

December 2004

A Fuzzy Description Logic Approach to Model Management in R&D Project Selection

Ou Liu

City University of Hong Kong

Qijia Tian

City University of Hong Kong

Jian Ma

City University of Hong Kong

Follow this and additional works at: <http://aisel.aisnet.org/pacis2004>

Recommended Citation

Liu, Ou; Tian, Qijia; and Ma, Jian, "A Fuzzy Description Logic Approach to Model Management in R&D Project Selection" (2004).
PACIS 2004 Proceedings. 6.
<http://aisel.aisnet.org/pacis2004/6>

This material is brought to you by the Pacific Asia Conference on Information Systems (PACIS) at AIS Electronic Library (AISeL). It has been accepted for inclusion in PACIS 2004 Proceedings by an authorized administrator of AIS Electronic Library (AISeL). For more information, please contact elibrary@aisnet.org.

A Fuzzy Description Logic Approach to Model Management in R&D Project Selection

Ou Liu, Qijian Tian, Jian Ma

Information System Department, City University of Hong Kong, Hong Kong, China
{isliuou, isqt, isjian }@cityu.edu.hk

Abstract

Project selection is an important task in R&D project management. Several mathematical and decision models have been proposed for R&D project selection and current efforts have been on the effective model management approaches. In order to provide a formal and semantically rich method for model representation and to support content-rich formal reasoning, this paper proposes a fuzzy description logic approach to model management in R&D project selection. In the proposed approach, ontology engineering is applied to design the architecture of model management system, and fuzzy description logic is proposed to represent the modeling knowledge. Thus it can represent the imprecise and vague information, which is common in R&D project selection. A decision support system has also been developed based on the proposed approach. It is used to facilitate the selection of project proposals in the National Natural Science Foundation of China (NSFC). The application results show that the proposed method supports effective model management for R&D project selection.

Keywords: Model Management, Description Logic, Fuzzy Theory, R&D Project Selection, Ontology

1. Introduction

R&D project selection is a very important task in organizations like government funding agencies, universities, research institutes, and technology-intensive companies. It is also a difficult task due to the nature that the future success of an R&D project can be hardly predicted. A number of mathematical decision models and methods (e.g. *Mathematical Programming & Optimization, Decision Analysis, Economic Models, and Interactive Method*) have been developed to help organizations make better decisions in R&D project selection (Ghasemzadeh & Archer, 2000; Henriksen & Traynor, 1999). However, current research findings indicate that many of the elaborated decision models and methods are not being used, and they have limited impacts on decision-makings for real-world project selection. In order to improve the usability of decision models and methods in real application, decision support systems (DSSs) have been proposed and developed to integrate decision models and methods with computer-based supports (Liberatore & Stylianou, 1995; Ghasemzadeh & Archer, 2000). They also combine the usage of data, models and knowledge rules at the organizational level (Tian et al., 2004).

However, few research addresses the problem of effective management of various decision models for project selection. Since there are many co-related decision models designed for the project selection process, it is important to manage them effectively and to facilitate the reuse of them in the DSSs for R&D project selection.

Several model management approaches have been proposed, including the relational approach (Blanning 1986, 1987), logic modeling approach (Bhargava & Kimbrough, 1993), structure modeling approach (Geoffrion, 1987, 1989, 1994), artificial intelligent approach (Liang, 1988, 1993; Liang & Konsynski, 1993), graph-based approach (Basu & Blanning,

1994, 1995, 1998), and object-oriented approaches (Huh, 1992; Ma, 1995). However, few of these approaches provide a formal and semantic-rich representation method that supports content-rich formal reasoning, such as a multi-criteria model retrieval task. Also, they can hardly represent imprecise and vague information commonly found in model management and project selection tasks.

This paper presents a fuzzy description logic approach to model management. The proposed methods and theories are applied in the selection of R&D projects at the National Natural Science Foundation of China (NSFC). Section 2 of this paper presents the research background. A model management approach based on fuzzy description logic is proposed in Section 3. Application result is presented in Section 4. Finally, a summary of the contribution is addressed in Section 5.

2. Research Background

NSFC (<http://www.nsf.gov.cn>) is the largest government funding agency in China with a primary aim to promote the basic research. One of the major tasks of NSFC is to select and fund R&D projects with great potential of scientific breakthrough or social impacts. The selection process in NSFC is carried out once in a year for most of its funding programs. In April 2004, NSFC received 34,820 project proposals. Five external reviewers were assigned to evaluate each proposal immediately after application deadline. The project selection process is coordinated by the top management and accomplished by the seven scientific departments as well as their divisions. The overall project selection task is assigned to departments, and then departments further assign their tasks to divisions. Division managers then invite and assign external reviewers and panel experts to evaluate the proposals. NSFC maintains a large database of external reviewers and panel experts from 69 disciplines.

