带函数的描述逻辑

丛晓青[1,2],曹存根[1],眭跃飞[1]
CONG Xiao-qing[1,2],CAO Cun-gen[1],SUI Yue-fei[1]

1.中国科学院 计算技术研究所 智能信息处理重点实验室,北京 100081
2.中国科学院 研究生院,北京 100080
1.Key Lab of Intelligent Information Processing,Institute of Computing Technology,CAS,Beijing 100081,China
2.Graduate University of Chinese Academy of Sciences,Beijing 100080,China
E-mail:congxiaoqing05@mails.gucas.ac.cn

CONG Xiao-qing,CAO Cun-gen,SUI Yue-fei.Description logics with functions.Computer Engineering and Applications,2008,44(22):46-50.

Abstract: Description logics are fragments of the first-order logic,which contain concepts,roles,and concept and role constructors;and have features of strong expressivity and decidable reasoning.But the existing description logics cannot express functions on concepts and roles so that some functions in the conceptual model of libraries cannot be expressed,such as the number of books,the number of borrowed books,and so on.This paper introduces a description logic with functions.Firstly the unexpressed functions are analyzed in the conceptual model of libraries using the existing description logics and a solution is proposed.Secondly the syntax and semantics of the description logic with functions are presented.Finally the description logics with functions to the conceptual model of libraries is applied.

Key words: description logics;role;concept;individual;expressiveness

摘 要:描述逻辑是包含了概念、角色以及概念和角色构造子的一阶逻辑的子逻辑,具有表达能力强且推理可判定的特征。但现有描述逻辑无法表示概念和角色上的函数,因此在图书馆的概念模型中,有一些问题便不能表示,如图书的本数、借书的条目数等。在现有描述逻辑基础上引入函数来解决这个问题。首先分析现有描述逻辑在图书馆的概念模型中不能表示的一些问题并提出解决方法,然后给出带函数的描述逻辑的语法和形式语义,最后用带函数的描述逻辑形式化表示图书馆概念模型中的一些实际问题。

关键词:描述逻辑;角色;概念;个体;表达能力
DOI:10.3778/j.issn.1002-8331.2008.22.013 **文章编号**:1002-8331(2008)22-0046-05 **文献标识码**:A **中图分类号**:TP182

1 简介

描述逻辑是包含了概念、角色以及概念和角色构造子的一阶逻辑的子逻辑,其概念类似于一阶逻辑中的一元谓词,角色类似于一阶逻辑中的二元谓词,由构造子形成复杂的概念和角色[1,5-7]。它具有的重要特征是表达能力强且提供可判定的推理服务[7,9]。不同的构造子集合决定不同的描述逻辑,构造子越多,往往描述逻辑的表达能力越强,但也会使得推理越复杂;构造子越少,推理过程越简单,但也可能使得表达能力太弱[3,4,7,9]。使用描述逻辑表示领域的概念模型, 概念用来表示领域中的术语,通过概念之间的包含关系形成领域的层次结构。角色用来表示领域中两个个体对象之间的关系,如借书这个性质可以用一个角色来表示,表示一个读者与一本书之间的关系[2,7]。通过概念和角色,可以描述领域中个体的性质:个体与概念之间的事例关系,个体与另外一个个体之间的关系,还可以描述概念之间的等价,包含与不交的关系[7]。

但是现有的描述逻辑不能表示概念和角色上的函数,因此在实际的应用中还有很多问题不能表示。以图书馆的概念模型[2]为例,使用现有的描述逻辑不能表示如下问题:书和员工都是图书馆的资源,由某某员工管理书,由某某员工管理读者,图书馆中图书的本数,读者的人数,借书是图书馆提供的服务之一,借书的条目数等。上述问题就算引入复杂的概念和角色构造子[3]或将描述逻辑扩展至非经典形式[8]也无法很好地表示。本文在描述逻辑中引入函数的概念便很好地解决了这方面的问题。

函数是关系的一种,一个集合 S 上的二元关系 f 是集合 $S\times S$ 的一个子集。一个 S 上的二元关系 $f\subseteq S\times S$ 如果满足条件:对

