ontology approach.

<sup>3</sup> In practice, designers may introduce some rules that are not correlated with the desired conceptualization or the sub-conceptualizations to form the desired conceptualization, especially when the destination is indistinguishable.

<sup>4</sup> If rule granules are small enough, there must be a rule set to fit the boundary of a real concept precisely.



(b) Granular level 1



(c) Granular level 2

Fig. 2. Knowledge structure of a conceptual-relation set in a granular view. In figure (b), these two rules cannot cover all the category of “birds”, for example, the penguins and ostriches are birds, but they cannot fly.

4. General approaches for granular ontology design

There are four steps in our approach of ontology design. The four steps are as follows:

4.1. Initialization

In the initialization phase, we mainly perform three preparatory tasks on the desired ontology domain: specifying the granular topology on which the ontology will be established, confirming the boundary of the ontology domain, and designing the evaluation standard for the instances of the specific ontology domain.

The choice of granular topology is the most important task in granular ontology design, and the proper granular partition will simplify the ontology design and reduce the vagueness of the ontology terms significantly. The specification of a granular topology is a complicated task, which may concern factors such as the complexity of the target, the hierarchical structure of the target, and methods adopted for construction of the ontology.

As mentioned above, using smaller granules for an ontology will approximate the real concept more accurately. So the question

arises: Should we design the ontology with granules as small as possible? The answer is obviously *no*. Representing an ontology with very small granules will lead to problem due to the intractability of the inter-relationships among tiny granules. Thus the cost of ontology design will increase significantly. For example, to describe “bird” using DNA sequences will likely be excessively difficult. A simple choice of granular topology for a desired domain may be addressed using the natural semantic partitions of the target in human cognition. For example, in our digital architecture ontology design, we use the basic architectural terminologies  $v_1, \dots, v_8$ , in Fig. 1.

The boundary of the ontology domain is established on the granular topology. With a clear granular topology for the ontology domain, we can specify the target boundary both on the macroscopic (or parent) conceptualization level and the microscopic (or children) granular level. This granular topology will improve the descriptions of the boundary.

Evaluation standards are also concerned with granular level—they reflect the granules as a complete instance in a certain granular topology. Note also that evaluation standards only focus on the features of the instances in desired domains, rather than conceptual relations. In this task, the evaluation items for an instance

we selected should be objective, correct, common, clear, and feasible. These items are important evaluation criteria in the following ontology optimization phase.

After we complete the three operations in the initialization step, the design of an ontology will be constrained within certain boundaries and the granules and granular topology structures used in the ontology will also be confirmed, thus the subjective bias will be reduced greatly.

#### 4.2. Extension

In this phase, we produce the primary ontology based on the granular topology and then extend the terms of the primary ontology. The representation of the ontology may be DL (e.g., “*has\_value*”), FOL (e.g., the procedural rules in Section 2) or a synthesis of both DL and FOL. We enumerate the entities involved in the ontology domain following the above granular topology, and then generate the corresponding conceptual relations for those entities. As the conceptualizing processing may introduce new entities, we extend these new entities into the entity set  $V$ , and then generate the corresponding conceptual relations iteratively until there are no more related entities. To avoid unlimited extension for the entities and conceptual relations, a proper pruning with the restrictions from the ontology domain boundary and the size of granules is needed. For example, a designer may present a rule “the roof is wooden”, but this rule will be pruned because it is unrelated to the ontology target boundary presented in the previous section.

In this phase, there may be multiple participants, and the granular partition for the ontology domain can naturally support multiple designers working in collaboration.

The primary ontology domain can also be constructed with traditional methods, such as an inspirational approach, inductive approach, deductive approach, or collaborative approach (Holsapple & Joshi, 2002), according to the predetermined granular topology. Then the entities and corresponding conceptual relations can be extended.

#### 4.3. Inconsistency reduction

Because the primary ontology may be designed by multiple participants with different understanding of the same ontology domain, designers may confuse and mismatch the granular topologies. Therefore, we must reduce the number of inconsistencies in the primary ontology.

Two types of inconsistencies should be removed—inconsistencies introduced because of differences in participant cognition and inconsistencies caused by conflicts among rules represented by non-monotonic DL. The former is a typical synthesis problem for multiple persons' viewpoints in collaborative ontology design. Our multilevel-projection design approach (Liu, Xu, Zhang, & Pan, 2006b; Liu et al., 2008) can process this condition comprehensively. The latter problem may be more covert and harder to reduce.

For the former inconsistency, we first mark all the entities and conceptual relations that are referred by more than one participant from the primary ontology. Considering that the participants may come from different fields and their views on the same domain may not agree, for each commonly marked pair of entities, we project the entities into the same cognition level with the predetermined granular partition, and check whether conflicts occur. Because the inconsistency may only appear at a certain cognition level, the projections for marked pairs should be performed at all the participants' cognition levels that involve in the pair. After conflicts are detected, we revise the corresponding ontology terms and

then repeat the projection and check iteratively until we reach a consensus.

#### 4.4. Optimization in granular cognition

For inconsistencies that arise from conflicts among rules represented by non-monotonic DL, the origin of the inconsistencies may be that humans always understand the real world using non-monotonic logic, so the designed ontology is only an approximation of human cognition with non-monotonic logic. Although various methods (Ensan & Du, 2008; Pührer, Heymans, & Eiter, 2010) can be used, and a clear granular topology for the desired concept (such as in Fig. 3) will reduce the number of inconsistencies, not all potential inconsistencies of DL implementation can be removed or even detected, so we must process those in the next optimization phase—optimization of the knowledge rule set by selecting a proper subsets without redundancies.

