信念修正开放过程模式可判定公式表达能力

张 伟[1]，柳玉辉[2]

（1. 东北大学 计算机科学与工程学院，沈阳 110819；2. 东北大学 研究院，沈阳 110819）

摘 要：提出了一种可以在两个极大可判定前缀词公式类上实现信念修正 OPEN 过程模式的算法，该算法从初始形式理论出发，依据不断获取的新事实信息，迭代修正形式理论，完成形式理论的进化。探讨了 OPEN 过程模式的两种可判定公式类的表达能力，证明了其表达能力强于一阶逻辑的子集 FO^2 和模态逻辑，并给出了实际应用例子。

关键词：人工智能；信念修正；开放过程模式；可判定公式类；表达能力；定理机器证明

中图分类号：TP181；TP18　　**文献标志码**：A　　**文章编号**：1671-5497(2016)06-2021-06

DOI：10.13229/j.cnki.jdxbgxb201606035

Expressive power of decidable subclasses of open proxcheme

ZHANG Wei[1], LIU Yu-hui[2]

(1. *College of Computer Science and Engineering, Northeastern University, Shenyang* 110819, *China*; 2. *Research Academy, Northeastern University, Shenyang* 110819, *China*)

Abstract: An algorithm for OPEN proxcheme is proposed that functions in the two maximal decidable classes of first-order logic. The algorithm iteratively revises the formal theories according to new information and completes the evolution of the formal theories. The expressive power of the OPEN proxcheme is discussed. It is demonstrated that the OPEN proxcheme and R-calculus preserve the decidability of formulas in the two maximal decidable classes. Since the decision problem for modal logic and FO^2 can be transferred to that of the two maximal decidable classes, it can be concluded that the OPEN proxcheme and R-calculus are more expressive than the modal logic and FO^2. Practical examples, such as hardware description and verification, are provided.

Key words: artificial intelligence; belief revision; OPEN proxcheme; decidable logic formula classes; expressive power; mechanical theorem proving

0 引 言

信念修正是由 Gardenfors 及他的两位同事 Alchourron 和 Makinson（简称 AGM）于 1985 年创立的一种用于描述信念和知识的动态演化的形式理论[1,2]。近年来随着海量数据处理对数据演化规律研究的兴起，信念修正成为研究热点[3,4]。

早期的信念修正理论方法都是一次性的修正方法，没有探讨信念集被多次修正和多次修正的收敛性等问题[5]。1992 年建立并发展起来的开放逻辑[2]开始对信念修正中的多次修正进行了系统的研究，并取得了引人注目的成果。1997 年

收稿日期：2015-07-02.

基金项目：国家科技支撑计划项目(2014BAI17B01)；软件开发环境国家重点实验室开放课题项目(SKLSDE-2012KF-02).

作者简介：张伟(1962-)，男，教授，博士. 研究方向：人工智能，机器学习. E-mail：zhangwei@cse.neu.edu.cn

Darwiche 和 Pearl 提出了信念修正的迭代方法[6]，但是该方法只对必须是理论闭包的信念集进行迭代修正，而且其收敛性没有得到证明。这种情况一直延续到 2007 年 Nayak 提出动态修正算子也没有实质性的改变[7,8]。2007 年，Li 提出的 R-演算理论却给出了一个解决该问题的方法：①R-演算可以求解出当前信念集关于新事实集合的极大协调子集；②R-演算理论给出了一个描述理论版本进化的形式化认识进程，并称之为 OPEN 过程模式，而且证明了对上述极大协调子集反复修正的 OPEN 过程模式最终必收敛到与所讨论问题相关的所有真值公式所构成的集合[9,10]。