基金项目:国家自然科学基金(the National Natural Science Foundation of China under Grant No.60373042,No.60273019,No.60073017);国家重点基础研究发展规划(973)(the National Grand Fundamental Research 973 Program of China under Grant No.G1999032701);国家科技部项目(the National Ministry of Science and Technology programme under Grant No.2001CCA03000)。
作者简介:丛晓青(1983-),女,硕士生,主要研究方向:人工智能、描述逻辑;曹存根(1964-),男,博士,研究员,主要研究方向:人工智能;眭跃飞(1963-),男,博士,研究员,主要研究方向:人工智能。
收稿日期:2008-01-09 **修回日期**:2008-04-08

任意的 $x,y,z\in S$，如果 $(x,y)\in f$ 并且 $(x,z)\in f$，则 $y=z$，那么 f 是一个函数。在描述逻辑中，角色是用来表示关系的，但是角色表示的只是个体与个体之间的关系[2,7]，而引入的函数却不仅仅是个体与个体之间的关系，还是概念上的函数或角色上的函数，表示的是概念与个体或角色与个体之间的关系。这一类的函数是不能用角色来表示的，给出带这种函数的描述逻辑是必要的，其表达能力是严格大于不带函数的描述逻辑的表达能力。

针对图书馆的概念模型中的一系列不能表示的问题，本文提出了概念上的个体化函数、概念上的一般化函数、角色上的具体化函数、角色上的一般化函数、个体到个体的函数 5 种函数。在现有描述逻辑语言的基础上添加这几种函数的符号，给出了带函数的描述逻辑的语言；通过改进现有描述逻辑的模型，将论域分成具体的论域和抽象的论域两个层次，用函数联系起两个论域，给出了带函数的描述逻辑的模型。最后用带函数的描述逻辑来解决图书馆的概念模型中的原有描述逻辑不能表示的问题。

2　描述逻辑 *ALC*

2.1　描述逻辑 *ALC* 的语法

ALC 语言包括如下符号：

(1)原子概念：$A_1,\cdots,A_n,\cdots$；

(2)原子角色：$R_1,\cdots,R_n,\cdots$；

(3)概念常量：$\top,\bot$；

(4)构造子：$\neg,\sqcap,\forall,\exists$；

(5)个体名称：$a_1,\cdots,a_n,\cdots$。

在元语言中使用 A 表示原子概念，使用 C,D 表示概念描述，使用 R 表示原子角色，使用 a,b 表示个体常量符号。

一个符号串 C 是一个概念描述，如果：

(1)C 是一个原子概念，或者；

(2)C 是一个泛概念 $\top$ 或者是一个底概念 $\bot$，或者；

(3)C 是一个概念的否定，即 $C=\neg D$，其中 D 是一个概念描述，或者；

(4)C 是两个概念的并，即 $C=D\sqcap E$，其中 D,E 是概念描述，或者；

(5)C 是一个值限制的概念，即 $C=\forall R.D$，或者是一个受限存在量词的概念，即 $C=\exists R.D$，其中 R 是原子角色，D 是一个概念描述。

形式地，概念描述定义如：$C=A|\top|\bot|\neg D|C\sqcap D|\forall R.C|\exists R.C$。

描述逻辑中的断言包括以下几种：

(1)$C(a)$表示 a 与 C 具有事例关系；

(2)$R(a,b)$表示 a 与 b 具有 R 关系；

(3)$C\equiv D$ 表示概念 C 与概念 D 是等价的，可以用来引入新的原子概念；

(4)$C\sqsubseteq D$ 表示概念 C 包含于概念 D。

2.2　描述逻辑 *ALC* 的形式语义

一个 *ALC* 描述逻辑系统的模型 M 是由一个非空的论域 Δ 和一个解释函数 I 组成的，记为 $M=(\Delta,I)$，其中解释函数 I 将每个原子概念 A 解释为论域上的子集 $A^I\subseteq\Delta$；将每个原子角色 R 解释为论域上的二元关系的子集 $R^I\subseteq\Delta\times\Delta$；将每个个体常量符号解释到论域中的一个元素 $a^I\in\Delta$。对复杂概念描述的解释根据其结构归纳定义如下：

