

描述逻辑 μ ALCIO 的语义及推理

蒋运承^{1),2)} 王 驹³⁾ 邓培民³⁾ 汤 庸⁴⁾ 周生明³⁾

¹⁾(华南师范大学计算机学院 广州 510631)

²⁾(中国科学院软件研究所计算机科学国家重点实验室 北京 100190)

³⁾(广西师范大学计算机科学与信息工程学院 广西 桂林 541004)

⁴⁾(中山大学计算机科学系 广州 510275)

摘 要 循环术语集是描述逻辑长期以来的研究难点,它的最基本的问题即语义及推理问题没有得到合理的解决. 分析了描述逻辑循环术语集的研究现状和存在的问题,基于混合 μ -演算将不动点构造算子引入到含有枚举构造算子的描述逻辑 ALCIO 中,提出了一种允许包含循环术语集的描述逻辑 μ ALCIO. 给出了 μ ALCIO 的语法和语义,证明了 μ ALCIO 的可满足性推理等价于混合 μ -演算的可满足性推理,并利用树自动机理论给出了 μ ALCIO 的可满足性推理算法以及给出了推理算法正确性证明和复杂性定理.

关键词 描述逻辑; μ ALCIO; 混合 μ -演算; 树自动机; 不动点构造算子

中图法分类号 TP18

DOI号: 10.3724/SP.J.1016.2009.01280

Semantics and Reasoning of Description Logic μ ALCIO

JIANG Yun-Cheng^{1),2)} WANG Ju³⁾ DENG Pei-Min³⁾ TANG Yong⁴⁾ ZHOU Sheng-Ming³⁾

¹⁾(School of Computer Science, South China Normal University, Guangzhou 510631)

²⁾(State Key Laboratory of Computer Science, Institute of Software, Chinese Academy of Sciences, Beijing 100190)

³⁾(College of Computer Science and Information Engineering, Guangxi Normal University, Guilin, Guangxi 541004)

⁴⁾(Department of Computer Science, Sun Yat-Sen University, Guangzhou 510275)

Abstract Terminological cycles have been a difficult spot in the study of description logics for quite a few years. The basic problems, such as their semantics and reasoning mechanisms, have not been reasonably well settled. The current research progresses and the existing problems of terminological cycles in description logics are analyzed in this paper. Based on hybrid μ -calculus, description logic μ ALCIO which may include terminological cycles is presented, and μ ALCIO is derived from description logic ALCIO which includes nominal constructors by adding least and greatest fixpoint constructors. The syntax and semantics of description logic μ ALCIO are given. The equality between satisfiability of description logic μ ALCIO and satisfiability of hybrid μ -calculus is proved. The satisfiability reasoning algorithm of description logic μ ALCIO is presented using tree automata. The correctness of the satisfiability reasoning algorithm is proved, and the complexity property of the reasoning algorithm is given.

Keywords description logic; μ ALCIO; hybrid μ -calculus; tree automata; fixpoint constructors

收稿日期:2007-04-05;最终修改稿收到日期:2008-06-10. 本课题得到国家自然科学基金(60663001,60573010,60673135,60373081)、中国科学院计算机科学国家重点实验室开放课题基金(SYSKF0904)、广西自然科学基金(桂科青 0640030,桂科自 0991100,0832103)资助. 蒋运承,男,1974年生,博士,教授,主要研究方向为描述逻辑、语义 Web 和 Web 智能. E-mail: jiangyc@ics.ict.ac.cn. 王 驹,男,1950年生,博士,研究员,主要研究领域为描述逻辑、分离逻辑和人工智能. 邓培民,男,1950年生,教授,主要研究领域为自动机理论及应用. 汤 庸,男,1964年生,博士,教授,博士生导师,主要研究领域为时态数据库、知识工程和 CSCW. 周生明,男,1956年生,硕士,副教授,主要研究方向为数理逻辑和人工智能.

1 引言

循环术语集(也称循环定义)是描述逻辑长期以来的研究难点. 它的最基本的问题即语义及推理问题没有得到合理的解决^[1-3],甚至在已给出的方法中存在错误,例如,不动点语义是循环术语集的理论基础,描述逻辑论著《The Description Logic Handbook: Theory, Implementation and Applications》给出了两个有关循环术语集不动点模型存在的命题(即命题 2.8 和命题 2.9)^[4],这两个命题是循环术语集的非常基本的重要定理(用来判断是否存在不动点模型),我们已经证明命题 2.9 是错误的^[5].