由于 OPEN 过程模式引用了对“$\Gamma\vdash A$ 是否为真”的判断，其求解极大协调子集的过程在一阶逻辑范围中是不可判定的，也就是说，在一阶逻辑的范围内，OPEN 过程模式是难以通过编写程序在计算机上实现的[11,12]。这个问题引发人们对开放逻辑 R-演算的可计算性质的研究，在不同的范围内讨论 R-演算的编程实施方法：文献[13]给出了在命题逻辑上的一种修改的 R-演算规则，使得对信念集中任何一个极大协调的子集都可以通过这组规则得到；文献[14]提出了一个当信念集为 Horn 子句集时求解所有极大缩减的交互式算法。该方法的新颖之处是给出了判定某些极小不协调 Horn 公式子集的必要条件，而在此条件以外的不协调 Horn 公式子集的判定则采用人机交互方式来完成。文献[15]认为若想实现对开放逻辑的计算机编程，应将其限制在其可判定的范围内，并描述了在纯谓词逻辑情况下开放逻辑的可判定公式类是两个前缀词公式类[$\exists^*\forall^*$;all;(0)]和[$\exists^*\forall\forall\exists^*$; all; (0)]，简称为“两个 GOOD 公式类”，最后给出了实现 R-演算的算法。

本文提出了一个在两个 GOOD 公式类上的实现开放逻辑 OPEN 过程模式的算法，该算法迭代修正形式理论，完成形式理论进化。证明了 OPEN 过程模式的两个可判定公式类的表达能力强于一阶逻辑的子集 FO^2 和模态逻辑，并给出了实际的例子。这个结果对将开放逻辑应用于实际应用领域具有一定意义。

1 本文算法

1.1 OPEN 过程模式及其实现算法

开放逻辑提出的 OPEN 过程模式从一组初始的假设出发(这组初始的假设，被当作对某一特定领域问题的公理系统的初始版本 Γ_1)，依据不断得到的新发现事实序列{$A_1,A_2,\cdots,A_n$}，进化出新的理论版本序列，具体方法如下：

(1)若 A_i 满足 $\Gamma_n\vdash A_i$ 则调用过程 NULL，产生当前理论 Γ_i 的理论 E-type 型新版本 Γ_{i+1}；(E-type 型新版本理论 Γ_{i+1} 与原理论 Γ_i 完全相等)。

(2)若 A_i 满足 $\Gamma_n\vdash\sim A_i$ 则调用过程 R-reconstruction(即 R-演算)，产生当前理论 Γ_i 的理论 R-type 型新版本 Γ_{i+1}(R-type 型新版本理论 Γ_{i+1} 是原理论 Γ_i 的一个 R 极大缩减)。

(3)若 A_i 满足 not($\Gamma_n\vdash A_i$&$\Gamma_n\vdash\sim A_i$)，则调用过程 N-reconstruction，产生当前理论 Γ_i 的理论 N-type 型新版本 Γ_{i+1}；(N-type 型新版本理论 Γ_{i+1} 是原理论 Γ_i 的一个 N 扩充)。

当以上 3 个步骤被 OPEN 过程模式依据新发现事实序列{$A_1,A_2,\cdots,A_n$}不断重复的时候，就形成了一个形式理论版本的序列{$\Gamma_1,\Gamma_2,\cdots,\Gamma_n$}。Li[2] 证明了形式理论版本的序列{$\Gamma_1,\Gamma_2,\cdots,\Gamma_n$}是收敛的，并且收敛到所研究的特定问题的全部真值公式集。

当开放逻辑的公式被限定在两个 GOOD 公式类范围内，每个公式的可满足性是可判定的，可以采用生成测试法简洁地构造出实现 OPEN 过程模式的算法。

算法 1 OPEN-Proxcheme-decidable (Γ_1, {$A_1,A_2,\cdots,A_n$})

{ Γ_n,Γ_{n+1}：形式理论；{$A_1, A_2,\cdots,A_n$}：有限的事实公式序列；

```
置当前理论版本记数 n:=1;
置当前事实公式记数 i:=1;
While (Γn没有收敛)
{
(1)If (Γn ⊢Ai)  /* Ai可由 Γn 导出 */
    then 令 Γn+1 := Γn; goto next-loop;
(2)If (Γn ⊢∼Ai) /* Γn 被 Ai反驳 */
    then
    令 Γn+1 :=
    R-calculus-decidable(Γn, Ai);
    goto next-loop;
(3)If (not(Γn ⊢Ai & Γn ⊢ ∼Ai))
    /* Ai是新假设 */
     then 令 Γn+1 := Γn ∪{Ai};
(4) next-loop:
     n:=n+1, i=i+1;
}  /* while */
```