The task of designing a good ontology for complex problems from a granular level can also be divided into two phases: the first one is to design the knowledge relation set under a perfect granular structure. The second is to present a primitive, integrated, redundant and approximate knowledge rule set under a tolerable ontology granular topology and then select a suitable subset that can fit the desired concept mostly, according to a certain evaluation method. From the granular viewpoint of ontology, we realize that it is quite difficult to design an ontology in a perfect granular topology that can represent all of the granules of knowledge crisply. So the task for ontology design always becomes a compromise between human cognition and practice implementation. The aim of this phase is to select the proper knowledge rule subset with respect to the specific concept. Optimization is different from inconsistency reduction—in the optimization phase, we do not revise the entities or conceptual relations, we only construct the optimal rule subset from the redundant raw set. The inconsistency reduction phase focuses on removing logical conflicts due to subjective bias in ontology design.

In the optimization phase, before optimal relation subset selection, we first classify the original knowledge rules into internal granules and external granules. This process is evaluated by several experts who come from different background and they discuss and confirm whether the conceptual relation is certain or marginal for the desired ontology domain.

Secondly, we should provide several optional rule sets for further evaluation. As the tuple-structure granules of conceptual relation may be order-related, which also implies different combinations of knowledge rules will produce varied extensions, it also should be noted that the knowledge set covered with all the redundancy and fuzzy granules (such as the internal granules and the external granules in Fig. 2) is not always optimal.<sup>5</sup> So those optional sets should be considered as a whole carefully. Each candidate should include all the internal granules and several elements from the external granule set. Designers should make a deliberate choice for those from external granules; a typical guide specification is to put fuzzy rules that may relate the same objects to a different candidate set, and this is addressed on that the knowledge in primitive ontology is redundant and we should evaluate which representation for the same object is better.

Thirdly, we use different knowledge rule candidate sets to generate a mount of instances,<sup>6</sup> evaluate those generated instances

<sup>5</sup> This is especially true, when there are many grammar rules implemented with FOL.

<sup>6</sup> With FOL-based rules to describe the combinations of basic components in  $V$ , a knowledge system can produce corresponding instances by enumerating the combinations of those rules, and this may be implemented by software toolkits or computer programs.

with the predetermined standards and criteria defined in the initialization phase, and choose the subset with the highest correct ratio as the optimal result. As there are many instances concerned in this step and most of the predetermined standards are subjective and can be represented in computers easily, we use a hybrid method for the evaluation of the instances. That is, most of the instances are automatically verified by some software toolkits, and some complex instances may be evaluated by experts to decide whether each instance belongs to the desired ontology domain. Obviously, the optimization result highly relies on the number of instances, and we cannot enumerate all the possible instances generalized by the optional conceptual relations sets, so the final result is only an optimal approximation, which is a compromise between accuracy and cost.

## 5. A case study that uses the granular ontology design approach

In this section, we use a real case to demonstrate the proposed granular ontology approach. The process is shown in Fig. 4. We design an ontology for modeling architecture with specific styles (Liu et al., 2006b, 2008, 2010), such as ancient Chinese architectural style. This case involves multiple fields and participants, such as computer programmers, architects, rendering artists, historians, and UI designers. The task is to design an ontology for digitally modeling the ancient Chinese architectural styles. In the following section we use  $C = \langle D, W, R, V \rangle$  to denote the desired ontology, where  $R$  is an approximate of the conceptual relations  $\mathfrak{R}$ , and  $V$  is a sub-concept (or sub-ontology) set involves in the domain of the ancient Chinese architecture.

We also use an example that contains two instances of ancient Chinese architecture to demonstrate the ontology design processing, shown in Fig. 5.

### 5.1. Initialization and extension

In the initial phase, we establish the architecture ontology on the semantic granular topology, which conceptualizes the ontology

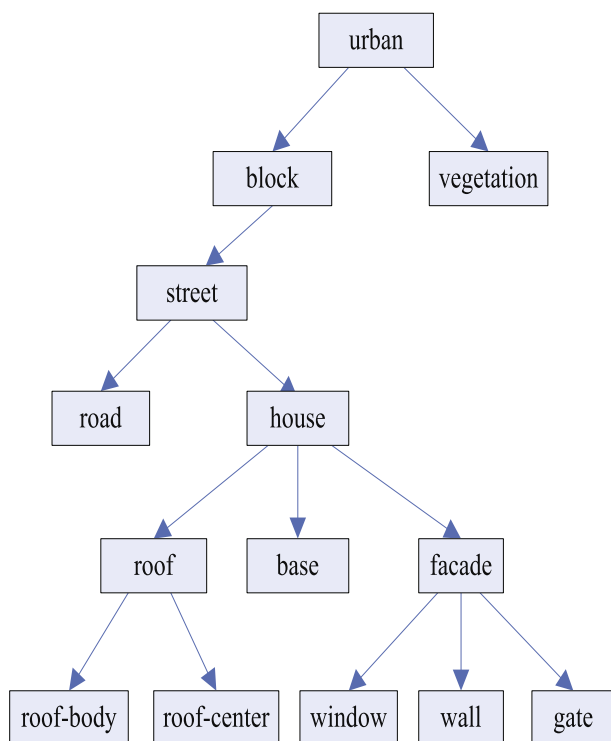


Fig. 3. Granular topology in an ontology of digital 3D architecture.

in natural semantic partitions (division into house, window, wall, etc.) and is easy to communicate for the different designers. Part of the topology is shown in Fig. 3. The sub-concepts in that granular topology consist of the  $V$  set in our designed ontology. Each sub-concept in  $V$  can also be regarded as a new ontology that is denoted as  $v_i = \langle D_{v_i}, W_{v_i}, R_{v_i}, V_{v_i} \rangle$ . For example, the  $V$  set of the ontology of “urban” in Fig. 3 contains two elements, “block” and “vegetation”, which can also be regarded as ontologies.