$\top^I=\Delta$

$\bot^I=\varnothing$

$(\neg C)^I=\Delta\backslash C^I$

$(C\sqcap D)^I=C^I\cap D^I$

$(\forall R.C)^I=\{a\in\Delta:\forall b((a,b)\in R^I\rightarrow b\in C^I)\}$

$(\exists R.C)^I=\{a\in\Delta:\exists b((a,b)\in R^I\wedge b\in C^I)\}$

描述逻辑中的可满足关系定义如下：

$M\models C\equiv D$ iff $C^I=D^I$

$M\models C\sqsubseteq D$ iff $C^I\subseteq D^I$

$M\models C(a)$ iff $a^I\subseteq C^I$

$M\models R(a,b)$ iff $(a^I,b^I)\in R^I$

3　现有描述逻辑不能表示的问题及解决方法

现有描述逻辑系统不能表示概念上的函数和角色上的函数，以图书馆的概念模型[2]为例来介绍现有描述逻辑不能表示的问题，并提出解决方法。

3.1　使用现有描述逻辑表示图书馆的概念模型

假设有这样一个图书馆，要表示如下的信息(为方便形式表示，缩小数据)：图书馆里有 4 本书可以供借阅，分别用符号 *book*1，*book*2，*book*3，*book*4 来表示；有 2 个读者，分别用符号 *borrower*1，*borrower*2 来表示；有 2 个员工，分别用符号 *staff*1，*staff*2 来表示；其中由 *staff*1 管理图书，*staff*2 管理读者。图书和员工都属于图书馆的资源。图书馆的书可以分为计算机方面的书和非计算机方面的书，其中 *book*1，*book*3 属于计算机类图书，其余为非计算机类图书；也可以分为硬皮书和软皮书，其中 *book*3 属于硬皮书，其余的属于软皮书。借书是图书馆提供的服务之一，目前有 3 个借书条目：*borrower*1 借了 *book*1，*borrower*2 借了 *book*2 和 *book*3。

用描述逻辑表示上述图书馆的概念模型，定义如下符号：

(1)原子概念：*Book*，*ComputerBook*，*SoftCoverBook*，*HardCoverBook*，*Staff*，*Borrower*，*Integer*；

(2)原子角色：*Borrow*；

(3)概念常量：$\top,\bot$；

(4)构造子：$\neg,\sqcup,\sqcap,\forall,\exists$；

(5)个体名称：*book*1，*book*2，*book*3，*book*4，*staff*1，*staff*2，*borrower*1，*borrower*2，1，2，$\cdots$。

其中概念 *Book* 用来表示类图书，*Staff* 用来表示类员工，*Borrower* 用来表示类读者，*Integer* 用来表示类整数，*ComputerBook* 是 *Book* 的子概念，用来表示计算机方面的书，*HardCoverBook* 和 *SoftCoverBook* 分别表示硬皮书和软皮书，也是 *Book* 的子概念，并且这两个概念的并集形成概念 *Book*。角色 *Borrow* 用来表示借书关系。

定义公理表示 *Book*，*ComputerBook*，*HardCoverBook* 和 *SoftCoverBook* 之间的关系：

$ComputerBook\sqsubseteq Book$

$Book\equiv HardCoverBook\sqcup SoftCoverBook$

$HardCoverBook\equiv Book\sqcap\neg SoftCoverBook$

该图书馆的模型是由一个二元组组成的，记为 $M=(\Delta,I)$，其中 Δ 是一个非空论域，表示的是图书馆的概念模型中所有相关个体的集合，I 是一个解释函数。Δ 的定义如：$\Delta=\{book1^I,\cdots,book4^I;staff1^I,staff2^I;borrower1^I,borrower2^I,1^I,2^I,\cdots\}$。

I 将每个个体名称解释到论域中的个体，$book1^I$ 等表示各个个体名称所对应的个体。I 对概念以及角色的解释如下：

$Book^I=\{book1^I, book2^I, book3^I, book4^I\}$

$ComputerBook^I=\{book1^I, book3^I\}$

$SoftCoverBook^I=\{book1^I, book2^I, book4^I\}$

$HardCoverBook^I=\{book3^I\}$

$Staff^I=\{staff1^I, staff2^I\}$

$Borrower^I=\{borrower1^I, borrower2^I\}$

$Integer^I=\{1^I, 2^I, \cdots\}$

$Borrow^I=\{(borrower1^I, book1^I), (borrower2^I, book2^I), (borrower2^I, book3^I)\}$