} /* main algorithm */

当形式理论序列 Γ_n 和事实集合$\{A_1, A_2, \cdots, A_n\}$有限时,如果 Γ_n 与事实集合$\{A_1, A_2, \cdots, A_n\}$协调,则称 Γ_n 收敛。关于一个无限的形式理论序列 Γ_n 收敛的严格定义参见文献[3]。算法 OPEN-Proxcheme-decidable 调用的算法 R-calculus-decidable 描述如 1.2 节所示。

1.2 实现 R-演算的算法

如前面所述,在两个 GOOD 公式类范围内,开放逻辑公式的可满足性是可判定的,因此 R-calculus 也是可计算的。换言之,R-calculus 可由算法实现,记这样的算法为 R-calculus-decidable。

算法 2　R-calculus-decidable(Γ_n, A_i)

```
{/* 在两个 GOOD 公式类范围内,求取理论 Γn 关于事实 Ai 的极大缩减 */
Γn:形式理论;Ai:事实公式;
生成理论 Γn 的所有子集,并记为
Subset(Γn)={S1,S2,…,Sm};
For (each Si in{S1,S2,…,Sm})do /* 删除 Subset(Γn)
    中与 Ai 不一致的子集 */
    {
     If (not consistent(Si, Ai))
     then Remove Si from Subset(Γn);
     }
选取剩下的 Subset(Γn)中极大者(可以不唯一),记之为{S′1,S′2,…,S′k};
求出极大缩减 Maximal-Contraction=
{ S′1∪{Ai},…,S′k∪{Ai}};
 返回结果 Maximal-Contraction;
}/* R-calculus-decidable */
```

值得注意的是,由于在两个 GOOD 公式类范围内,开放逻辑公式的可满足性是可判定的,即语句 consistent(S_i, A_i)是有算法可以判定的,所以整个 R-calculus-decidable(Γ_n, A_i)是一个算法;同时,由于该算法使用简单的生成测试法,枚举了理论 Γ_n 的所有子集,并一一测试,然后取其极大者,文献[3]中描述的"过程模式 OPEN Proxcheme 可能丢失信息"的情况在算法 OPEN-Proxcheme-decidable 中不可能出现。

2　两类可判定公式表达能力及其应用

将开放逻辑使用的公式范围限定在两个 GOOD 公式类中可以得到很好的计算性质:OPEN 过程模式和 R-演算都变成 Turing 可计算的,换言之,尝试将开放逻辑在计算机上编程实现是可行的。进而需要回答的问题是:两个 GOOD 类中逻辑公式的表达能力如何?它们能描述哪些实际问题?本节将对这些问题进行讨论,并给出几个实际的例子。

2.1 模态逻辑和两变量一阶逻辑

命题模态逻辑(Modal logic,ML)虽然名称上有命题的字样,但实际命题模态逻辑公式是一阶逻辑的片段,目前被广泛地应用于计算机科学的各个领域,如程序规约验证、硬件系统验证、人工智能、数据库等[16-18]。这些形式化方法的本质都是模态逻辑[19]。模态逻辑提供了模态算子来表达"必然"(记为"□")和"可能"(记为"◊")两个逻辑概念。在具体的应用中,"必然"可以表示"依照物理定律"、"按照人们的信念"等概念;"可能"表示"将会有…"等概念,加之其具有可判定性,因此模态逻辑被广泛使用。

为了考查两个 GOOD 公式类的表达能力与模态逻辑的关系,先回顾 Henkin[19,20] 提出的两变量一阶逻辑 FO^2 的概念。

考虑具有确定谓词集的一阶逻辑 FO,k 变量一阶逻辑 FO^k 是由所有最多有 k 个不同个体变量的一阶逻辑公式组成,即:

$$FO = \bigcup_{k=1}^{k=\infty} FO^k$$

因此 FO^2 是由所有最多有 2 个不同个体变量的逻辑公式组成的一阶逻辑片段。例如,可以用 FO^2 的公式表示在图 $G=(V,E)$ 中"有一条从 x 到 y 的长度为 5 的路径":

$$\exists x \exists y(Exy \wedge \exists x(Eyx \wedge \exists y(Exy \wedge \exists x(Eyx \wedge \exists y Exy))))$$