During the basic stages of selection process (including proposal submission, selection of external reviewers, peer review, aggregation of review results, panel evaluation and final decision), a number of decision models are involved (Tian et al., 2004). For example, in the stage of review result aggregation, a model supporting the integration of subjective information with objective information is used (Ma et al., 1999; Zhang, 2002); and during panel evaluation, a group of highly related models for the uniformity and aggregation of preference information in multiple formats are applied for journal grading exercises (Zhou et al., 2002), which is a key criterion to assess the performance of projects supported by NSFC. From the experience of using and managing these decision models, we realize two important points:

- 1) Management of the decision models is far away from that of mathematical models, because they are highly related to the application domain of R&D project selection. In fact, this has been implied in previous research. For example, by structured modeling, Geoffrion suggested a “definitional system” based on domain entities for model management (Geoffrion, 1987); and in (Liang, 1993; Bhargava, 1995), researchers try to give definition to decision variables to differentiate them from mathematical variables.
- 2) Imprecise and vague information is very important for representing decision tasks and decision models in R&D project selection. The above introduced model examples for panel evaluation all involve vague information. This requires that the desired model management system (MMS) should support the representation of vagueness.

3. A Model Management Framework for R&D Project Selection

This section introduces a model management framework for R&D project selection based on ontology engineering and fuzzy description logic.

3.1 Ontology Engineering and Fuzzy Description Logic for Model Management

Ontology engineering involves methodologies to achieve knowledge sharing and reuse in distributed environments (Guarino, 1998). An ontology is “an explicit specification of a conceptualization” (Gruber, 1993), which defines the terminology of an application domain including the concepts inside the domain and the relationships between them. Its benefits include sharing a common understanding of information by providing precisely defined terms of relevant subject domains, and reusing knowledge by separating domain knowledge from operational knowledge.

We use the ontology engineering approach to design MMSs for its powerful support to domain definition and knowledge sharing, in order to solve the problems we identify before. A decision ontology is designed to conceptualize the knowledge for decision making processes, which includes not only the knowledge of the decision models, but also the meta-knowledge about them. It can be divided into two parts, i.e. domain ontology and modeling ontology. The latter is further divided into the following three parts: the domain-related variables (*decision variables*), the domain-independent mathematical models (*model types*) and how these models can be applied in the domains (*model templates*).

In order to represent decision ontology in a formal and semantic-rich way, Fuzzy Description Logic (FDL) is proposed and used. Description logic (Baader et al., 2003) is a unified formalism of several knowledge representation methods such as object-orientation, frame-based systems and semantic networks. FDL extends the standard description logic (Schmidt-Schau & Smolka, 1991) with fuzzy set theory, and is suitable to represent various kinds of information in decision ontology. FDL provides both semantically rich representation and powerful formal reasoning services for model management.

3.2 The Conceptual Architecture of the MMS

The conceptual architecture of the proposed model management system is shown in Figure 1. It contains a decision ontology, an algorithm library, an adapter and a user interface.

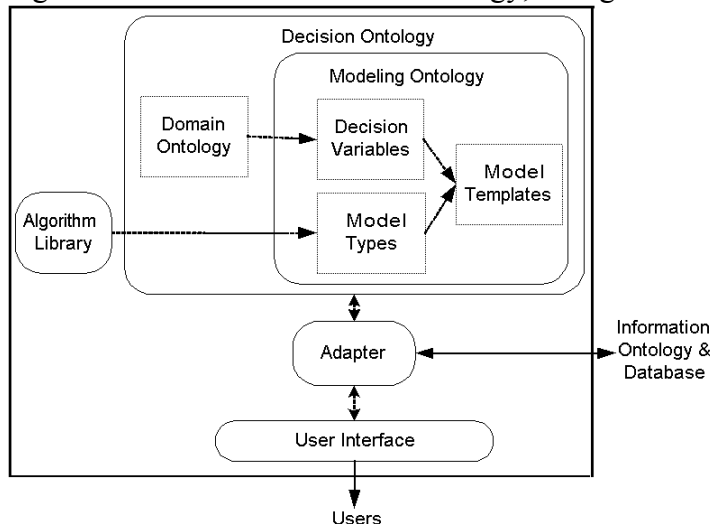


Figure 1. The Conceptual Architecture of the MMS

The decision ontology conceptualizes the knowledge and meta-knowledge of decision models. It provides the knowledge related to decision models, rather than the model instances and algorithms for model solving. To instantiate and solve models, firstly, the adapter has to retrieve the model templates from decision ontology and data from the information ontology and database component. The adapter then combines them to formulate model instances, and call on the algorithms from the algorithm library to solve these model instances.

The decision ontology is the main component of the architecture. It consists of two parts: domain ontology and modeling ontology. The domain ontology refers to the terminology for decision-making, defining the concepts and terms used in the decision-making processes of a domain. It mainly includes the terminology from the information ontology (data schema), and some collective concepts such as *Proposals* (meaning a set of proposals). The modeling ontology is further divided into the following three categories of knowledge:

- Domain-related variables (*decision variables*). A decision variable usually appears in the form of a feature of a domain concept specified with a dimension and a unit. It can be instantiated with a mathematical value.
- Domain-independent mathematical models (*model types*). A model type is an interface of some algorithm stored in computers (Mannino et al., 1990; Hong et al., 1993, 1995). It specifies a relation among relevant *mathematical variables*.
- And the application of model types in the domains (*model templates*). Model templates describe how model types can be applied in specific decision-making tasks (Hong et al., 1993, 1995). To obtain a model template, model types are instantiated using *decision variables*.