As we cannot enumerate all of the conceptual relations (also called knowledge) in  $\mathfrak{R}$ , we use some FOL rules and DL rules to approximate the conceptual relation set. We use L-system-based grammar as the knowledge rule, e.g.,  $r_1 - r_4$  in Section 2, which uses recursive grammar to generate different architecture instances (Liu, Xu, Pan, & Pan, 2006a; Müller et al., 2006). The subjective evaluation features and boundary are also presented by those experts carefully with DL rules based on their observations of the instances of the desired domain. Once the granular topology is confirmed by experts from different domains, the boundary of our ancient Chinese architecture ontology can be clarified with the hierarchical granular topology, and the evaluation standards for the desired concept or involved sub-concepts can be generally presented by those experts who only need to focus on one of the sub-concepts with DL rules. For example, the experts from architecture may present some combinations of the windows and walls that belongs to the type of ancient Chinese architectures, while the experts from arts may focus on the color feature of the windows, walls, or roofs. All these DL rules from different experts consist of the initial evaluation standards.

With a granular topology, the ontology can be designed collaboratively, thus each designer may only focus on a certain sub-concept, such as windows or walls, that are involved in the ancient Chinese architecture concept. Then the extension phase in our granular ontology design process may be parallel and independent.

In our example of Fig. 5, each participant presents an ontology for ancient Chinese architecture. The architect will present a sample ontology  $C_a(D_a, W_a, V_a, R_a)$  to define the vernacular house in ancient Southeast China. The example only contains two instances of the Southeast vernacular houses ( $w_1$  and  $w_2$ ),<sup>7</sup> and two combination rules with DL for the domain<sup>8</sup>; The programmer uses FOL-based grammar, which could also support the quantitative control terms and spatial control terms. In our approach, we use the procedural modeling engine (Müller et al., 2006) to randomly select the rules, such as the programmers’ rules,  $R_p$  in Fig. 5, and recurs those selected rules to generate semantic scene instances; The ontology presented by historians will focus on the detailed features of the components, such as the roof body ( $v_2$ ) in Southeast ancient Chinese architecture, the example is given in  $R_h$  of Fig. 5; The ontology of rendering artists normally presents the basic geometry of components and some complex components such as the roof body, which may be represented by the point-mesh files, such as the “roof.obj” in the example; UI designers design the annotations with the different requests of end users. The scene for historical users may focus on the annotation of the typical events and backgrounds, while the scene for architectural users may focus on the annotations of the layout and combination of each component. As shown in Fig. 5, rule one of the UI designer, all the combinations of  $(v_1, v_2)$  will be linked with the annotations defined in that rules automatically, when the huge scene is generated. Then

<sup>7</sup> In a real case, there may be many more instances in the instance set  $W$  or even innumerable instances

<sup>8</sup> Both rules in the example describe the combination of houses. In practical implementation, it may contain the combination for a group of the components in  $V$  or some attribute description for the components similar with the example of a historian’s ontology.

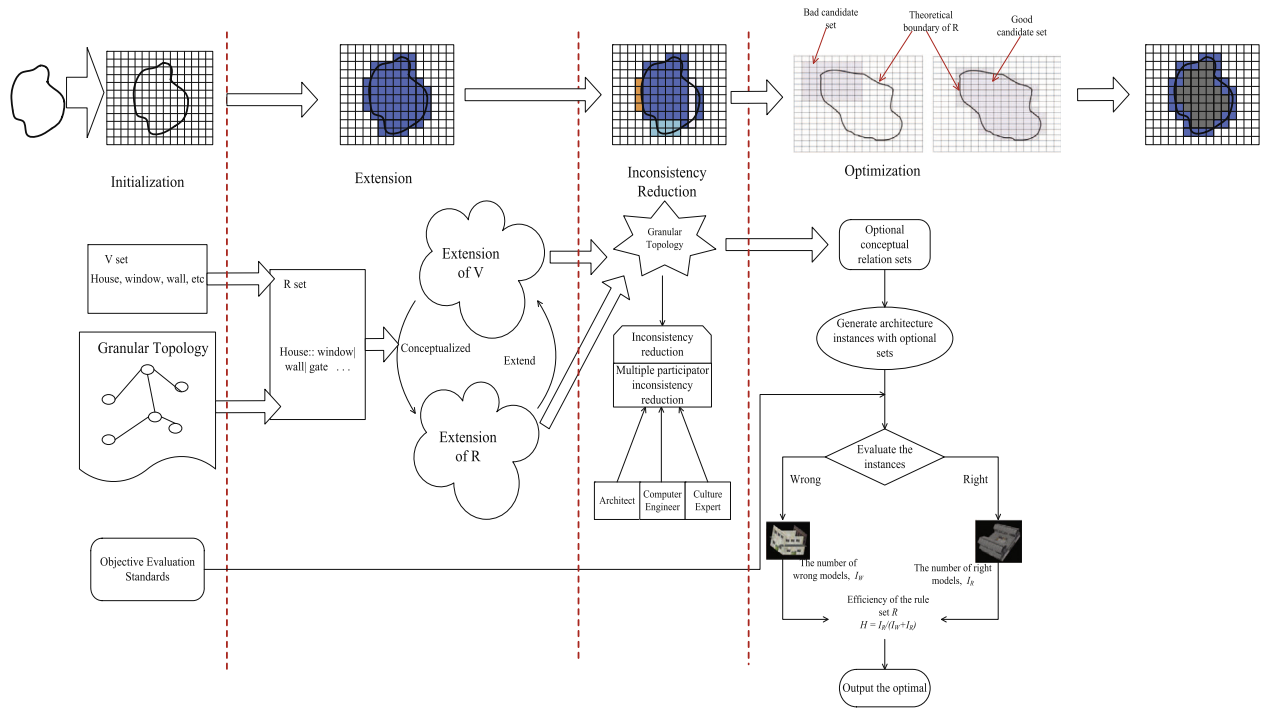


Fig. 4. A granular ontology design approach to architecture modeling.

the modeling engine will automatically check the scene and attach these annotations when outputting the final results.