Exy 表示节点 x、y 之间有边相连。这里变量 x、y 在不同的约束作用域中是被交替重复使用的[20,21]。约束变量是哑变量,可以自由变名。

研究表明[22] FO^2 与模态逻辑有密切的关系,后者在计算机领域有很多实际应用的例子。时态逻辑是模态逻辑应用于实时系统描述中的一个变种,它将模态词"□"解释为"将永远…",把模态词"◊"解释为"将会有…",即□A 表示"A 将永远真",◊A 表示"A 将会真"。除模态词"□"和"◊"之外,时态逻辑还引进了另外两个模态词:"○"(next),○A 表示 A 在下一时刻为真;"▷"(until),A▷B 表示 A 一直为真直到 B 为真。这

使得时态逻辑有了更强的表达能力。时态逻辑一直是计算机科学的一个研究热点，在硬件描述、硬件验证、软件描述、软件验证等都得到了广泛的应用。以下是几个硬件描述的例子。

例 1　电位状态变化

设 P 表示 p 点的电位为高位；¬P 表示 p 点的电位为低位；P↑ 表示 p 点的电位从低位到高位的状态变化；P↓ 表示 p 点的电位从高位到低位的状态变化，其时态逻辑公式表示为：

$$P\uparrow = \neg P \wedge \bigcirc P \qquad P\downarrow = P \wedge \bigcirc \neg P$$

例 2　通路晶体管

如图 1(a)所示，在通路晶体管电路中，当 v_3 门的输入信号激活时，v_1 门的信号将被传送到 v_2 门，即如果 v_2 门的电位为低位而 v_1 门的电位为高位，则 v_3 门的电位变为高位时，v_2 门的电位将变为高位；如果 v_2 门的电位为高位而 v_1 门的电位为低位，则 v_3 门的电位变为高位时，v_2 门的电位将变为低位。通路晶体管的电路逻辑描述为：

$$\Box(v_3\uparrow \rightarrow \bigcirc(v_1 \wedge \neg v_2 \rightarrow v_2\uparrow)) \wedge$$
$$\Box(v_3\uparrow \rightarrow \bigcirc(\neg v_1 \wedge v_2 \rightarrow v_2\downarrow))$$

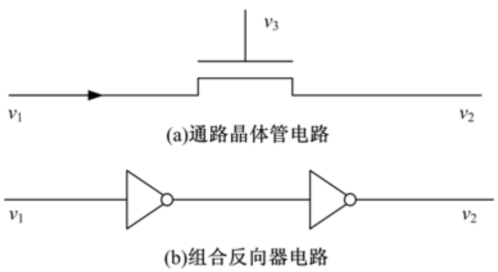

图 1　通路晶体管电路和组合反向器电路

Fig. 1　Circuits of pathway transistor and combined reverser

例 3 反向器

如图 1(b)所示，在组合反向器电路中，当 v_1 门的电位为高位时，下一时刻 v_2 门的电位也将为高位；当 v_1 门的电位为低位时，下一时刻 v_2 门的电位也将为低位，该电路的时态逻辑描述为：

$$\Box(v_1 \rightarrow \bigcirc v_2) \wedge \Box(\neg v_1 \rightarrow \bigcirc \neg v_2)$$

由此可见，模态逻辑以及一元时态逻辑有较强的表达能力和很好应用价值。

2.2　FO² 片段及两个 GOOD 公式类

Gabbay[22] 研究了模态逻辑与一阶逻辑的关系，结论是：模态逻辑是一阶逻辑的一个片段，模态逻辑以及一元时态逻辑实际上是两变量一阶逻辑 FO^2 的一个片段。Scott[23] 首先研究了两变量一阶逻辑 FO^2 片段的可判定性问题，证明了 FO^2 片段的判定问题可以规约为具有 $\forall\forall\exists^*$ 型前缀词公式类的判定问题，而 $\forall\forall\exists^*$ 型前缀词公式类是两个 GOOD 公式类之一 Godel 公式类（$\exists^*\forall\forall\exists^*$）的一个子集。因为 Godel 公式类是可判定的，所以 FO^2 公式也是可判定的。

定理 1　限制于两个 GOOD 公式类上的开放逻辑的表达能力，满足下面的关系：

ML⊆FO^2⊆Godel 类⊆两个 GOOD 类。

证明　开放逻辑是由 R-删除规则、R-逻辑连接词规则、R-量词规则、R-公理组成的形式逻辑推理系统。在全体一阶逻辑上它是不可判定的，但限制在两个 GOOD 类范围后，需要考查开放逻辑在其推理过程中是否改变公式及公式集合的可判定性质。