3.3 The Proposed Fuzzy Description Logic for Model Representation

To represent decision ontology formally and semantically is a key to model sharing and reuse. Fuzzy Description Logic (FDL) is proposed for this purpose. It is an extension to the standard description logic (Schmidt-Schau & Smolka, 1991) with fuzzy set theory (Zadeh, 1965).

To distinguish decision variable and mathematical variables (Mannino et al., 1990; Bhargava, 1991; Liang, 1993; Holsapple & Whinston, 1996), the proposed logic introduces so-called abstract domains (problem domains) and concrete domains (mathematical knowledge) (Baader & Hanschke, 1991; Baader & Sattler, 1998).

An abstract domain focuses on a description of what individuals (objects) and their relationships existing in the problem domain. It is usually summarized on the conceptual level. Concrete domains are used to further characterize objects and their quantitative relationships in abstract domains on the detailed level. They are bridged through so-called *features*, which are functions assign individuals of abstract domains values in concrete domains.

An abstract domain is modeled in terms of their individuals as well as the concepts, roles (relations) and features defined on them. A concept is an abstract of a set of individuals with similar characteristics. Roles are binary relations between individuals. Features are partial functions defined on the abstract domain. Features can be classified into abstract features and concrete features according to their values in the abstract domain or a concrete domain. Because abstract features can be reduced into roles (so called functional roles (Baader et al., 2003)), we only consider concrete feature here.

In decision-making modeling, features are used to capture the measurable properties of individuals, particularly the numerical properties such as the supply-capacity of Plant. They are functions from a set of individuals to certain numerical domains, such as real numbers and integers. These well-developed and reusable numerical domains are called *concrete domains*.

A concrete domain could be well-known mathematical structures such as real numbers and integers, which have considerable complex structures. It also could be just a set of possible values defined by decision-makers without any additional structures. An example of concrete domains is the set of real numbers $\mathbb{R} = (\mathbb{R}; >, \geq, <, \leq, =, \neq, \{>_x\}_{x \in \mathbb{R}}, \{\geq_x\}_{x \in \mathbb{R}},$

$\{<_r\}_{r \in \mathbf{R}}, \{\leq_r\}_{r \in \mathbf{R}}\}$ is the concrete domain used most commonly in decision analysis, where $>_r \triangleq \{s \in \mathbf{R} \mid s >_r\}$ and $\geq_r, <_r$ and \leq_r can be defined in the similar way.

The semantics for FDL is based-on an *interpretation* $\mathcal{I} = (\Delta^{\mathcal{I}}, \bullet^{\mathcal{I}})$, where $\Delta^{\mathcal{I}}$ is the universe of the interpretation, and $\bullet^{\mathcal{I}}$ is the interpretation function over such a universe. It is different from the usual notation of interpretation for the import of fuzzy constructs. In the fuzzy interpretation, $\bullet^{\mathcal{I}}$ is a function that maps each concept C into a membership function $C^{\mathcal{I}}: \Delta^{\mathcal{I}} \rightarrow [0, 1]$; and each role R into a membership function $R^{\mathcal{I}}: \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}} \rightarrow [0, 1]$.

$C^{\mathcal{I}}$ works as the *membership degree function* of the fuzzy concept C , i.e. for any object of the domain $d \in \Delta^{\mathcal{I}}$, $C^{\mathcal{I}}(d)$ is the degree of being an element of the fuzzy concept C under the interpretation \mathcal{I} . As for roles, the semantics is given similarly. The conjunction, disjunction, negation of fuzzy set can be easily applied to the interpretation of complex concept (Tresp & Molitor, 1998; Straccia, 2001).

Please note that the fuzzy interpretation is reduced into the normal crisp case when all membership degrees of concepts and roles are set as 1.

Furthermore, a fuzzy concept can also be specified explicitly with the membership function defined on concrete features of another concept. For example, based on the evaluation grade of a project, we can define to what degree it is a good project.

Like other kinds of description logic, the knowledge base of FDL includes two parts, called terminology (TBox) and world description (ABox) (Baader et al., 2003). TBox is used to conceptualize the domain world, while ABox to describe the instances of those concepts in TBox. Their relationship is similar to that between classes and objects in object-orientation.

A *fuzzy terminological axiom* in TBox is either a fuzzy concept specialization or a fuzzy concept definition. A *fuzzy concept specialization* is an expression of the form $A \sqsubseteq C$, and a *fuzzy concept definition* is an expression of the form $A = C$, where A is a primitive concept and C is a concept. $A \sqsubseteq C$ iff. $\forall d \in \Delta^{\mathcal{I}}, A^{\mathcal{I}}(d) \leq C^{\mathcal{I}}(d)$, whereas $A = C$ iff. $\forall d \in \Delta^{\mathcal{I}}, A^{\mathcal{I}}(d) = C^{\mathcal{I}}(d)$.