则该模型满足以下断言：

$Book(book1), \cdots, Book(book4)$

$Staff(staff1), Staff(staff2)$

$Borrower(borrower1), Borrower(borrower2)$

$Integer(1), Iteger(2), \cdots$

$ComputerBook(book1), \cdots$

$HardCoverBook(book3)$

$SoftCoverBook(book1)$

$Borrow(borrower1, book1), \cdots$

3.2 函数的引入

上述模型显然还有很多信息没有表示出来，例如：

(1)书和员工都属于图书馆的资源，由 *staff*1 管理书，由 *staff*2 管理读者；

(2)图书馆共有 4 本书，2 个员工，2 个读者；

(3)借书是图书馆提供的服务之一；

(4)目前有 3 个借书条目。

对于(1)，引入概念上的个体化函数。“书是图书馆的资源”和“书由 *staff*1 管理”描述的是个体化之后的书的性质。引入个体化函数将一个概念映射到它个体化之后的个体，继而可以表示其性质。用 i 表示该函数。则 $i(Book)$ 表示个体化之后的书，用 *ibook* 来表示这个个体，即 $i(Book)=ibook$。“书是图书馆的资源”可以表示成 $Resource(ibook)$ 或 $Resource(i(Book))$。

“*staff*1 是图书的管理员”描述的是两个个体之间的关系，可以用角色表示，为更方便更直观地表示这种关系，引入从个体到个体的函数。个体到个体的函数是对原有描述逻辑中角色的扩展，不仅可以表示多个个体之间的关系，还可以表示出这种关系的函数特性。例如管理用函数 *manager* 表示，$manager(a)=b$ 表示个体 a 的管理人是个体 b。多个个体之间的函数关系例如 *add*，用来表示两数相加的和等于第三个数，如 $add(a,b)=c$ 表示 a 加上 b 等于 c。“*staff*1 是图书的管理员”表示为 $manager(ibook)=staff1$ 或 $manager(i(Book))=staff1$。

对于(2)，引入概念上的一般化函数。与概念上的个体化函数一样，概念上的一般化函数也是一个从概念描述到个体的部分映射，但这个个体可以是论域中任一个个体，它表示的是概念中所有事例组成的集合的性质。在元语言中，用 f 表示这种函数。C 表示一个概念描述，则 $f(C)$ 为一个个体名称。概念到个体的一般化函数解释为论域上的幂集到论域的部分映射。如定义函数 *countBook* 表示概念 *Book* 的事例的数目，则要表示图书馆中有 4 本书，可形式化为 $countBook(Book)=4$。

原子概念上的函数的值，可以由定义直接得到。由原子概念可以形成各种各样复杂的概念描述，这样的概念描述将有无穷多个，但不可能将每一种可能的复杂的概念描述在一个函数上的值都定义出来。复杂的概念描述的值可以由其中所包含的原子概念的值的组合得到，不同的构造子具有不同的组合情况。例如 $C\sqcup D$ 在 *countBook* 上的值是由 C 的值，D 的值和 $C\sqcap D$ 的值组合得到，并且由前两个值的和减去第三个值得到。定义 $countBook(C\sqcup D)$ 如：$countBook(C\sqcup D)=f^*(countBook(C), countBook(D), countBook(C\sqcap D))$，其中 $f^*()$ 表示的是一个包含三个参数的个体到个体的函数，其值由前两个参数的和减去第三个参数得到，解释为：$(f^*)^I=\{((a_1,a_2,a_3),a_4): a_1,a_2,a_3,a_4\in Integer^I \text{ and } a_1+a_2-a_3=a_4\}$。

若已知以下断言成立：

$countBook(ComputerBook)=2$

$countBook(HardCoverBook)=1$

$countBook(ComputerBook\sqcap HardCoverBook)=1$

则有下面的断言成立：

$$countBook(ComputerBook\sqcup HardCoverBook)=$$
$$f^*(count(ComputerBook), countBook(HardCoverBook),$$
$$countBook(ComputerBook\sqcap HardCoverBook))=$$
$$f^*(2,1,1)=2$$

$countBook(C\sqcap D)$ 定义如：$countBook(C\sqcap D)=f^*(countBook(C), countBook(D), countBook(C\sqcup D))$。

有时复杂概念描述的函数也可以不经过对其中原子概念的值的组合得到结果，而由解释函数得到，例如可以对 *countBook* 的解释定义如：$countBook^I=\{(\{a_1,a_2,\cdots,a_n\},k): a_1,\cdots,a_n\in Book^I, k\in Integer^I \text{ and} |\{a_1,\cdots,a_n\}|=k\}$。则对任意一个原子概念或者复杂的概念描述都可以直接得到它的值。