由于描述逻辑循环术语集的语义及推理问题没有得到合理的解决,因此在目前已实现的描述逻辑推理系统中都给出了强制规定: TBox 不允许出现循环定义. 但循环定义可以扩充描述逻辑的表达能力,而且在许多实际应用中循环定义是不可避免的^[6-8]. 另外,循环定义还能够方便用户建立描述逻辑知识库,并使所表示的知识或公理符合人们的直觉,如果没有循环定义,则只能用非循环定义来描述相应的循环定义,这样会使知识库变得非常复杂,用户也很难理解^[8-9]. 因此,无论从理论上还是实际应用上讲,研究循环定义都相当有意义.

Nebel 最早深入研究了描述逻辑循环定义,提出了循环定义的 3 种语义:最大不动点语义、最小不动点语义和描述语义,并利用自动机给出了描述逻辑 TL 循环术语集的可满足性和包含推理算法^[8-9]. Baader 也利用自动机给出了描述逻辑 FL₀ 循环术语集的可满足性和包含推理算法^[6]. Baader 使用描述图给出了描述逻辑 ϵL 循环术语集的可满足性和包含推理算法^[10]. 可以看出, Nebel^[8-9] 和 Baader^[6,10] 都针对一些很小的不带否定构造算子的描述逻辑,如何将 Nebel^[8-9] 和 Baader^[6,10] 的结果直接推广(即不引入不动点构造算子)到包含否定构造算子的情形,仍然是一个未解决的问题.

为了循环定义能处理否定构造算子,许多学者将 μ -演算的不动点构造算子引入到描述逻辑中,用不动点构造算子来描述循环术语集,从而循环术语集可以同时采用 Nebel 提出的 3 种语义^[8-9]. 基于命题 μ -演算, Schild 最早将不动点构造算子引入到描述逻辑 ALC 中,提出了描述逻辑 μALC , 并给出了 μALC 的表达能力和推理复杂性定理^[11]. 基于模态 μ -演算, Giacomo 将不动点构造算子引入到描述逻辑

ALCQ 中,提出了描述逻辑 μALCQ , 证明了 μALCQ 的可满足性推理等价于模态 μ -演算的可满足性推理^[2]. 基于完全 μ -演算, Calvanese 将不动点构造算子引入到描述逻辑 ALCQI 和 DLR 中,提出了描述逻辑 μALCQI ^[4] 和 μDLR ^[12], 并利用树自动机给出了 μDLR 的可满足性推理算法^[12]. 但 μALC ^[11]、 μALCQ ^[2]、 μALCQI ^[4] 和 μDLR ^[12] 都不能处理枚举构造算子,而枚举构造算子是描述逻辑的一个重要构造算子^[13]. 为此,有必要将 μALC ^[11]、 μALCQ ^[2]、 μALCQI ^[4] 和 μDLR ^[12] 推广到含有枚举构造算子的描述逻辑中. 实际上已有这方面的工作,例如, Bonatti 提出了描述逻辑 $\mu\text{ALCIO}_{\text{fa}}$ ^[13], 但 $\mu\text{ALCIO}_{\text{fa}}$ 是不可判定的.

基于上述原因,本文基于混合 μ -演算^[14] 将不动点构造算子引入到含有枚举构造算子的描述逻辑 ALCIO 中,提出描述逻辑 μALCIO , 给出 μALCIO 的语法和语义,证明 μALCIO 的可满足性推理等价于混合 μ -演算的可满足性推理,并利用树自动机给出 μALCIO 的可满足性推理算法. 可以看出,与 Schild^[11]、Giacomo^[2] 和 Calvanese^[4,12] 的工作相比,本文的工作是在一个新的方向上的结果. 与不可判定的 $\mu\text{ALCIO}_{\text{fa}}$ ^[13] 相比,本文提出的 μALCIO 是一种可判定的描述逻辑.

2 基本概念

2.1 混合 μ -演算

混合 μ -演算是命题 μ -演算^[15] 的扩充,即增加了逆程序、全局关系和枚举个体^[14].

假设 AP 是原子命题集合, Var 是命题变量集合, Nom 是枚举个体集合, $Prog$ 是原子程序集合,并且存在一个全局程序 $o \in Prog$. 一个程序是一个原子程序,或是一个原子程序 $a \in Prog$ 的逆 a^- . 混合 μ -演算的公式是由下列规则生成的最小集合:

(1) true , false , p 和 $\neg p$ 是公式,其中 $p \in AP \cup Nom$;

(2) $x \in Var$ 是公式;

(3) 如果 φ_1 和 φ_2 是公式, α 是程序, y 是命题变量,则 $\varphi_1 \wedge \varphi_2$, $\varphi_1 \vee \varphi_2$, $\langle \alpha \rangle \varphi_1$, $[\alpha] \varphi_1$, $\mu y. \varphi_1(y)$ 和 $\nu y. \varphi_1(y)$ 是公式.