A *fuzzy assertion* in ABox is an expression having one of the following form $\langle \alpha \geq n \rangle$ or $\langle \alpha \leq m \rangle$, where α is a crisp assertion (like in ALC (Schmidt and Smolka, 1991)), $n \in (0, 1)$ and $m \in [0, 1]$. Semantically, a fuzzy assertion $\langle \alpha \geq n \rangle$ constrains the truth-value of α to be less or equal to n (similarly for $\langle \alpha \leq m \rangle$).

Just like in crisp DL, reasoning services for FDL usually deal with the problems of satisfiability, subsumption, equivalence and disjointness. In fact, all these can be reduced to deciding satisfiability problem. Since the fuzzy construct only affect the interpretation and A-Box, the reasoning algorithm about T-Box in the proposed fuzzy DL can be adapted from that of crisp DL, i.e., Tableau-based algorithm (Schmidt and Smolka, 1991).

Based on the proposed FDL, we can describe the decision ontology formally. In next two sections, FDL will be applied for the representation of domain ontology and modeling ontology for model management in R&D project selection respectively.

3.4 Domain Ontology for R&D Project Selection

A very simplified terminology of Domain ontology for R&D project selection is illustrated as follows:

Concepts

- User = Int-User \sqcup Ext-User
- Int-User = Top-Manager \sqcup Dept-Manager \sqcup Div-Manager

- $\text{Ext-user} = \text{Panel-Expert} \sqcup \text{Ext-Reviewer} \sqcup \text{P-Coordinator} \sqcup \text{Org-User}$
- $\text{P-Coordinator} = \text{P-Investigator} \sqcup \text{Participant}$
- $\text{Project} = \text{On-Going-Project} \sqcup \text{Completed-Project}$
- Proposal
- $\text{Good-Project} \sqsubseteq \text{Completed-Project}$
- $\text{Good-PI} \sqsubseteq \text{P-Investigator} \sqcap \exists \text{participateIn}.\text{Good-Project}$

Roles

- participateIn : P-Coordinator can participate in Project and Proposal.
- reviewedBy : Proposal is reviewed by Ext-Reviewer.
- match : A fuzzy role, represent how well an Ext-Reviewer matches with a proposal.

Features

- name : Proposal, Project and User have their own names.
- age : Each User has an age.
- keywords : Proposal, Project, Panel-Expert, and Ext-Reviewer have their own keyword list to indicate the subject areas they fall in.
- p-invstg : Proposal and Project has an P-Investigator.
- num-reviewed : Proposal is reviewed by a number of external reviewers.
- num-review : Ext-reviewer review a number of proposals.
- sbj-grade : Each individual of Proposal has a subjective grade.
- obj-grade : Each individual of Proposal has an objective grade.
- grade : Each individual of Proposal has a overall grade.

In this fragment of terminology, we define a classification of users (staff inside and outside the organization), proposals, projects, as well as their relationship and properties/features. The meanings of the concepts, roles and features are easy to understand by name and the explanation. The terminology can be easily extended to include more complex concepts by FDL.

Example 1. The following are samples of complex concepts defined by the logic.

(1) The concept of all applicants with age less than 35 can be defined as

$$C_1 = \text{Applicant} \sqcap \exists (\text{age}) . <_{35}.$$

(2) The concept of reviewers who review more than 10 proposals can be defined as

$$C_2 = \text{Reviewer} \sqcap \exists (\text{num-review}) . >_{10}.$$

(3) All proposal whose principal investigator is less than 35 years old can be described as

$$C_3 = \text{Proposal} \sqcap \exists \text{participateIn}^{-1}. (C_1 \sqcap \text{P-Investigator}).$$

(4) All proposals as a concept: $\text{Proposals} = \sigma(\text{Proposal})$.

Constraints can be easily set by the logic.

Example 2. The constraint “Each proposal should be reviewed by at least five reviewers.” can be represented as: $\text{Proposal} \sqcap (\exists (\text{num-reviewed}) . <_5) = \perp$.

One thing to be noticed is that some vague information required by R&D project selection can be easily presented, which is hard in previous approaches. We illustrate it by examples in the following.

Example 3. After a project terminate, project evaluation will be done. Investigator of good project will be more possible to be granted a new project. Here, we define fuzzy concepts for the requirement. Good-Project is a fuzzy concept. $\text{Good-Project} \leq \text{Project}$. We may have project A that $< \text{Good-Project}(A), 0.8 >$, so that we can say A is likely to be a good project with a truth value 0.8.

Principal Investigator (PI) who leads a good-Project will be a Good-PI. It is represented by

$\text{Good-PI} \sqsubseteq \text{P-Investigator} \sqcap \exists \text{participateIn.Good-Project}.$

Project selection can be regarded as a (fuzzy) MCDM problem (Zhou, 2000; Zhang, 2002). As a criterion, a proposal from Good-PI will give a plus to the obj-grade. It can be implemented by using the truth-value ($\text{Good-PI}^1(A)$) for project A) as an element of the decision matrix.