5.2. Inconsistency reduction

A hierarchical granular topology can simplify the design of a complex ontology. However it may introduce inconsistencies, because ontologies created by different domain experts may have conflicts. Inconsistency reduction can identify and resolve these inconsistencies. Several key operations used in our approach are as follows:

5.2.1. Mapping

Inconsistencies among ontologies mostly occur in corresponding domains that describe the same concept. We call the relationship between corresponding concept domains “mapping”. If ontology  $C_1$  is in a mapping relationship with ontology  $C_2$ , denoted as  $maps(C_1, C_2)$ , then

$$maps(C_1, C_2) \rightarrow \exists C(D, W, R, V), v = C_1 \cup C_2, v \in V$$

This definition states that if  $C_1$  and  $C_2$  are in a mapping relationship with each other, the union concept of  $C_1$  and  $C_2$  can be viewed as a sub-concept element in a higher level ontology  $C$ . For example, in Fig. 6, house in the street and architecture ontologies, roof in the house and architecture ontologies, and gate in the house and architecture ontologies are in mapping relationships.

If  $C_1$  is mapping with  $C_2$ ,  $w$  is an instance of ontology  $C_1 (w \in W_1)$ , then we call instance  $w$  semantically related to ontology  $C_2$ .

$$maps(C_1, C_2), w \in W_1 \rightarrow w \text{ is semantically related to } C_2$$

5.2.2. Part-of and consistency

When using FOL rules to generate ontology instances, the sub-concepts (vocabularies) will first generate their own instances and then they can be combined into instances of parent concepts

using the domain knowledge  $R$ . Before detecting inconsistencies between ontologies, we define the operation of decomposing an instance. This is the Part-of operation.

Let  $C_k(D_k, W_k, R_k, V_k)$  be a sub-concept of Ontology  $C(D, W, R, V)$ , where  $w \in W, C_k \in V$ , Part-of( $w$ ) $_{C_k}$  is defined as

$$Part-of(w)_{C_k} = \{v_1, v_2, \dots, v_i\}, v_i \in w, v_i \in W_k$$

Here  $C_k$  is a sub-concept of the  $C$ , so  $v_i$  should belong to both the instance  $w$  and  $C_k$ 's instance set  $W_k$ . If  $C_k$  is not a sub-concept of  $C$ ,  $Part-of(w)_{C_k} = \{\cup Part-of(w)_{C_j} | C_j \in V, maps(C_j, C_k)\}$

Obviously, if  $C_k$  is a concept that contains  $C$ , or it cannot find any sub-concept mapping with  $C_k$  in  $V$ ,  $Part-of(w)_{C_k} = \phi$ .

We introduce the concept of consistent set and conflict set (Chen, Chen, & Zhang, 2007) to distinguish between the types of inconsistencies among ontologies presented by different experts.

Let  $w$  be an instance of  $C(D, W, R, V)$ . To an ontology  $C'(D', W', R', V')$ ,  $w$ 's consistent set with respect to  $C'$  is defined as