对于(3)，引入角色上的具体化函数。在“借书是图书馆提供的服务之一”这句话里，借书不再是一个性质，借书被具体化为一个个体。引入角色上的具体化函数将一个角色映射到它具体化之后的个体，继而可以表示其性质。在元语言中用 r 表示，若 R 是一个角色，则 $r(R)$ 是一个个体名称。$r(Borrow)$ 表示借书关系具体化后的个体，可以用 *rborrow* 表示，即 $r(Borrow)=rborrow$。“借书是图书馆的服务之一”可表示为 $Service(r(Borrow))$ 或 $Service(rborrow)$，其中 *Service* 是一个概念，表示图书馆的服务。

对于(4)，引入角色上的一般化函数。图书馆目前共有 3 条借书条目，分别用 $Borrow(borrower1, book1)$，$Borrow(borrower2, book2)$，$Borrow(borrower2, book3)$ 表示，将它们看作是 *Borrow* 角色的事例。用角色上的一般化函数来表示由所有角色事例组成的集合的性质。角色上的一般化函数是一个从角色到个体的部分映射，该个体可以是论域中任意的个体。在元语言中用 g 表示，若 R 是一个角色，则 $g(R)$ 是一个个体名称。如定义函数 *countBorrow* 表示借书事例的个数，则目前图书馆中有 3 个借书条目则表示为 $countBorrow(Borrow)=3$。与概念上的函数不一样的是，在 *ALC* 中，角色只有原子角色，没有复杂角色，所有不存在对复杂角色上的函数的定义。原子角色上的函数由定义直接给出。

4 带函数的描述逻辑的形式定义

引入概念上的个体化函数和角色上的具体化函数之后，产生了两个层次的个体：一个是具体的个体组成的层次，如 *book*1、*staff*1 等；一个是具体化之后形成的抽象的个体组成的层次，如 *ibook*、*rborrow* 等。按原来的描述逻辑的模型，这会使得概念 $\neg C$ 个体化之后的个体可能成为 $\neg C$ 的事例。为了避免这种逻辑矛盾，将论域分成两个部分，一部分包含具体的个体，称为具体的论域；一部分包含抽象的个体，称为抽象的论域。在

这两个论域上独立形成各自的概念和角色。具体论域中的个体分类抽象形成的概念称为具体的概念;抽象论域中的个体分类抽象形成的概念称为抽象的概念。对角色也是如此,分为抽象的角色和具体的角色。概念和角色都只能在它对应的论域层次上解释。通过函数将两个层次的论域联系起来。

4.1 带函数的描述逻辑的语法

在 ALC 中引入函数,其语言包括如下符号:

(1)具体原子概念:$A_1,\cdots,A_n,\cdots$;

(2)抽象原子概念:$B_1,\cdots,B_n,\cdots$;

(3)具体原子角色:$R_1,\cdots,R_n,\cdots$;

(4)抽象原子角色:$S_1,\cdots,S_n,\cdots$;

(5)概念常量:$\top_1,\top_2,\bot$;

(6)构造子:$\neg,\sqcap,\forall,\exists$;

(7)个体名称:$a_1,\cdots,a_n,\cdots$;

(8)个体到个体的函数:$h_1,\cdots,h_n,\cdots$;

(9)概念上的个体化函数:i;

(10)概念上的一般化函数:$f_1,\cdots,f_n,\cdots$;

(11)角色上的具体化函数:r;

(12)角色上的一般化函数:$g_1,\cdots,g_n,\cdots$。

在元语言中用 A 表示具体原子概念,用 B 表示抽象原子概念。由具体的原子概念和具体的原子角色形成具体的概念描述,用 C 等表示,其形式定义如:$C=A|\top_1|\bot|\neg D|C\sqcap D|\forall R.C|\exists R.C$。

由抽象的原子概念和抽象的原子角色形成抽象的概念描述,用 C' 等表示,其形式定义如:$C'=B|\top_2|\bot|\neg D'|C'\sqcap D'|\forall S.C'|\exists S.C'$。

在元语言中,用 h 表示个体到个体的函数,用 f 表示概念上的一般化函数,用 g 表示角色上的一般化函数。那么 $h(a_1,\cdots,a_n)$ 是一个个体名称,$f(C)$ 或 $f(C')$ 是一个个体名称,$g(R)$ 或 $g(S)$ 是一个个体名称。i 表示概念上的个体化函数,它将具体的概念个体化,$i(C)$ 表示将概念描述 C 个体化之后的个体的名称。r 表示角色上的具体化函数,它将具体的角色具体化,$r(R)$ 表示将角色 R 具体化之后的个体的名称。