Furthermore, Fuzzy concepts can be defined based on concrete domain.

Example 4. Good-Project can be defined by $A = \mu_{\mu_A}(\text{grade})$, where *grade* is the standardized overall grade of the project in the post-project evaluation. Supposed *grade* falls [0,1] after standardization, we can use it as the membership degree of a Completed-Project to be a Good-Project directly.

Example 5. The role “Match” is a fuzzy role between Ext-Reviewer and proposal, represent how well an Ext-Reviewer matches with a proposal. We will know how it can be used to define a decision variable in next section.

3.5 Modeling Ontology for R&D Project Selection

As presented in conceptual architecture (Fig.1), the modeling ontology includes the knowledge about decision variables, modeling types and model templates. We define them as concepts in FDL, and draw out roles (relations) among them, so that reasoning can be applied into the modeling ontology.

In most of the previous researches on model management, decision variables are represented as a kind of mathematical variables, which can be characterized by their names and value types. However, it has been noticed by several researches (Mannino et al., 1990; Bhargava, 1991; Liang, 1993; Holsapple & Whinston, 1996) that decision variables have finer structures than mathematical variables, which is important for model management. Semantic information, which defines the meanings of decision variables, should be added to the representations of decision variables, in addition to their names and value types represented in mathematical variables.

We derive decision variables from domain ontology so as to give them semantic information. Three types of decision variables are considered:

- 1) Concept features. Representing domain ontology with FDL, the semantic information of the decision variable has been given by the specific *feature* of the specific *concept*.
- 2) Concept truth-values. The truth-value of an individual belonging to a fuzzy concept is an important source of input information for decision-making. This is very different

from decision variables in previous researches. It comes naturally from the rich representation capability of FDL.

- 3) Role truth-values. Similar to concept truth-values, they refer to the truth-value of the role between two concept individuals.

Definition 1. Decision variables as a whole is defined as a specific concept named DV. There are three sub-concepts subsuming it: FDV, CDV and RDV, corresponding to concept features, concept truth-values and role truth-values respectively. DV has two features “type” and “name”; CDV has “cName”; RDV has “rName”; while FDV has four features: <cName, fName, dimension, unit>. These features have the following meaning respectively:

- type: the decision variable type, one of the string values: “Cfeature”, “Ctruth” and “Rtruth”;
- name: the name of the decision variable;
- cName: concept name, to which the decision variable is related;
- rName: role name, to which the decision variable is related;
- fName: feature name, to which the decision variable is related;
- dimension: the feature’s dimension;
- unit: the feature’s measurement unit.

The reason for using *dimension* here is that it is more important for matching two variables when they represent the same kind of things (e.g., length, or time), rather than when they use the same unit or when they are in the same co-domain. More verification of using dimension can be found in (Bhargava et al., 1991). The *unit* is also used here for a finer-granularity. There is a special value *nil* for both dimension and unit. It is used for variables which do not have a unit such as the variable “the number of people in the room”. For further treatments to unit see (Gruber & Olsen, 1994).

For example, as a decision variable, the age of a principal investigator can be represented by a concept individual of FDV with its features as following:

- name: PI-Age,
- cName: P-Investigator,
- fName: Age,
- dimension: time,
- unit: year.

It means that the decision variable named “PI-Age” represents the feature “Age” of the concept “P-Investigator” with dimension as time and measured in years.

Our framework divides representations of decision models into model type, model template, and model instance. Model type represents the mathematical aspect of a decision models, and it describes the measurable quantities in a particular setting (both constants and variables) and their relationships. Model template describes how a model type can be applied in a class of circumstances. Model instance is an instantiation of some model template for a specific problem. Therefore, it achieves one of the main objectives of model management, i.e., “model-data independence” and “model-solver independence” (Geoffrion, 1987; Muhanna & Pick, 1994; D. Dolk & Kottmann, 1993)

Definition 2. A model type is defined as $mt_y = ((x_1, \dots, x_s); (y_1, \dots, y_t); constraints)$, where x_i and y_j are mathematical variables. (x_1, \dots, x_s) is the input of the model type and denoted as $INPUT(mt_y)$. (y_1, \dots, y_t) is the output of the model type and denoted as $OUTPUT(mt_y)$. $Constraints$ is a set of formula like $x_i \in T_{Z^+}$ and $y_i \in T_{Z^+}$. (Assume all constraints are written in the form $x \in T_D$ for some concrete domain D). On the other hand,

model types as a whole is defined as a specific concept named MTY, with each model type as an individual of the concept in the interpretation.