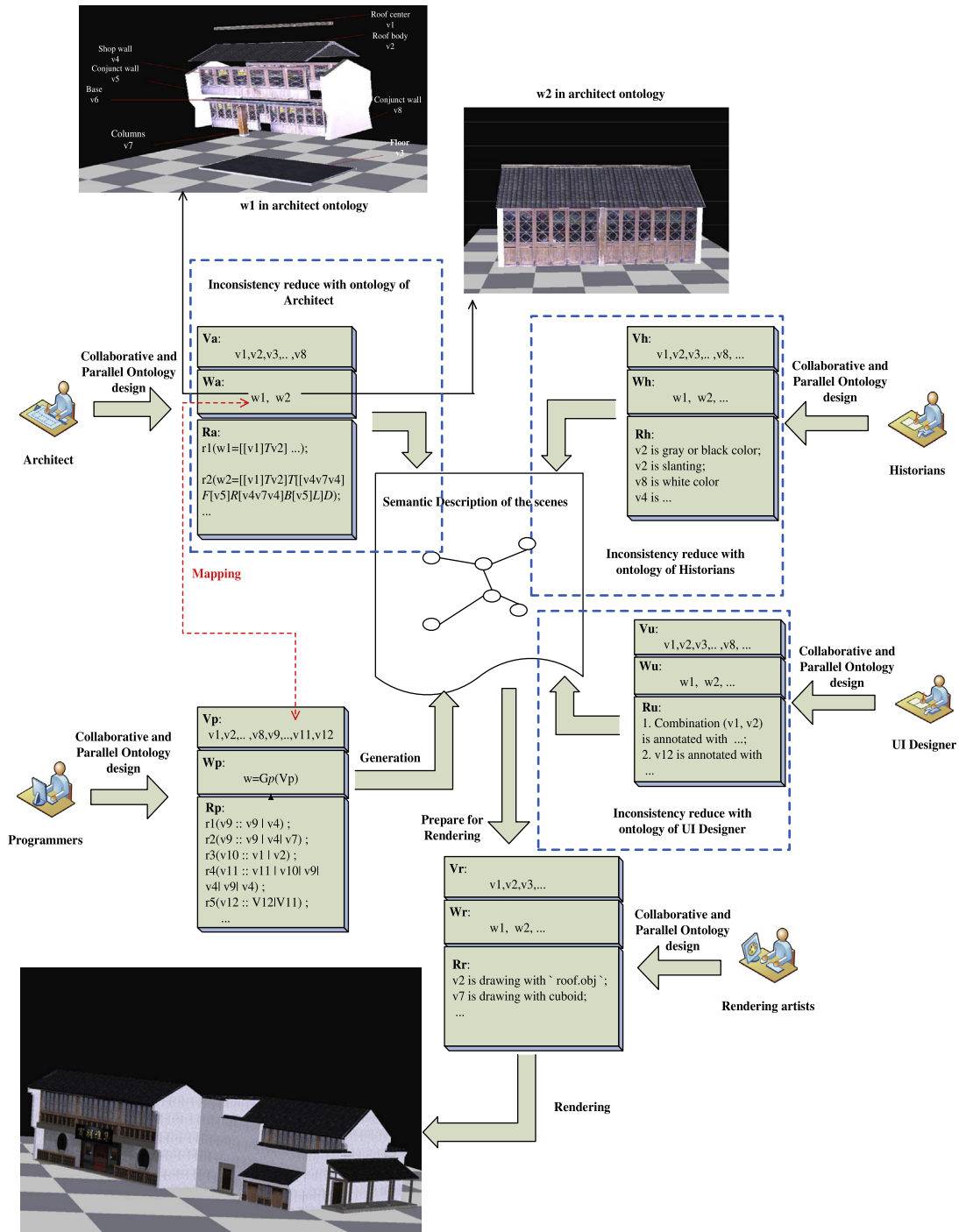
$$consistent(w)_{C'} = \{\beta | \beta \in Part-of(w)_{C'}, \forall r \in R', \beta \text{ satisfy } r\}$$

And  $w$ 's conflict set with respect to  $C'$  is defined as

$$conflict(w)_{C'} = \{\beta | \beta \in Part-of(w)_{C'}, \exists r \in R', \beta \text{ unsatisfy } r\}$$

$\beta$  satisfy  $r$  means that the component  $\beta$  can fit the domain knowledge  $r$ . And there are two unsatisfy conditions in our ancient Chinese ontology design case:

- Attribute inconsistency, which means that the attributes generated in instances are in conflict with the corresponding knowledge rules. For example, the rule  $r(\text{roof.has\_shape} = \text{cornice})$  will be in conflict with an instance of  $\text{roof.shape} = \text{plane}$ .
- Combination inconsistency, which means that the combinations generated for instances conflict with the corresponding knowledge rules, e.g., the combination rule  $r(P = v_1, v_2, v_3)$  conflicts with the combination in instance  $P_I = v_3, v_3, v_1$ .



**Fig. 5.** Ontology examples from multiple domain experts. Here, the means of  $v_1, v_2, \dots, v_8$  are shown in the left-top figure, and  $v_9$  is the window wall component,  $v_{10}$  is the roof component,  $v_{11}$  is the house (which also maps with the instances  $w_1$  and  $w_2$  in the ontology of architect),  $v_{12}$  is the urban block. Sample ontology of architect presents a southeast ancient Chinese architecture domain with only two rules, which indicate two styles of houses shown in the top of this figure. The hierarchical concept domains should be established by the participants earlier. The inconsistency reduction phase could proceed the mapping relationships between the instances and ontologies automatically, detect and reduce the inconsistencies (Liu et al., 2012b).

If  $conflict(w)_C = \phi$ ,  $w$  is consistent with ontology  $C'$ , otherwise  $w$  is inconsistent with ontology  $C'$ .

In our ancient Chinese architecture ontology design problem, we aim to reconstruct 3D architectural scenes of ancient China. We can reduce inconsistencies by minimizing conflicts among concepts and sub-concepts designed by different experts. In our design approach, we use the FOL rules and DL rules in each sub-concept to generate a number of instances and then input the sub-ontologies (or sub-concepts)  $C_1, C_2, \dots$ , designed by different experts, to test whether the generated instance  $w$  is in conflict with the

sub-concepts. By this method, we can remove the inconsistencies among the ontology concepts designed by different experts semi-automatically—our system can automatically generate the instances and locate the conflict rules, then revise the rules or remove the rules manually (Liu et al., 2012b).

### 5.3. Optimization in granular cognition

In the processing of real architecture modeling, the FOL rules will generate arbitrary combinations of architecture instances.



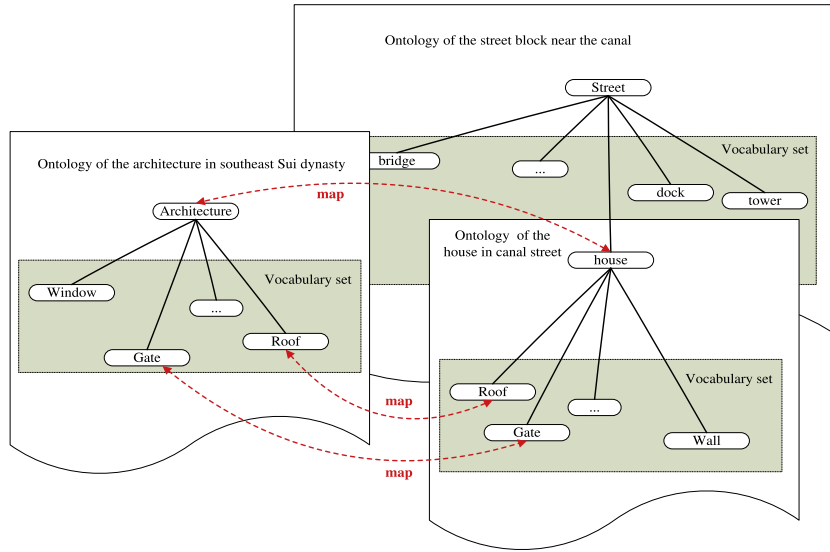


Fig. 6. Hierarchical ontology mapping with multiple sub-concepts (Liu et al., 2012b).