带函数的描述逻辑的断言包括如下:

(1)原有的描述逻辑中的断言:$C(a)$,$C'(a)$,$R(a,b)$,$C\equiv D$,$C'\equiv D'$,$C\sqsubseteq D$,$C'\sqsubseteq D'$;

(2)$a=b$,表示两个个体等价;

(3)$a=i(C)$,表示概念 C 个体化之后的个体是 a;

(4)$a=f(C)$,表示 f 作用在概念 C 上得到个体 a;$a=f(C')$,表示 f 作用在概念 C' 上得到个体 a;

(5)$a=r(R)$,表示角色 R 具体化之后的个体是 a;

(6)$a=g(R)$,表示 g 作用在角色 R 上得到个体 a;$a=g(S)$,表示 g 作用在角色 S 上得到个体 a;

(7)$a=h(a_1,\cdots,a_n)$,表示 h 作用在 $a_1,\cdots,a_n$ 上得到个体 a。

4.2 带函数的描述逻辑的语义

带函数的描述逻辑的模型 M 是由两个二元组(Δ,I)组成的,其论域 Δ 由两部分组成,$\Delta=\Delta_1\cup\Delta_2$ 且 $\Delta_1\cap\Delta_2=\emptyset$。$\Delta_1$ 和 Δ_2 分别表示论域的两个层次,Δ_1 称为具体的论域,包含的是领域中具体的个体;Δ_2 称为抽象的论域,包含的是领域中抽象的个体。I 是一个解释函数,定义如下:

(1)将每个具体原子概念解释到子论域 Δ_1 上的一个子集,即 $A^I\subseteq\Delta_1$;

(2)将每个抽象原子概念解释到子论域 Δ_2 上的一个子集,即 $B^I\subseteq\Delta_2$;

(3)将每个具体原子角色解释到子论域 Δ_1 上的一个二元关系,即 $R^I\subseteq\Delta_1\times\Delta_1$;

(4)将每个抽象原子角色解释到子论域 Δ_2 上的一个二元关系,即 $S^I\subseteq\Delta_2\times\Delta_2$;

(5)将每个个体名称解释到论域 Δ 中的一个元素,即 $a^I\in\Delta$;

(6)将每个复杂的概念描述的解释递归定义如下:

$\top_1^I=\Delta_1$

$\top_2^I=\Delta_2$

$\bot^I=\emptyset$

$(\neg C)^I=\Delta_1\backslash C^I$

$(\neg C')^I=\Delta_2\backslash C'^I$

$(C\sqcap D)^I=C^I\cap D^I$

$(C'\sqcap D')^I=C'^I\cap D'^I$

$(\forall R.C)^I=\{a\in\Delta_1:\forall b((a,b)\in R^I\rightarrow b\in C^I)\}$

$(\forall S.C')^I=\{a\in\Delta_2:\forall b((a,b)\in S^I\rightarrow b\in C'^I)\}$

$(\exists R.C)^I=\{a\in\Delta_1:\exists b((a,b)\in R^I\wedge b\in C^I)\}$

$(\exists S.C')^I=\{a\in\Delta_2:\exists b((a,b)\in S^I\wedge b\in C'^I)\}$

(7)个体到个体的函数 h 解释为从论域 Δ 上的 n 元关系的幂集即 2^{Δ^n} 到论域 Δ 的部分映射;

(8)概念上的个体化函数 i 解释为从子论域 Δ_1 上的幂集即 2^{Δ_1} 到子论域 Δ_2 的部分映射;

(9)角色上的具体化 r 解释为从子论域 Δ_1 上的二元关系的幂集即 $2^{\Delta_1\times\Delta_1}$ 到子论域 Δ_2 的部分映射;

(10)概念上的一般化函数 f 解释为从子论域 Δ_1 上的幂集与 Δ_2 上的幂集的并集即 $2^{\Delta_1}\cup 2^{\Delta_2}$ 到子论域 Δ_1 与 Δ_2 的并集即 $\Delta_1\cup\Delta_2$ 的部分映射;

(11)角色上的一般化函数 g 解释为从子论域 Δ_1 上的二元关系的幂集与 Δ_2 上的二元关系的幂集的并集即 $2^{\Delta_1\times\Delta_1}\cup 2^{\Delta_2\times\Delta_2}$ 到子论域 Δ_1 与 Δ_2 的并集即 $\Delta_1\cup\Delta_2$ 的部分映射。