Definition 3. Model templates as a whole is defined as a specific concept name MTE, which have three features: $\langle \text{mt}_y, \mathbf{X}, \mathbf{Y} \rangle$. They are all abstract features. mt_y is the corresponding model type, i.e. $\text{codom}(\text{mt}_y) = \text{MTY}$. And $\text{codom}(\mathbf{X}) = \text{codom}(\mathbf{Y}) = \sigma(\text{DV})$, i.e., a set of decision variables. Thereinto, \mathbf{X} refers to input and is denoted as $\text{INPUT}(\text{mt}_y)$, while \mathbf{Y} refers to output and is denoted as $\text{OUTPUT}(\text{mt}_y)$. Both $\mathbf{X} = (X_1, \dots, X_k)$ ($k = |\text{INPUT}(\text{mt}_y)|$) and $\mathbf{Y} = (Y_1, \dots, Y_l)$ ($l = |\text{OUTPUT}(\text{mt}_y)|$) are a vector of decision variables.

The benefit of defining model type and model template as a specific concept is to make the decision ontology for model base uniformed with domain terminology in fuzzy DL, and so as to make the reasoning service of fuzzy DL applicable to the whole decision ontology.

The relations among decision variables and decision models are the key to reasoning about decision models. In decision ontology, we define three roles:

- “About” between two concepts, “MTE” (model template) and “DV” (decision variable), We may have an assertion $\langle \text{About}(M, X), 0.8 \rangle$ in the knowledge base Σ , meaning that the model template M is relevant to the decision variable X by a membership degree (truth-value) 0.8.
- “Related” between two “DV”s. $\text{Related}(X, Y)$ represents a relation that we can get Y wholly or partly based on X . Note that “Related” is not a symmetric role, i.e., $\text{Related}(X, Y)$ doesn’t conclude $\text{Related}(Y, X)$.
- “Call” between “MTE” and “MTY”, meaning that a model template uses a model type.

Thus, we can summary the TBox of the modeling ontology as following:

Concepts

- $\text{DV} \sqsubseteq \text{T}$
- $\text{FDV} \sqsubseteq \text{DV}$
- $\text{CDV} \sqsubseteq \text{DV}$
- $\text{RDV} \sqsubseteq \text{DV}$
- $\text{MTY} \sqsubseteq \text{T}$
- $\text{MTE} \sqsubseteq (\exists \text{Call}.\text{T}) \sqcap (\forall \text{Call}.\text{MTY}) \sqcap (\exists_{\geq 1} \text{About}.\text{T}) \sqcap (\forall \text{About}.\text{DV})$

Roles

- About: A model template is about a decision variable.
- Call: A model template uses a model type.
- Related: A decision variable is related to another one.

Features

- MTE has three features: $\text{mt}_y, \mathbf{X}, \mathbf{Y}$ (see Definition 11 for details)
- DV, FDV, CDV, RDV have relevant features respectively (see Definition 10 for details)

When applied to manage the decision models for R&D project selection, the number of assertions in the knowledge base is very large although the structure of the terminology is simple. We will show the representation of the modeling ontology by two examples, with one

for the concepts and features (including imprecise information) and the other for roles and reasoning.

Example 6. The ABox of a model for matching between external reviewers and proposals is shown below:

DV:

prp-kwds := < Cfeature, Proposal, keywords >

rvw-kwds := < Cfeature, Ext-Reviewer, keywords >

prp-rvw-match := < Rtruth, Match >

MTE:

mte_fuzzyMatching

MTY:

mt_y_fuzzyMatching := < mte_fuzzyMatching, { σ (prp-kwds), σ (rvw-kwds)}, { σ (prp-rvw-match)} >

The model does matching between keywords of proposals and reviewers, and produces matching values. In this example, { σ (prp-rvw-match)} as output with “prp-rvw-match := <Match>” means that the output is a matrix whose elements are truth values of instances of the role “Match”.

Example 7. In R&D project selection, suppose we have had a model named “Int-sub-obj” (Zhou, 2000; Zhang, 2002) for integrating subjective grade and objective grade into an overall grade. Then there are at least the following assertions in ABox (for simplicity, the detailed features of decision variables are omitted here):

<About (Int-sub-obj, prp-ob-grade), 0.8>,
 <About (Int-sub-obj, prp-sb-grade), 0.8>,
 <About (Int-sub-obj, prp-grade), 0.9>,
 <Related (prp-ob-grade, prp-grade), 0.9>,
 <Related (prp-sb-grade, prp-grade), 0.9>.

If now adding a model name “Avg-prps” for calculating the average grade of a set of proposals, then the following assertions will be added into ABox:

<About (Avg-prps, prp-grade), 0.8>,
 <About (Avg-prps, prps-avg-grade), 0.9>,
 <Related (prp-grade, prps-avg-grade), 0.9>.

Furthermore, according to a reasoning algorithm, the following can also be concluded:

<Related (prp-ob-grade, prps-avg-grade), 0.81>
 <Related (prp-sb-grade, prps-avg-grade), 0.81>

which represent indirect relations between decision variables through more than one decision models.

The meta-knowledge of the model base has been described by the A-box of the decision ontology, which builds up a foundation for reasoning about the decision models. With the fuzzy role “About” and “Related” defined, we can retrieve decision variables or decision models according to users’ vague requirements in the order of match degree. The problems include: “I want to search for a model that is about the grade of a proposal”; “In order to calculate the average grade of a set of proposals, what information is needed beforehand?”; et al. All the answers can be an ordered list according to the relevant degree.