They cannot match the conceptual relations of the ontology very well. The real knowledge set of a specific domain will lie in an irregular shape, like the one in Fig. 2. Each grid represents a FOL rule in the approximate rule set  $R$ , and we denote all the FOL rules as a set  $U$ . The real knowledge set will cross some grids, which means that those crossed rules will be classified either as that knowledge set or some other knowledge set, and within vague boundaries in the rule based granule. According to the figure, we can see that none of the rule subsets of  $U$  can match the boundary of the real knowledge set exactly. However,  $\mathfrak{R}$  can be approximated by choosing the best-matched subset in  $U$ . That is the core idea of our optimization phase in the granular ontology design approach.

$P$  is a subset of  $U$ : it contains rules that may not match well with the boundary of the knowledge set  $\mathfrak{R}$ .

Though  $\mathfrak{R}$  in  $C = \langle D, W, \mathfrak{R}, V \rangle$  cannot be enumerated directly, a small rule set  $K$  in  $\mathfrak{R}$  can be determined manually. In the implementation,  $K$  is initialized by several basic architecture modeling rules that are necessary for generating architecture. Those rules in  $K$  are decided upon by all the experts involved in the design of the ontology. Then, one rule at a time, we tests whether a rule  $r$  in  $P$  can be mapped by  $F_{K \cup \{r\}}(V)$  into an instance  $w \in W$ . By this method, we can find the upper and lower approximations of  $P$  with respect to the ontology domain  $D$ . Similarly,  $\underline{NP}_D$  and  $\overline{NP}_D$  can also be defined by testing the rule  $r$  not in  $P$  (that is  $r \in U - P$ ) whether  $w = F_{K \cup \{r\}}(V)$  belongs to the instances set  $W$ . We proceed as follows:

Let the universe  $U (U \subseteq R)$  be the whole FOL rule library (knowledge) in our auto-modeling system (Liu et al., 2006a). Let  $P$  be a subset of  $U$ , then approximate sets within the context of  $C = \langle D, W, R, V \rangle$  is<sup>9</sup>:

$$\overline{P}_D = \bigcup \{r \mid \exists w, \text{ having } w \in W, w = F_{K \cup \{r\}}(V), \text{ and } r \in P, K \subseteq R\} \quad (1)$$

The lower approximation of  $P$  is

$$\underline{P}_D = \bigcup \{r \mid \forall w, \text{ having } w \in W, w = F_{K \cup \{r\}}(V), \text{ and } r \in P, K \subseteq R\} \quad (2)$$

Here,  $F_{K \cup \{r\}}(V)$  is the architecture instance function of the ontology domain  $C = \langle D, W, R, V \rangle$ . In our modeling system, the architecture instance function refers to the process of using the rule set  $K \cup \{r\}$  to combine the vocabularies (sub-concepts)  $V$  into a single architecture instance.

An intuitive explanation for the upper and lower approximations can be found in Fig. 2. The upper approximation  $\overline{R}$  consists of all of the granules related to  $R$  (the external granule set of  $R$ ), and the lower  $\underline{R}$  consists of all of the granules that are all strictly within  $R$  (the internal granule set of  $R$ ).

Similarly, we can also define the approximate sets of the knowledge set  $\overline{P}$  ( $\overline{P} = U - P$ ) with respect to the concept domain  $D$ .

The knowledge set  $\overline{P}$ 's upper approximate set with respect to the concept domain  $D$  is

$$\overline{NP}_D = \bigcup \{r \mid \exists w, \text{ having } w \in W, w = F_{K \cup \{r\}}(V), \text{ and } r \in U - P, K \subseteq R\} \quad (3)$$

The knowledge set  $\overline{P}$ 's lower approximate set with respect to the concept domain  $D$  is

$$\underline{NP}_D = \bigcup \{r \mid \forall w, \text{ having } w \in W, w = F_{K \cup \{r\}}(V), \text{ and } r \in U - P, K \subseteq R\} \quad (4)$$

Though the notion of roughness (Pawlak, 1991) is a rough set notion, it is easy to generalize to granular computing. The roughness of the knowledge set  $P$  with respect to the domain  $D$  can be calculated as

$$\chi_{P_D} = \frac{|\overline{P}_D| - |\underline{P}_D|}{|\overline{P}_D|} \quad (5)$$

The corresponding roughness of knowledge set  $\overline{P}$  with respect to the domain  $D$  can be calculated as

$$\chi_{NP_D} = \frac{|\overline{NP}_D| - |\underline{NP}_D|}{|\overline{NP}_D|} \quad (6)$$

### 5.3.1. Measurement on minimal knowledge boundary

After defining the upper and lower approximations and the roughness of the rule set  $P$  with respect to a certain ontology domain  $C = \langle D, W, R, V \rangle$ , we need to select the best FOL rule set for the ontology domain. That is we must find the rule set  $P$  that is closest to the knowledge set  $\mathfrak{R}$  in the ontology domain. To

<sup>9</sup>  $\forall w \in W$  can be determined by enumerating a large number  $N$  of instances. We believe the negative condition will occur when  $N$  is large enough. Note that  $K$  is a small rule set in  $R$ .

determine the closeness between the rule set  $P$  and the knowledge set  $\mathfrak{R}$  in an ontology domain, we need a critical metric for closeness. The measurement should be able to determine the *fitness* of the rule set  $P$  with respect to the real domain knowledge  $\mathfrak{R}$ .