如果命题变量 $x \in Var$ 出现在公式 φ 的不动点算子(最小不动点算子 μ 或最大不动点算子 ν)的辖域之外,则称 x 是自由变量. 句子是不包含自由命题变量的公式,即变量 x 的每次出现都在不动点

算子 μ 或 ν 的辖域之内. 下面用 λ 表示不动点算子 μ 或 ν . 给定 λ -公式 $\lambda x. \varphi(x)$, $\varphi(\lambda x. \varphi(x))$ 表示用 $\lambda x. \varphi(x)$ 替换 φ 的每一个自由变量 x 后得到的公式.

混合 μ -演算的语义是通过 Kripke 结构 $K = (W, R, L)$ 定义的, 其中,

- (1) W 是点的集合;
- (2) $R: Prog \rightarrow 2^{W \times W}$ 将每个原子程序映射为 W 上的一个二元关系的集合;
- (3) $R(o) = W \times W$;
- (4) $L: AP \cup Nom \rightarrow 2^W$ 将每个原子命题或枚举个体映射为点的集合, 即对任意 $p \in AP \cup Nom$, $L(p)$ 表示 p 满足的点的集合. 其中对任意 $n \in Nom$, $L(n)$ 是单个元素的集合.

R 可以扩展到逆程序, 其中 $R(a^-) = \{(v, u) \mid (u, v) \in R(a)\}$.

给定一个 Kripke 结构 $K = (W, R, L)$, 变量 x_1, \dots, x_n , 赋值 $V: \{x_1, \dots, x_n\} \rightarrow 2^W$ 将每个变量映射为 W 的子集. 给定一个赋值 V 、一个变量 x 以及点的集合 $W' \subseteq W$, $V[x/W']$ 表示从 V 得到的一个赋值, 其中将 x 映射为 W' .

含有自由变量 x_1, \dots, x_n 的公式 φ 在 Kripke 结构 $K = (W, R, L)$ 上的解释是一个映射 φ^K , 将每个赋值 V 映射为 W 的子集 $\varphi^K(V)$, 其中映射 φ^K 归纳定义如下:

- (1) $\text{true}^K(V) = W$, $\text{false}^K(V) = \emptyset$;
- (2) 对任意 $p \in AP \cup Nom$, $p^K(V) = L(p)$, $(\neg p)^K(V) = W \setminus L(p)$;
- (3) $(\varphi_1 \wedge \varphi_2)^K(V) = (\varphi_1)^K(V) \cap (\varphi_2)^K(V)$;
- (4) $(\varphi_1 \vee \varphi_2)^K(V) = (\varphi_1)^K(V) \cup (\varphi_2)^K(V)$;
- (5) $(\langle \alpha \rangle \varphi)^K(V) = \{u \in W \mid \text{存在一个 } v \in W, \text{ 使得 } (u, v) \in R(\alpha) \text{ 和 } v \in \varphi^K(V)\}$;
- (6) $([\alpha] \varphi)^K(V) = \{u \in W \mid \text{对任意 } v \in W, \text{ 如果 } (u, v) \in R(\alpha), \text{ 则 } v \in \varphi^K(V)\}$;
- (7) $(\mu y. \varphi(y))^K(V) = \bigcap \{W' \subseteq W \mid \varphi^K(V[y/W']) \subseteq W'\}$;
- (8) $(\nu y. \varphi(y))^K(V) = \bigcup \{W' \subseteq W \mid \varphi^K(V[y/W']) \supseteq W'\}$.

因为句子不包含自由变量, 因而对句子的解释不需要引入赋值. 给定一个句子 ψ 、Kripke 结构 $K = (W, R, L)$ 以及 $w \in W$, ψ 在 K 中的 w 满足, 记为 $K, w \models \psi$, 当且仅当 $w \in \psi^K$, 并且称 K 是 ψ 的一个模型. 如果句子 ψ 存在一个模型, 则称 ψ 是可

满足的.

2.2 树自动机

首先给出无限树上两路交替自动机^[16]的有关概念.