设 Γ_i 是理论序列 $\{\Gamma_1(=\Gamma), \Gamma_2, \cdots, \Gamma_n\}$ 的某一个元素。从开放逻辑的形式定义可知，开放逻辑的演绎推理过程与新形式理论的重构过程是混合进行的：①演绎推理的初始公式集合 S_0^i（即 Γ_i）是属于两个 GOOD 类公式集的，推理的中间过程使用归结原理可能产生不属于两个 GOOD 类的公式，但演绎推理结束时，一定是其理论闭包 Th(S_0^i)中产生了与初始公式集合 S_0^i 中某个公式 A_i（注意 A_i 属于两个 GOOD 类）相矛盾的导出公式 $\neg A_i$，从而引发了 R-演算。即开放逻辑演绎推理的入口和出口公式是保留在两个 GOOD 类中的，因此对开放逻辑的理论进化有影响的公式的可判定性质不变（仍是可判定的）。②在新形式理论的重构过程（即认识进程 $\{\Gamma_1(=\Gamma), \Gamma_2, \cdots, \Gamma_n\}$ 的演变过程）中，引用了过程 NULL、过程 R-reconstruction 和过程 N-reconstruction：其中过程 NULL 不改变当前理论 Γ_i，过程 N-reconstruction 只对 Γ_i 增加属于两个 GOOD 类新事实公式 A（因为新事实 A 是原子公式或是原子公式的非，仍属于两个 GOOD 公式类），而过程 R-reconstruction（即 R-演算）虽引用了 R-删除规则、R-逻辑连接词规则、R-量词规则、R-公理，但注意到这些公理和规则都是对 Γ_i 进行缩减（求 maximal contractions），因而它们都保持形式理论公式集 Γ_i 被缩减后仍属于两个 GOOD 类公式集，因此仍是可判定的。于是，结合 Gabbay[22] 和 Scott[23] 的结论，有以下表达能力的关系：

ML⊆FO^2⊆Godel 类⊆两个 GOOD 类。

由定理 1 可知，两个 GOOD 类的表达能力大大强于 FO^2 公式和模态逻辑 ML，上述所有例子

都可用开放逻辑来表达和进化，因此开放逻辑在两个 GOOD 公式类范围内可以解决很多诸如硬件验证和软件自动纠错等实际应用问题[24-26]。

结合 Gabbay 关于模态逻辑与一阶逻辑关系的结论“模态逻辑实际上是 FO^2 的一个片段”，通过以下定义的“标准翻译”可以将任一命题模态逻辑公式翻译为对应的一阶逻辑的 FO^2 公式[23-25]：

定义　把 ML 的公式翻译为一阶逻辑公式：对任一模态逻辑公式 $\varphi \in ML$，使用变量 x 和 y，如下递归定义标准翻译 $ST_x(\varphi)$：

对于每个 $P_i \in ML$，$ST_x(P_i) = P_i(x)$；

$ST_x(\neg\varphi) = \neg ST_x(\varphi)$；

$ST_x(\varphi \to \psi) = ST_x(\varphi) \to ST_x(\psi)$；

$ST_x(\Diamond\varphi) = \exists y(R(x,y) \wedge ST_y(\varphi))$；

$ST_x(\Box\varphi) = \forall y(R(x,y) \to ST_y(\varphi))$；

对于每个 $P_i \in ML$，$ST_y(P_i) = P_i(y)$；

$ST_y(\neg\varphi) = \neg ST_y(\varphi)$；

$ST_y(\varphi \to \psi) = ST_y(\varphi) \to ST_y(\psi)$；

$ST_y(\Diamond\varphi) = \exists x(R(y,x) \wedge ST_x(\varphi))$；

$ST_y(\Box\varphi) = \forall x(R(y,x) \to ST_x(\varphi))$；

其中，R 为 ML 的 Kripke 模型中的可达关系；φ 和 ψ 均为 ML 的任意公式。如上交替使用 x 和 y，可以把 ML 的每个公式翻译到一阶逻辑的两变元片断 FO^2。

例 4 通过标准翻译将模态逻辑公式翻译为 FO^2 公式。

$ST_x(\Diamond(\Box p \to q)) = \exists y(R(x,y) \wedge$
$(\forall x(R(y,x) \to P(x)) \to Q(x)))$