In our solution, we adopt a roughness-function-based metric to identify the minimal knowledge boundary between the selected rule set  $P$  and the ontology domain knowledge  $\mathfrak{R}$ . Intuitively, the roughness function definition can be understood as the selected rule set  $P$  containing at most the true knowledge (represented by  $\chi_{P_D}$ ) and  $U - P$  containing at least the true knowledge (represented by  $\chi_{NP_D}$ ).

In this article, we need a roughness function metric that can minimize the roughness of the knowledge set  $P$  with respect to the desired ontology domain and maximize the roughness of knowledge set  $\bar{P}$  with respect to the desired ontology domain. The roughness function measurement is defined as

$$E_{P_D} = \chi_{NP_D} - \chi_{P_D} \quad (7)$$

**Table 1**  
User study results.

Input set	Average score	Standard deviation
$U$	6.95	2.85
$P$	7.32	1.40
$T$	3.31	1.03

Then the task of finding the most suitable knowledge set with respect to the ontology domain is equivalent to calculating each knowledge rule set's roughness function measurement and choosing the knowledge set with the maximum roughness function measurement as the best domain knowledge set. The appropriate knowledge set  $P$  with respect to the ontology domain  $D$  can be calculated by the following formula:

$$P^* = \arg \max (E_{P_D}) \quad (8)$$

Then the optimization can be regarded to select a proper FOL rule subset  $P$  that can maximize  $E_{P_D}$  (Liu et al., 2010).

```

<Urban name = "Southeast_Urban_near_Canal">
  <Block>
    <hasTopology rdf:resource = "Topology.owl#block_southeast"/>
    <hasRoad rdf:resource = "Road.owl#road_in_block_southeast"/>

    <House name= "Southeast_House_1023">

      <hasCombination>
        <!--To identify the combination more clearer, ----->
        <!--we use the symbols in the ontology example, ----->
        <!--that is "[[v1] Tv2]T[[v4]F[v5]R[v4v7v4] B[v5]L]D " -->

        <Component Layout="Top">
          <Component Layout="Top">Roof_Center_2131</Component>
          <Component>Roof_body_3207</Component>
        </Component>
        ...
        <Component Layout="Left"> Conjunct_wall_1124</Component >
        <Component Layout="Right"> Conjunct_wall_1124</Component >
        ...
      </hasCombination>
      ...
      <Roof_Center name="Roof_Center_2131">
        ...
      </Roof_Center>
      <Conjunct_wall name="Conjunct_wall_1124">
        <hasRender rdf:resource="Render.owl#cw_southeast"/>
        <hasBoundary>
          <BoundaryType>Box</BoundaryType>
          <BoundaryWidth>3.2</BoundaryWidth>
          <BoundaryHeight>7.5</BoundaryHeight>
          <BoundaryThickness>0.3</BoundaryThickness>
        </hasBoundary>
        <hasColor>
          <BodyColor>Red</BodyColor>
        </hasColor>

      </Conjunct_wall>
    </House>
    ...
  </Block>
</Urban>

```

**Fig. 7.** Sample semantic description of a scene instance (here limited to a page). We have omitted the complex control tags for the components (Liu et al., 2010).

### 5.3.2. Results analysis on optimization

In our ancient Chinese architecture ontology design, there are a totally of 224 FOL-based knowledge rules, which are similar with the  $r_1$  in Section 2, after the extension. As there may be inconsistencies among the rules presented by multiple experts, inconsistency reduction is also carried out for the DL-based knowledge rules both with manual checks and automatic detection.

During the optimization phase, shown in Fig. 4, we must evaluate the FOL-based rule set by instance, so the computer expert implements an instance-generation toolkit (Liu et al., 2010) that can randomly choose rules from the rule subset that needs to be evaluated and generate corresponding architecture instances. The instances are represented in XML format. A sample is shown in Fig. 7, and our team also implement a toolkit to render the instances of architecture models. Then the evaluation of instances is first executed automatically by matching the DL-based rules and XML-based instances. Experts review some of the intractable instances using the visualization toolkit.

In the optimization phase, we generate a total of 2000 instances and record the domain hit ratio of each candidate rule subset. The domain hit ratio is calculated as the number of instances that correctly belongs to the desired domain divided by the total number of instances (Liu et al., 2010). The rule subset with the highest domain hit ratio is our optimal result. Finally, we obtain a subset with 105 FOL based rules.

To evaluate the effectiveness of our result, we invited 40 graduated students from different background to evaluate the instances generated by different knowledge rule subsets, which are the full rule set  $U$  with 224 rules, the optimal rule set  $P$  with 105 rules, and a rule subset  $T$  with 93 rules, which is randomly drawn from  $U$ . Each rule sets is used to generate 29 instances by the instance generation toolkit, and the 87 instances are randomly presented to the testers and the testers grade the quality of the instances with respect to the desired concept, using the following scores: 0 for totally wrong, 5 for moderate, 10 for perfectly right. Value between 0 and 5 are for less correct and values between 5 and 10 are for more correct. The average scores of every sets are shown in Table 1. From the table, we can conclude that the optimization result can achieve the best performance.