无限树是自然数集合 IN 上前缀封闭的无限个字符串的集合. 形式上说, 一棵无限树是字符串 $T \subseteq IN^*$ 的集合, 并且满足: 如果 $x.c \in T$, 其中 $x \in IN^*$, $c \in IN$, 则 $x \in T$. 如果树是完全的, 则对任意 $0 < c' < c$, 有 $x.c' \in T$. T 的元素称为节点, 空字符串 ϵ 是 T 的根节点. 对任意 $x \in T$, 节点 $x.c$ 是 x 的后继节点, 其中 $c \in IN$. 约定: $x.0 = x$, $(x, i).-1 = x$ 以及 $\epsilon.-1$ 没有定义. 节点 x 的后继节点的个数称为 x 的分枝度 $d(x)$. 对任意节点 $x \in T$, 如果 $d(x) = k$, 则称树是 k -度树. T 的一条无限路径 P 是一个前缀封闭的集合 $P \subseteq T$, 并且满足: 对任意 $i \geq 0$, 存在唯一一个 $x \in P$, 使得 $|x| = i$, 其中 $|x|$ 表示节点 x 的层数, 即 x 到根结点的距离. 字母表 Σ 上的一棵标签树是一个序对 (T, V) , 其中 T 是一棵树, $V: T \rightarrow \Sigma$ 是一个映射.

下面给出无限树上的交替自动机的定义.

无限树上的交替自动机是非确定树自动机的扩展^[16]. 给定集合 X , 假设 $B^+(X)$ 是 X 上正布尔公式的集合, 即将 X 中的元素通过算子 \wedge (合取) 和 \vee (析取) 构造出来的布尔公式, 并且 true 和 false 也是公式. 给定集合 $Y \subseteq X$, 公式 $\theta \in B^+(X)$, Y 满足 θ , 当且仅当将 Y 中的元素赋值为 true , 将 $X \setminus Y$ 中的元素赋值为 false , 使得公式 θ 为真.

引入符号 $[k] = \{-1, 0, 1, \dots, k\}$. 无限 k -度 Σ -标签树上的一个两路交替自动机 (记为 2ATA 自动机) 是一个元组 $A = (\Sigma, Q, \delta, q_0, F)$, 其中 Σ 是输入字母表的集合, Q 是状态的有限集合, $\delta: Q \times \Sigma \rightarrow B^+([k] \times Q)$ 是转移函数, $q_0 \in Q$ 是起始状态, F 是接受条件.

k -度 Σ -标签树 (T, V) 上的一个 2ATA 自动机 A 的一个运行是一棵 $(T \times Q)$ -标签树 (T_r, r) , 即 T_r 中的每个节点用一个元素 $(x, q) \in T \times Q$ 来标注, 用来描述自动机 A 在状态 q 输入 T 的节点 x 时的一个复制. 注意: T_r 中的节点和 T 中的节点之间没有一一对应. 并且节点和后继节点 (即相邻节点) 上的标注必须满足转移函数. 形式上说, 一个运行是一棵 Σ_r -标签树 (T_r, r) , 其中 $\Sigma_r = T \times Q$, 并且 (T_r, r) 满足下列条件:

- (1) $\epsilon \in T_r$, $r(\epsilon) = (\epsilon, q_0)$;
- (2) 如果 $y \in T_r$, 并且 $r(y) = (x, q)$ 和 $\delta(q,$

$V(x)) = \theta$, 则存在一个集合(可能是空集) $S \subseteq [k] \times Q$, S 满足 θ , 并且对任意 $(c, q') \in S$, 存在节点 $y.i \in T_r$ 满足下列条件:

- ① 如果 $c = \epsilon$, 则 $r(y.i) = (x, q')$;
- ② 如果 $c \geq 1$, 则 $r(y.i) = (x.c, q')$;
- ③ 如果 $c = -1$, 则存在 $1 \leq i \leq k$, $x = x'.i$, 并且 $r(y.i) = (x', q')$.

一个运行 (T_r, r) 被接受, 当且仅当它的所有无限路径满足接受条件. 在基于混合 μ -演算的 2ATA 自动机中需要奇偶接受条件^[14,17]. 状态集 Q 上的一个奇偶接受条件是一个序列 $F = (G_1, G_2, \dots, G_m)$, 其中 $G_1 \subseteq G_2 \subseteq \dots \subseteq G_m = Q$. 给定一个运行 (T_r, r) 、一条无限路径 $P \subseteq T_r$, 集合 $\text{inf}(P) \subseteq Q$ 定义如下: $q \in \text{inf}(P)$, 当且仅当存在无限个 $y \in P$, 使得 $r(y) \in T \times \{q\}$. 也就是说, $\text{inf}(P)$ 仅仅包含路径 P 上出现无限次的状态的集合. 给定路径 P , 如果存在一个偶数 i , 使得 $\text{inf}(P) \cap G_i \neq \emptyset$ 以及 $\text{inf}(P) \cap G_{i-1} = \emptyset$, 则称路径 P 满足接受条件 F . 一个 2ATA 自动机 A 接受标签树 (T, V) , 当且仅当 A 存在一个接受标签树 (T, V) 的运行.