For example, to answer the second question, the model management system will search the decision ontology for the form <Related (X, prps-avg-grade), w>, and get the result including at least:

<Related (prp-grade, prps-avg-grade), 0.9>,
 <Related (prp-ob-grade, prps-avg-grade), 0.81>
 <Related (prp-sb-grade, prps-avg-grade), 0.81>

Then, it will tell the user that it needs to know the overall, subjective, and objective grade of the proposals beforehand, with a ranking based on the relevant degree. The algorithms for building up ABox and reasoning will be presented in a separated article.

To summarize, Section 3.4 and 3.5 have shown the formal representation of domain ontology and modeling ontology in FDL respectively. They form a solid foundation for sharing and reusing the decision resources for R&D project selection.

4. Application of the Framework to R&D Project Selection in NSFC

Within the Internet-based Science Information System (ISIS: <http://isis.nsfc.gov.cn>), a DSS is developed to support R&D project selection in NSFC. The proposed framework has been applied for the model management in ISIS.

The project selection process in NSFC can be divided into six stages or tasks (Tian et al., 2004): proposal submission, selection of external reviewers, peer review, aggregation of review results, panel evaluation and final decision, during which a number of decision models are involved. Table 1 shows major decision models in these tasks.

Table 1. Major decision models in R&D project selection tasks

Task name	Decision Models
1. Proposal submission	Proposal validation PI duplication checking
2. Selection of external reviewers	Fuzzy matching between experts and proposals Assignment of external reviewers
3. Peer review	Review result validation
4. Aggregation of review results	Aggregation of subjective and objective information
5. Panel evaluation	Aggregation of multiple preference formats
6. Final decision	Project distribution checking

In some of these decision models, there are many sub models for supporting. For example, as stated before, there are several models cooperating for the aggregation of multiple preference formats.

Based on the requirements, we define the decision ontology. A short list of part of the elements in the modeling ontology for R&D project selection is shown in Table 2.

Table 2. Decision Ontology for R&D Project Selection

Decision Variables: Prp-grade Prp-sb-grade Prp-ob-grade Prp-kwds Rvw-kwds ...	Model Types: Fuzzy-Model-for-Fitness-Matrix Job-Assignment Model MCDM-Aggregation Preference-Convert-Model Subjective-Objective-Aggregate-Model ...
---	---

Model Templates:

Proposals-Experts-Fitness
Proposals-Assign-to-Experts
Proposal-Evaluation-Aggregation
Panel-Experts-Evaluation-Aggregation
Distribution-Analysis-for-Final-Decision
...

Major model management functions provided in ISIS are as follows.

- Editing the domain ontology: End-users can add new concepts, modify or delete an existing one, and re-organize concepts. This function enables the reuse of existing concepts for different purposes and also facilitates the customization of existing concepts for particular decision-making situations.
- Maintain decision variables: decision variables can be added, modified and related to information in the domain ontology.
- Maintain model types: Existing model types can be modified and deleted. New ones can be added.
- Maintain model templates: The bidding of model types with concepts can be changed using this function. In other word, this function enables end-users to define their usages of model types for their own decision situations.
- Retrieve model templates: End-users can search required model templates semantically using concepts. Multiple search methods are possible which provide flexible ways to find the exact models end-users intend to find.

During the exercises of project selection through ISIS in 2001, 2002 and 2003, there are 13, 37 and 52 (all) divisions in NSFC involved respectively. Results of the utilization and management of decision models are very positive, especially the success rate of electronic peer review has reached 96% in 2003.

5. Summary

A model management approach based on ontology engineering and fuzzy description logic is presented for R&D project selection. It includes a conceptual architecture based on the decision ontology. The decision ontology conceptualizes terminologies commonly used in the decision-making process, and it is formalized by a semantic-rich language FDL. It also specifies the relations between decision models and decision variables, as well as that between decision variables and domain knowledge. The theoretical results have been applied in the DSS for selection of R&D projects in NSFC. There are many advantages of the proposed model management approach: 1) Decision ontology is defined by FDL, thus it provides a foundation to develop a unified representation of decision resources, and to represent and handle imprecise information in model management. 2) Decision ontology has the properties of sharable, reusable and interoperable across organization hierarchies. It can be accumulated as organization knowledge for long-term utilization. 3) From the perspective of end-users, decision ontology also provides a semantically rich device to manipulate models. Thus decision resources are represented using their own terminologies/concepts and organized in their familiar ways.

References:

Baader, F., McGuinness, D., Nardi, D., and Schneider, P.P. (eds.) *The Description Logic Handbook: theory, implementation and applications* Cambridge University Press, 2003.