## 6. Conclusion

We have presented three views of granular ontology observation from the real world, and a corresponding granular ontology design approach based on the three views. This granular approach can be applied to generate ontologies for various applications, such as the Semantic Web, information classification, service discovery, and other complex phenomena.

The granular ontology design approach integrates the inductive and deductive methods and the granular partition in our approach will help multiple participants to design the ontologies collaboratively. Although ontology design is subject to much subjective bias, the granular approach provides a framework that can enable more objective ontology design—with both less subjectivity and more unique results.

In the granular approach, we introduce the concept of a fuzzy granule, which can provide a wide tolerance for the representation of an ontology domain. Fuzzy granules promote the flexibility and adaptation of the designed ontology.

Because the design pattern in the granular approach is general, it can be implemented in most ontology designs. The optimal

framework can enable designers to implement quantitative evaluation criteria and use computers to aid the optimal process automatically. Thus, our ontology design approach reduces the cost of creating ontologies and at the same time increases the efficiency of those ontologies.

## Acknowledgment

This research is based upon work supported in part by National Natural Science Foundation of China (61173123, 61379034), and Natural Science Foundation Project of Zhejiang Province (LR13F030003).

## References

- Bittner, T., & Smith, B. (2003). Granular spatio-temporal ontologies. In *AAAI symposium: Foundations and applications of spatio-temporal reasoning (FASTR 2003)* (pp. 12–17). AAAI Press.
- Calegari, S., & Ciucci, D. (2010). Granular computing applied to ontologies. *International Journal of Approximate Reasoning*, 51(4), 391–409.
- Chen, Q., Chen, Y.-P. P., & Zhang, C. (2007). Detecting inconsistency in biological molecular databases using ontologies. *Data Mining and Knowledge Discovery*, 15(2), 275–296.
- DiPietro, I., Pagliarecci, F., & Spalazzi, L. (2012). Model checking semantically annotated services. *IEEE Transactions on Software Engineering*, 38(3), 592–608.
- Ensan, F., & Du, W. (2008). Aspects of inconsistency resolution in modular ontologies. In *Canadian AI'08: Proceedings of the Canadian society for computational studies of intelligence, 21st conference on advances in artificial intelligence* (pp. 84–95). Berlin, Heidelberg: Springer-Verlag.
- Gruber, T. R. (1993). A translation approach to portable ontology specifications. *Knowledge Acquisition*, 5(2), 199–220.
- Guarino, N., & Welty, C. (2002). Evaluating ontological decisions with ontoclean. *Communications of the ACM*, 45(2), 61–65.
- Holsapple, C. W., & Joshi, K. D. (2002). A collaborative approach to ontology design. *Communications of the ACM*, 45(2), 42–47.
- Keet, C. M. (2010a). On the feasibility of description logic knowledge bases with rough concepts and vague instances. In *Description logics*.
- Keet, C. M. (2010b). Ontology engineering with rough concepts and instances. In *EKAU* (pp. 503–513).
- Lin, T. Y. (1997). Granular computing. In *Announcement of the BISC special interest group on granular computing*.
- Lin, T. Y. (2009). Granular computing: Practices, theories, and future directions. In *Encyclopedia of complexity and systems science* (pp. 4339–4355).
- Liu, Y., Jiang, Y., & Huang, L. (2010). Modeling complex architectures based on granular computing on ontology. *Transactions on Fuzzy Systems*, 18(3), 585–598.
- Liu, Y., Xu, C., Pan, Z., & Pan, Y. (2006a). Semantic modeling for ancient architecture of digital heritage. *Computers and Graphics*, 30(5), 800–814.
- Liu, Y., Xu, C., Zhang, Q., & Pan, Y. (2006b). Ontology based semantic modeling for chinese ancient architectures. In *IAAI'06: Proceedings of the 18th conference on innovative applications of artificial intelligence* (pp. 1808–1813). AAAI Press.
- Liu, Y., Xu, C., Zhang, Q., & Pan, Y. (2008). The smart architect: Scalable ontology-based modeling of ancient chinese architectures. *IEEE Intelligent Systems*, 23(1), 49–56.
- Liu, Y., Zhang, M., Jiang, Y., & Zhao, H. (2012a). Improving procedural modeling with semantics in digital architectural heritage. *Computers and Graphics*, 36(3), 178–184.
- Liu, Y., Zhang, M., Tang, F., Jiang, Y., Pan, Z., Liu, G., et al. (2012b). Constructing the virtual jing-hang grand canal with onto-draw. *Expert Systems With Applications*, 39(15), 12071–12084.
- Müller, P., Wonka, P., Haegler, S., Ulmer, A., & Van Gool, L. (2006). Procedural modeling of buildings. *ACM Transactions on Graphics (TOG)*, 25(3), 614–623.
- Pawlak, Z. (1991). *Rough sets: Theoretical aspects of reasoning about data*. Dordrecht: Kluwer Academic Publishing.
- Pührer, J., Heymans, S., & Eiter, T. (2010). Dealing with inconsistency when combining ontologies and rules using dl-programs. In *ESWC (1)* (pp. 183–197).
- Sure Staab, Y., Gomez-Perez, G.-P., Daelemans, W., Reinberger, M.-L., Guarino, N., & Noy, N. F. (2004). Why evaluate ontology technologies? because it works! *IEEE Intelligent Systems*, 19(4), 74–81.
- Wand, Y., & Weber, R. (1990). An ontological model of an information system. *IEEE Transactions on Software Engineering*, 16(11), 1282–1292 <http://dx.doi.org/10.1109/32.60316>.
- 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(1), 23–25.