2ATA 自动机的一个重要推理问题是空问题推理, 即给定一个 2ATA 自动机 A , 是否存在一棵标签树 (T, V) , 使得 A 接受 (T, V) , 即 A 存在一个接受标签树 (T, V) 的运行. 已经证明: 2ATA 自动机的空问题推理是指数时间复杂的^[14,16].

3 描述逻辑 $\mu\text{ALC}\text{IO}$

3.1 语 法

描述逻辑 $\mu\text{ALC}\text{IO}$ 是描述逻辑 ALC ^[18] 的扩展, 即增加了逆构造算子、枚举个体构造算子和不动点构造算子, 因而 $\mu\text{ALC}\text{IO}$ 的语法是 ALC 的语法的扩充.

下面用 λ 表示最小不动点算子 μ 或最大不动点算子 ν . 将不动点算子 μ 或 ν 看成量词, 则辖域、变量的自由或受限出现、封闭公式的概念与标准的一阶逻辑相同.

$\mu\text{ALC}\text{IO}$ 的原始符号是原子概念(用 A 表示)、概念变量(简称变量, 用 X, Y, \dots 表示)或原子关系(用 P 表示). $\mu\text{ALC}\text{IO}$ 的概念(用 C 表示)和关系(用 R 表示)可以如下归纳定义:

$$\begin{aligned} C &\rightarrow A \mid \top \mid \bot \mid \neg C \mid C_1 \sqcap C_2 \mid C_1 \sqcup C_2 \mid \\ &\quad \exists R.C \mid \forall R.C \mid \{o\} \mid \mu X.C \mid \nu X.C \mid X; \\ R &\rightarrow UR \mid P \mid P^-. \end{aligned}$$

其中概念 $\lambda X.C$ 中的概念 C 必须是语法单调的, 也就是说, 概念 C 中变量 X 的每次自由出现必须在偶数个否定构造算子的辖域中, 从而可以保证概念 C 是单调操作, 因此 $\lambda X.C$ 的最小不动点和最大不动点唯一存在^[2,12].

最小不动点算子 μ 和最大不动点算子 ν 可以相互定义, 即 $\nu X.C = \neg \mu X. \neg C [X/\neg X]$, 其中 $C[X/\neg X]$ 表示将概念 C 中变量 X 的所有自由出现用 $\neg X$ 替换以后得到的概念.

3.2 语 义

$\mu\text{ALC}\text{IO}$ 的解释 $I = (\Delta^I, \cdot^I)$ 由解释论域 Δ^I 和解释函数 \cdot^I 构成, 其中解释函数 \cdot^I 将原子概念解释为解释论域 Δ^I 的子集, 原子关系解释为 $\Delta^I \times \Delta^I$ 上的子集. 因为 $\mu\text{ALC}\text{IO}$ 含有自由变量, 因而不能直接用解释函数 \cdot^I 来解释 $\mu\text{ALC}\text{IO}$ 的概念, 为此引入赋值的概念.

给定解释 I , I 上的赋值 ρ 是从变量到解释论域 Δ^I 的子集的映射. 给定赋值 ρ , $\rho[X/\epsilon]$ 表示 ρ 除了 $\rho[X/\epsilon](X) = \epsilon$. 也就是说, 对任意变量 Y :

$$\rho[X/\epsilon](Y) = \begin{cases} \epsilon, & \text{如果 } Y = X \\ \rho(X), & \text{如果 } Y \neq X \end{cases}$$

给定解释 I 以及 I 上的赋值 ρ , 可以通过扩充解释函数 \cdot_ρ^I 给出 $\mu\text{ALC}\text{IO}$ 的语义解释, 即 \cdot_ρ^I 将 $\mu\text{ALC}\text{IO}$ 的概念映射为解释论域 Δ^I 的子集, 关系映射为 $\Delta^I \times \Delta^I$ 上的子集, 定义如下:

- (1) $X_\rho^I = \rho(X) \subseteq \Delta^I$;
- (2) $A_\rho^I = A^I \subseteq \Delta^I$;
- (3) $\top_\rho^I = \Delta^I$;
- (4) $\bot_\rho^I = \emptyset$;
- (5) $(\neg C)_\rho^I = \Delta^I - C_\rho^I$;
- (6) $(C_1 \sqcap C_2)_\rho^I = (C_1)_\rho^I \cap (C_2)_\rho^I$;
- (7) $(C_1 \sqcup C_2)_\rho^I = (C_1)_\rho^I \cup (C_2)_\rho^I$;
- (8) $(\exists R.C)_\rho^I = \{s \in \Delta^I \mid \exists s'. (s, s') \in R^I \wedge s' \in C_\rho^I\}$;
- (9) $(\forall R.C)_\rho^I = \{s \in \Delta^I \mid \forall s'. (s, s') \in R^I \rightarrow s' \in C_\rho^I\}$;
- (9) $\{o\}_\rho^I = \{o_\rho^I\} = \{o\}$;
- (10) $(\mu X.C)_\rho^I = \bigcap \{\epsilon \subseteq \Delta^I \mid C_{\rho[X/\epsilon]}^I \subseteq \epsilon\}$;
- (11) $(\nu X.C)_\rho^I = \bigcup \{\epsilon \subseteq \Delta^I \mid \epsilon \subseteq C_{\rho[X/\epsilon]}^I\}$;
- (12) $(UR)_\rho^I = \Delta^I \times \Delta^I$;
- (13) $(P^-)_\rho^I = \{(s, s') \mid (s', s) \in P_\rho^I\}$.

说明: $C_{\rho[X/\epsilon]}^I$ 表示从 Δ^I 的子集 ϵ 到 Δ^I 的子集的操作, 由变量的语法单调条件可知, 操作 $C_{\rho[X/\epsilon]}^I$ 在集合包含 \subseteq 的条件下是单调的, 从而可以用 $\mu X.C$ 表示 $C_{\rho[X/\epsilon]}^I$ 的最小不动点, 用 $\nu X.C$ 表示 $C_{\rho[X/\epsilon]}^I$ 的最大

不动点。

给定概念 C , 如果存在一个 I 以及 I 上的赋值 ρ , 使得 $C_\rho^I \neq \emptyset$, 则称 C 是可满足的; 否则称 C 是不可满足的。给定概念 C_1 和 C_2 , 如果对任意解释 I 以及 I 上的赋值 ρ , 使得 $(C_1)_\rho^I \subseteq (C_2)_\rho^I$, 则称 C_2 包含 C_1 或 C_1 包含于 C_2 (记为 $C_1 \sqsubseteq C_2$)。

μ ALCIO 的 TBox 是概念包含公理 $C_1 \sqsubseteq C_2$ 的有限集合 (可能是空集), 其中 C_1 和 C_2 是 μ ALCIO 的封闭概念。如果 $C_1 \sqsubseteq C_2$ 和 $C_2 \sqsubseteq C_1$, 则记为 $C_1 \equiv C_2$, 称为概念等价公理。

给定解释 I 以及 I 上的任意赋值 ρ , 如果有 $(C_1)_\rho^I \subseteq (C_2)_\rho^I$, 则 I 满足概念包含公理 $C_1 \sqsubseteq C_2$ 。因为 C_1 和 C_2 是 μ ALCIO 的封闭概念, 因对 C_1 和 C_2 的解释不依赖于赋值 ρ , 也就是说, 给定解释 I , 如果有 $(C_1)^I \subseteq (C_2)^I$, 则 I 满足概念包含公理 $C_1 \sqsubseteq C_2$ 。给定解释 I 以及 μ ALCIO 的 TBox T , 如果 I 满足 T 的所有概念包含公理, 则 I 是 T 的一个模型。如果 TBox T 存在一个模型, 则称 T 是可满足的。给定 TBox T , 对 T 的任意模型 I 以及 I 上的任意赋值 ρ , 如果有 $(C_1)_\rho^I \subseteq (C_2)_\rho^I$, 则称 T 蕴含概念包含公理 $C_1 \sqsubseteq C_2$, 记为 $T \models C_1 \sqsubseteq C_2$ 。

下面给出 μ ALCIO 的不动点构造算子的性质。

用 $C(X)$ 表示概念 C 含有自由变量 X (当然也可以含有其它自由变量), 用 $C(D)$ 表示 $C(X)[X/D]$, 其中 D 是一个概念, 即 $C(D)$ 表示将概念 D 替换 $C(X)$ 中所有自由出现的变量 X 后得到的概念。