- Basu, A., and Blanning, R.W. "Model integration using metagraphs," *Information Systems Research* (5:3) 1994, pp 195-218.
- Basu, A., and Blanning, R.W. "Metagraphs," *Omega* (23:1) 1995, pp 13-25.
- Basu, A., and Blanning, R.W. "The analysis of assumptions in model bases using metagraphs," *Management Science* (44:7) 1998, pp 982-995.
- Blanning, R.W. "An entity-relationship approach to model management," *Decision Support Systems* (2:1) 1986, pp 65-72.
- Blanning, R.W. "A relational theory of model management," in: *Decision Support Systems: Theory and Applications*, C.W. Holsapple and A.B. Winston (eds.), Springer, 1987.
- Bhargava, H.K., Kimbrough, S., and Krishnan, R. "Unique names violations: A problem for model integration," *ORSA Journal of Computing* (3:2) 1991, pp 107-120.
- Bhargava, H.K., and Kimbrough, S.O. "Model management: An embedded languages approach," *Decision Support Systems* (10) 1993, pp 277-299.
- Geoffrion, A.M. "An introduction to structured modeling," *Management Science* (33:5) 1987, pp 547-588.
- Geoffrion, A.M. "The formal aspects of structured modeling," *Operations Research* (37:1) 1989, pp 30-51.
- Geoffrion, A.M. "Structured Modeling: survey and future directions," *ORSA CSTS Newsletter* (15:1) 1994.
- Ghasemzadeh, F., and Archer, N.P. "Project portfolio selection through decision support," *Decision Support Systems* (29) 2000, pp 73-88.
- Gruber, T.R. "A translation approach to portable ontologies," *Knowledge Acquisition* (5:2) 1993, pp 199-220.
- Guarino, N. "Formal ontology and information systems," in: *Formal ontology in information systems*, N. Guarino (ed.), IOS Press, Amsterdam Netherlands, 1998, pp. 3-15.
- Henriksen, A.D., and Traynor, A.J. "A practical R&D project-selection scoring tool," *IEEE Transactions on Engineering Management* (46:2) 1999, pp 158-170.
- Holsapple, C.W., and Winston, A.B. *Decision Support Systems: A Knowledge-Based Approach* West Publishing St. Paul MN, 1996.
- Liang, T.P. "Model management for group decision support," *MIS Quarterly* (4) 1988, pp 667-680.
- Hong, S.N., Mannino, M.V., and Greenberg, B. "Measurement theoretic representation of large, diverse model base: the unified modeling language Lu," *Decision Support Systems* (10) 1993, pp 319-340.
- Hong, S.N., and Mannino, M.V. "Formal semantics of the unified modeling language Lu," *Decision Support Systems* (13) 1995, pp 263-293.
- Huh, S.Y. "Modelbase construction with object oriented constructs," *Decision Science* (24) 1992, pp 409-434.
- Liang, T.P. "Analogical reasoning and case-based learning in model management," *Decision Support Systems* (10) 1993, pp 137-160.
- Liang, T.P., and Konsynski, B. "Modelling by analogy: use of analogical reasoning in model management systems," *Decision Support Systems* (9) 1993, pp 113-125.
- Liberatore, M., and Stylianou, A. "Expert support systems for new product development decision making: Modeling framework and applications," *Management Science* (41:2) 1995, pp 1296-1316.
- Ma, J. "An object-oriented framework for model management," *Decision Support Systems* (13) 1995, pp 133-139.
- Ma, J., Fan, Z.P., and Huang, L.H. "A Subjective and Objective Integrated Approach to Determine Attribute Weights," *European Journal of Operational Research* (112:2) 1999, pp 397-404.

- Mannino, M.V., Greenberg, B.S., and Hong, S.N. "Model libraries: knowledge representation and reasoning," *ORSA Journal of Computing* (2) 1990, pp 287-301.
- Muhanna, W.A., and Pick, R.A. "Meta-modeling concepts and tools for model management: A systems approach," *Management Science* (40:9) 1994, pp 1093-1123.
- Schmidt-Schau, M., and Smolka, G. "Attributive concept descriptions with complements," *Artificial Intelligence* (48:1) 1991, pp 1-26.
- Straccia, U. "Reasoning within Fuzzy Description Logics," *Journal of Artificial Intelligence Research* (14) 2001, pp 137-166.
- Tian, Q.J., Ma, J., Liang, J.Z., Kwok, C.W., and Liu, O. "An organizational decision support system for effective R&D project selection," *Decision Support Systems* (to be published) 2004.
- Tresp, C., and Molitor, R. "A description logic for vague knowledge," Proc. of the 13th European Conf. on Artificial Intelligence (ECAI-98), Brighton (England), 1998.
- Zadeh, L.A. "Fuzzy sets," *Information and Control* (8) 1965, pp 338-353.
- Zhang, Q. "Web-based Group Decision Support Based on Integrating Subjective and Objective Information," PhD Dissertation, City University of Hong Kong, 2002.
- Zhou, D.N. "Fuzzy Group Decision Support System Approach to Group Decision Making under Multi-Criteria," PhD Dissertation, City University of Hong Kong, 2000.
- Zhou, D.N., Ma, J., Turban, E., and Bolloju, N. "A fuzzy set approach for evaluating grades of journals," *Journal of Fuzzy Sets and Systems* (131) 2002, pp 63-74.