可满足关系定义如下:

$M\models C(a)$ iff $a^I\in C^I$

$M\models C'(a)$ iff $a^I\in C'^I$

$M\models R(a,b)$ iff $(a^I,b^I)\in R^I$

$M\models S(s,b)$ iff $(a^I,b^I)\in S^I$

$M\models C\equiv D$ iff $C^I=D^I$

$M\models C'\equiv D'$ iff $C'^I=D'^I$

$M\models C\sqsubseteq D$ iff $C^I\subseteq D^I$

$M\models C'\sqsubseteq D'$ iff $C'^I\subseteq D'^I$

$M\models a=b$ iff $a^I=b^I$

$M\models a=i(C)$ iff $a^I=i^I(C^I)$

$M\models a=f(C)$ iff $a^I=f^I(C^I)$

$M\models a=f(C')$ iff $a^I=f^I(C'^I)$

$M\models a=r(R)$ iff $a^I=r^I(R^I)$

$M\models a=g(R)$ iff $a^I=g^I(R^I)$

$M\models a=g(S)$ iff $a^I=g^I(S^I)$

$M\models a=h(a_1,\cdots,a_n)$ iff $a^I=h^I(a_1^I,\cdots,a_n^I)$

5 在图书馆例子中的应用

在这一部分,使用带函数的描述逻辑来解决第 3 章中提出

的图书馆的概念模型中不能表示的一些问题：图书馆共有4本书，书和员工都属于图书馆的资源，由 *staff*1 管理书，由 *staff*2 管理读者，借书是图书馆提供的服务之一，目前有3个借书条目。

除了原来论域中的个体，还要引入将 *Book*、*Staff* 和 *Borrower* 这些概念个体化后得到的个体，分别用 *ibook*、*istaff* 和 *iborrower* 表示，以及将角色 *Borrow* 具体化得到的个体 *rborrow*。*ibook* 和 *istaff* 属于图书馆的资源，形成概念 *Resource*。*rborrow* 形成概念 *Service*。新增加的个体都属于论域的第二个层次。

对3.1节中定义的图书馆的概念模型中的符号作如下修改：定义的原子概念成为具体的原子概念；定义的原子角色成为具体的原子角色；去掉（3）中的$\top$，添加$\top_1$和$\top_2$；在（5）中添加 *ibook*、*istaff*、*iborrower* 和 *rborrow*；增加如下符号：

（6）抽象原子概念：*Resource*，*Service*；

（7）个体到个体的函数：*manager*，f^*；

（8）概念上的个体化函数：i；

（9）概念上的一般化函数：*countBook*；

（10）角色上的具体化函数：r；

（11）角色上的一般化函数：*countBorrow*。

该图书馆的模型是由一个二元组组成的，记为 $M=(\Delta, I)$，其中 $\Delta=\Delta_1\cup\Delta_2$，$I$ 是一个解释函数。Δ_1 的定义与前面一样，Δ_2 的定义如：$\Delta_2=\{ibook^I, istaff^I, iborrower^I, rborrow^I\}$。

I 对新增加的概念和角色以及函数的解释如下：

$i^I=\{(\{book1^I,\ \cdots, book4^I\}, ibook^I), (\{staff1^I, staff2^I\}, istaff^I), (\{borrower1^I, borrower2^I\}, iborrower^I)\}$

$manager^I=\{(ibook, staff1), (iborrower2, staff2)\}$

$r^I=\{(\{(borrower1^I, book1^I), (borrower2^I, book2^I), (borrower2^I, book3^I)\}, rborrow^I)\}$

$countBook^I=\{(\{a_1,\cdots,a_n\}, k): a_1,\cdots,a_n\in Book^I, k\in Integer^I \text{ and } |\{a_1,\cdots,a_n\}|=k\}$

$f^{*I}=\{((a_1,a_2,a_3),a_4): a_1,a_2,a_3,a_4\in Integer^I \text{ and } a_1+a_2-a_3=a_4\}$

$countBorrow^I=\{(\{r_1,\cdots,r_n\}, k): r_1,\cdots,r_n\in Borrow^I \text{ and} |\{r_1,\cdots,r_n\}|=k\}$