与 μ ALCQ^[2] 一样, μ ALCIO 也具有下列性质:

(1) 如果变量 Y 是 $C(X)$ 中的自由变量, 则 $\lambda X.C(X) \equiv \lambda Y.C(Y)$ 。

(2) 如果变量 X 在概念 C 中不出现, 则 $\lambda X.C \equiv C$ 。

(3) 因为 $\lambda X.C(X)$ 是一个不动点, 从而有 $C(\lambda X.C(X)) \equiv \lambda X.C(X)$ 。

(4) $\mu X.C(X) \sqsubseteq \nu X.C(X)$ 。

下面给出两个定理, 用来说明 μ ALCIO 的不动点构造算子的性质, 这些性质是 μ ALCQ^[1] 的性质的推广。

定理 1. 给定 μ ALCIO 的 TBox T 、概念 C 和 D , 并且概念 C 和 D 中变量 X 的每次自由出现在偶数个否定构造算子的辖域中, 如果 $T \models C \sqsubseteq D$, 则 $T \models \lambda X.C \sqsubseteq \lambda X.D$ 。

证明。因为概念 C 和 D 中变量 X 的每次自由出现在偶数个否定构造算子的辖域中, 从而 $\lambda X.C$ 和 $\lambda X.D$ 存在。下面分两种情况证明: 变量 X 是概

念 C 或 D 中的自由变量, 变量 X 不是概念 C 和 D 中的自由变量。

如果变量 X 不是概念 C 和 D 中的自由变量, 则变量 X 出现在不动点算子 λ 的辖域之内。不妨假设 $C = \lambda X.C_1$, $D = \lambda X.D_1$, 由 $T \models C \sqsubseteq D$ 可知 $T \models \lambda X.C_1 \sqsubseteq \lambda X.D_1$ 。又因为 $\lambda X.(\lambda X.C_1) = \lambda X.C_1$, $\lambda X.(\lambda X.D_1) = \lambda X.D_1$, 从而有 $T \models \lambda X.(\lambda X.C_1) \sqsubseteq \lambda X.(\lambda X.D_1)$, 即 $T \models \lambda X.C \sqsubseteq \lambda X.D$ 。

如果变量 X 是概念 C 或 D 中的自由变量, 用反证法证明。

因为 $T \models C \sqsubseteq D$, 所以对 T 的任意模型 I 以及 I 上的任意赋值 ρ , $(C_1)_\rho^I \subseteq (C_2)_\rho^I$ 。假设存在 T 的一个模型 I 以及 I 上的一个赋值 ρ , 有 $(\lambda X.C)_\rho^I \not\subseteq (\lambda X.D)_\rho^I$ 。下面仅证明 $\lambda = \mu$ 的情况, $\lambda = \nu$ 的情况类似可证。

因为 $(\mu X.C)_\rho^I \not\subseteq (\mu X.D)_\rho^I$, 则存在个体 $s \in (\mu X.C)_\rho^I$, 并且 $s \notin (\mu X.D)_\rho^I$ 。又因为 $(\mu X.C)_\rho^I = \bigcap \{ \epsilon \sqsubseteq \Delta^I \mid C_{\rho[X/\epsilon]}^I \subseteq \epsilon \}$, 从而 $s \in (\mu X.C)_\rho^I$, 当且仅当 $s \in \bigcap \{ \epsilon \sqsubseteq \Delta^I \mid C_{\rho[X/\epsilon]}^I \subseteq \epsilon \}$, 当且仅当 $\forall \epsilon \sqsubseteq \Delta^I. (C_{\rho[X/\epsilon]}^I \subseteq \epsilon \rightarrow s \in \epsilon)$ 。又因为 $(\mu X.D)_\rho^I = \bigcap \{ \epsilon \sqsubseteq \Delta^I \mid D_{\rho[X/\epsilon]}^I \subseteq \epsilon \}$, 从而 $s \notin (\mu X.D)_\rho^I$, 当且仅当 $s \notin \bigcap \{ \epsilon \sqsubseteq \Delta^I \mid D_{\rho[X/\epsilon]}^I \subseteq \epsilon \}$, 当且仅当 $\exists \epsilon' \sqsubseteq \Delta^I. (D_{\rho[X/\epsilon']}^I \subseteq \epsilon' \wedge s \notin \epsilon')$ 。又因为 $(C_1)_\rho^I \subseteq (C_2)_\rho^I$, 从而 $C_{\rho[X/\epsilon']}^I \subseteq D_{\rho[X/\epsilon']}^I \subseteq \epsilon'$, 因此 $s \notin \epsilon'$ 。又因为 $\forall \epsilon \sqsubseteq \Delta^I. (C_{\rho[X/\epsilon]}^I \subseteq \epsilon \rightarrow s \in \epsilon)$ 和 $C_{\rho[X/\epsilon']}^I \subseteq \epsilon'$, 所以 $s \in \epsilon'$, 与 $s \notin \epsilon'$ 矛盾。