例 5 将反向器的时态逻辑公式翻译为 FO^2 公式。

反向器的时态逻辑描述为：

$ST_x(\Box(v_1 \to \bigcirc v_2) \wedge \Box(\neg v_1 \to \bigcirc \neg v_2))$

使用标准翻译，可得：

$ST_x(\Box(v_1 \to \bigcirc v_2) \wedge \Box(\neg v_1 \to \bigcirc \neg v_2)) =$
$ST_x(\Box(v_1 \to \bigcirc v_2)) \wedge ST_x(\Box(\neg v_1 \to \bigcirc \neg v_2))$

其中，

$ST_x(\Box(v_1 \to \bigcirc v_2)) =$
$\forall y(R(x,y) \to ST_y(v_1 \to \bigcirc v_2)) =$
$\forall y(R(x,y) \to ST_y(v_1) \to ST_y(\bigcirc v_2)) =$
$\forall y(R(x,y) \to (ST_y(v_1) \to \exists x(R(y,x) \wedge$
$ST_x(v_2)))) = \forall y(R(x,y) \to (v_1(y) \to$
$\exists x(R(y,x) \wedge v_2(x))))$

类似地，可得到：

$ST_x(\Box(\neg v_1 \to \bigcirc \neg v_2)) = \forall y(R(x,y) \to$
$(\neg v_1(y) \to \exists x(R(y,x) \wedge \neg v_2(x))))$

因此，将反向器的时态逻辑公式翻译为 FO^2 公式的结果是：

$\forall y(R(x,y) \to (v_1(y) \to \exists x(R(y,x) \wedge$
$v_2(x)))) \wedge \forall y(R(x,y) \to (\neg v_1(y) \to \exists$
$x(R(y,x) \wedge \neg v_2(x))))$

本节实例验证了在描述可判定公式类与不可判定公式类的分界线时，有前缀词类和 FO^k 片段两种方法。后者得到的结果是，FO^2 公式是可判定的，FO^3 公式（因为它包含著名的 $\forall\exists\forall$ 公式类）是不可判定的，所以有“FO^2 片段和 FO^3 片段落在一阶逻辑可判定分界线的两边”的结论。相比之下，前缀词类研究得到的结论（两个 GOOD 公式类就落在可判定分界线上）是更加精确的。由此可知，将开放逻辑限定在两个 GOOD 公式类范围不仅保持了良好的表达能力，而且是一阶逻辑公式可判定的极大范围。

3　结束语

开放逻辑 OPEN 过程模式是信念修正的一种多次修正方法，并且具有收敛性。但是，由于 OPEN 过程模式在一阶逻辑范围中是不可判定的，因此该方法难于编程实现。本文提出了一种在两个 GOOD 公式类上的实现开放逻辑 OPEN 过程模式的算法。由于使用的是生成测试法，枚举了理论 Γ_n 的所有子集并测试其一致性，然后取其极大者，因此避开了文献[3]中描述的过程模式 OPEN 过程可能丢失信息的情况。同时，本文证明了 OPEN 过程模式的两个可判定公式类的表达能力强于一阶逻辑的子集 FO^2 和模态逻辑，并给出了实际的例子。

参考文献：

[1] Alchourron C E, Gardenfors R, Makinson D. On the logic of theory change: partial meet contraction and revision functions[J]. Journal of Symbolic Logic, 1985, 50(2): 510-530.

[2] Li W. An open logic system[J]. Science China Series A, 1992, 35(10): 1103-1113.

[3] Li W. Mathematical Logic—Foundations for Information Science [M]. Basel: Birkhauser Verlag, 2010: 45-185.

[4] Benthem J. Reasoning and programming: analogies between logic and computation[C]// Proceedings of

the 1991 International Symposium on Logic Programming, San Diego, USA, 1991:717-718.

[5] Benthem J. Dynamic logic for belief revision[J]. Journal of Applied Non-Classical Logics, 2007, 17(2):129-155.

[6] Darwiche A, Pearl J. On the logic of iterated belief revision[J]. Artificial Intelligence, 1997, 89(1/2):1-29.