该模型所满足的断言除了3.1节中的，还包括：

$i(Book)=ibook$

$i(Staff)=istaff$

$i(Borrower)=iborrower$

$Resource(ibook)$

$Resource(istaff)$

$r(Borrow)=rborrow$

$Service(rborrow)$

$manager(ibook)=staff1$

$manager(iborrower)=staff2$

$f^*(2,4,2)=4$

$countBook(Book)=4$

$countBook(ComputerBook)=2$

$countBook(SoftCoverBook)=3$

$countBook(ComputerBook\sqcap SoftCoverBook)=1$

$countBook(ComputerBook\sqcup SoftCoverBook)=4$

$countBorrow(Borrow)=3$

6 总结

本文完成了如下工作：（1）使用现有描述逻辑表示图书馆的概念模型，发现其不足之处，引入函数来增强其表达能力，提出了概念和角色上的4种函数：概念上的个体化函数将一个概念映射到它个体化之后的个体，从而可以表示概念的性质，解决了概念的个体化问题；概念上的一般化函数将一个概念映射到论域中某一个个体，表示该概念所有事例组成的集合的性质，解决了个体集合的性质的问题；角色上的具体化函数将一个角色映射到它具体化之后的个体，从而可以表示角色的性质，解决了角色的具体化问题；角色上的一般化函数将一个角色映射到论域中某一个个体，表示该角色所有事例组成的集合的性质，解决了角色事例集合的性质的问题。这4种函数增强了描述逻辑的表达能力，可以很好地解决图书馆的概念模型中不能表示的那些问题。还提出了个体到个体的函数以扩展角色，引入它并没有增强描述逻辑的表示能力，但是使用它可以更直观更自然表示多个个体之间的函数关系，也更易于建模。（2）给出了带函数的描述逻辑具体的语法和形式语义，提出了将模型的论域分成具体论域和抽象论域两个层次的思想。具体的论域上抽象形成具体概念和具体角色，抽象的论域上抽象形成抽象概念和抽象角色，通过概念上的个体化函数和角色上的具体化函数将两个论域联系起来：概念上的个体化函数将具体概念映射到抽象论域中的一个个体，角色上的具体化函数将具体角色映射到抽象论域中的一个个体。通过对论域分层清楚地区分了两种不同层次的个体，同时还避免了概念个体化引起的逻辑矛盾。（3）将带函数的描述逻辑应用到图书馆的概念模型上，表示其用原有描述逻辑不能表示的问题，给出了具体的语言，语法和语义模型。

在描述逻辑中引入函数不仅增强了描述逻辑的表达能力，使得表达方便直观，而且解决了本体上具体化引起的多个层次的问题，使得不同的层次之间既有区别又有联系。在具体的应用中使用带函数的描述逻辑也能较好的解决实际问题。

参考文献：

[1] Alex B.On the relative expressive power of description logics and predicate calculus[J].Artificial Intelligence，1996，82（1/2）：353-367.

[2] Alex B，Ronald J B.Conceptual modeling with description logics[M]//Franz B，Diego C，Deborah M，et al.the Description Logic Handbook.[S.l.]：Cambridge University Press，2003：359-381.

[3] Diego C，Giuseppe D G.Expressive description logics[M]//Franz B，Diego C，Deborah M，et al.the Description Logic Handbook.[S.l.]：Cambridge University Press，2003：184-225.

[4] Diego C，Giuseppe D G，Maurizio L，et al.Reasoning in expressive description logics[M]//Alan R，Andrei V.Handbook of Automated Reasoning.Amsterdam：Elsevier Science Publishers，2001：1581-1634.

[5] Daniele N，Ronald J B.An introduction to description logics[M]//Franz B，Diego C，Deborah M，et al.the Description Logic Handbook.[S.l.]：Cambridge University Press，2003：1-46.

[6] Description logics website[EB/OL].http：//dl.kr.org/.

[7] Franz B，Werner N.Basic description logics[M]//Franz B，Diego C，Deborah M，et al.the Description Logic Handbook.[S.l.]：Cambridge University Press，2003：47-100.

[8] Franz B，Ralf K，Frank W.Extensions to description logics[M]//Franz B，Diego C，Deborah M，et al.the Description Logic Handbook.[S.l.]：Cambridge University Press，2003：226-268.

[9] Francesco M D.Complexity of reasoning[M]//Franz B，Diego C，Deborah M，et al.the Description Logic Handbook.[S.l.]：Cambridge University Press，2003：101-141.