[7] Nayak A, Goebel R, Orgun M. Iterated belief contraction from first principles[C]// Proceedings of 20th International Joint Conference on Artificial Intelligence, Hyderabad, India, 2007:2568-2573.

[8] Fermé E, Hansson S O. AGM 25 years: twenty-five years of research in belief change[J]. Journal of Philosophical Logic, 2011, 40(2):295-331.

[9] Li W. R-calculus: an inference system for belief revision[J]. Computer Journal, 2007, 50(4):378-390.

[10] Li W, Sui Y F, A sound and complete R-calculi with respect to contraction and minimal change[J]. Frontiers of Computer Science, 2014, 8(2):184-191.

[11] Zhang W. On the decidability of open logic[J]. Science China: Information Science, 2009, 52(8):1283-1291.

[12] Li H, Luo J, Li W. A formal semantics for debugging synchronous message passing-based concurrent programs[J]. Science China: Information Science, 2014, 57:128101.

[13] Luan S M, Dai G Z, Li W. A programmable approach to revising knowledge base[J]. Science China: Information Science, 2005, 48(6):681-692.

[14] Luo J, Li W. An algorithm to compute maximal contractions for Horn clauses[J]. Science China: Information Science, 2011, 54(2):244-257.

[15] Zhang W. Decidable subsets of open logic and an algorithm for R-calculus[J]. Science China: Information Science, 2015, 58(1):012103.

[16] 刘磊,张银平. 一种基于描述逻辑的构件检索匹配算法[J]. 吉林大学学报:工学版, 2008, 38(3):671-675.
Liu Lei, Zhang Yin-ping. Component retrieval and matching algorithm based on description logics[J]. Journal of Jilin University(Engineering and Technology Edition), 2008, 38(3):671-675.

[17] 黄宏涛,王静,叶海智,等. 基于惰性切片的线性时态逻辑性质验证[J]. 吉林大学学报:工学版, 2015, 45(1):245-251.
Huang Hong-tao, Wang Jing, Ye Hai-zhi, et al. Lazy slicing based method for verifying linear temporal logic property[J]. Journal of Jilin University (Engineering and Technology Edition), 2015, 45(1):245-251.

[18] 王生生,刘大有,吴瑕,等. 模糊空间描述逻辑及应用[J]. 吉林大学学报:工学版, 2010, 40(6):1634-1638.
Wang Sheng-sheng, Liu Da-you, Wu Xia, et al. Fuzzy spatial description logic and application[J]. Journal of Jilin University(Engineering and Technology Edition), 2010, 40(6): 1634-1638.

[19] Bärger E, Grädel E, Gurevich Y. The Classical Decision Problem[M]. Berlin: Springer, 2001:17-285.

[20] Gradel E, Kolaitis P, Vardi M. On the decision problem for two-variable first-order logic[J]. The Bulletin of Symbolic Logic, 1997, 3(1):53-69.

[21] Gradel E, Walukiewicz I. Guarded fixed point logic[C]// Proceedings of 14th Symposium on Logic in Computer Science (LICS), Trento, Italy, 1999:45-54.

[22] Gabbay D M. Expressive functional completeness intense logic[DB/OL]. [2015-06-23]. https://www.researchgate.net/publication/268658624_Expressive_Functional_Completeness_in_Tense_Logic_Preliminary_report.

[23] Scott D. A decision method for validity of sentences in two variables[J]. Journal of Symbolic Logic, 1962, 27(4):477-478.

[24] 张兴亮,郭立红,张传胜,等. CO_2 激光器高压脉冲触发系统的设计[J]. 中国光学, 2012, 5(4):416-422.
Zhang Xing-liang, Guo Li-hong, Zhang Chuan-sheng, et al. Design of high-voltage pulse trigger system for CO_2 laser[J]. Chinese Optics, 2012, 5(4):416-422.

[25] Etessami K, Vardi M Y, Wilke T. First-order logic with two variables and unary temporal logic[J]. Information and Computation, 2002, 179(2):279-295.

[26] 张旭霞,李斌,张黎明,等. 有机-无机复合纳米材料的传感应用及机理[J]. 中国光学, 2015, 8(4):651-666.
Zhang Xu-xia, Li Bin, Zhang Li-ming, et al. Sensing application and mechanism of organic-inorganic nanocomposites[J]. Chinese Optics, 2015, 8(4):651-666.