证毕。

定理 2. 给定 μ ALCIO 的 TBox T 以及概念 $D(X)$, 对 μ ALCIO 的任意概念 C_1 和 C_2 , 如果 $T \models C_1 \sqsubseteq C_2$, 则

(1) 当概念 $D(X)$ 中变量 X 的每次自由出现在偶数个否定构造算子的辖域中时, 有 $T \models D(C_1) \sqsubseteq D(C_2)$;

(2) 当概念 $D(X)$ 中变量 X 的每次自由出现在奇数个否定构造算子的辖域中时, 有 $T \models D(C_2) \sqsubseteq D(C_1)$ 。

证明。通过对概念 $D(X)$ 的结构进行归纳证明。

如果 $D(X) = X$, 则 $D(X)$ 中变量 X 的自由出现在偶数个 (即 0 个) 否定构造算子的辖域中, 并且 $D(C_1) = C_1$, $D(C_2) = C_2$ 。由 $T \models C_1 \sqsubseteq C_2$ 可知 $T \models D(C_1) \sqsubseteq D(C_2)$ 。

如果 $D(X) = \{o\}$, 则 $D(X)$ 中不含自由变量 X , 即变量 X 的自由出现在偶数个 (即 0 个) 否定构造算

子的辖域中,并且 $D(C_1) = \{o\}$, $D(C_2) = \{o\}$. 因此 $T \models D(C_1) \equiv D(C_2)$, 从而有 $T \models D(C_1) \sqsubseteq D(C_2)$.

如果 $D(X) = \neg D'(X)$, 则有两种情况: ① 如果 $D(X)$ 中变量 X 的自由出现在偶数个否定构造算子的辖域中, 则 $D'(X)$ 中变量 X 的自由出现在奇数个否定构造算子的辖域中; ② 如果 $D(X)$ 中变量 X 的自由出现在奇数个否定构造算子的辖域中, 则 $D'(X)$ 中变量 X 的自由出现在偶数个否定构造算子的辖域中. 对于第①种情况, 由归纳假设可知 $T \models D'(C_2) \sqsubseteq D'(C_1)$. 由构造算子 \neg 的语义可知 $T \models \neg D'(C_1) \sqsubseteq \neg D'(C_2)$, 即 $T \models D(C_1) \sqsubseteq D(C_2)$. 对于第②种情况, 由归纳假设可知 $T \models D'(C_1) \sqsubseteq D'(C_2)$. 由构造算子 \neg 的语义可知 $T \models \neg D'(C_2) \sqsubseteq \neg D'(C_1)$, 即 $T \models D(C_2) \sqsubseteq D(C_1)$.

如果 $D(X) = D_1(X) \sqcap D_2(X)$, 则有两种情况: ① 如果 $D(X)$ 中变量 X 的自由出现在偶数个否定构造算子的辖域中, 则 $D_1(X)$ 和 $D_2(X)$ 中变量 X 的自由出现都在偶数个否定构造算子的辖域中. ② 如果 $D(X)$ 中变量 X 的自由出现在奇数个否定构造算子的辖域中, 则 $D_1(X)$ 和 $D_2(X)$ 中变量 X 的自由出现都在奇数个否定构造算子的辖域中. 对于第①种情况, 因为 $D_1(X)$ 和 $D_2(X)$ 中变量 X 的自由出现都在偶数个否定构造算子的辖域中, 由归纳假设可知 $T \models D_1(C_1) \sqsubseteq D_1(C_2)$ 和 $T \models D_2(C_1) \sqsubseteq D_2(C_2)$, 从而 $T \models D_1(C_1) \sqcap D_2(C_1) \sqsubseteq D_1(C_2) \sqcap D_2(C_2)$, 即 $T \models D(C_1) \sqsubseteq D(C_2)$. 对于第②种情况, 因为 $D_1(X)$ 和 $D_2(X)$ 中变量 X 的自由出现都在奇数个否定构造算子的辖域中, 由归纳假设可知 $T \models D_1(C_2) \sqsubseteq D_1(C_1)$ 和 $T \models D_2(C_2) \sqsubseteq D_2(C_1)$, 从而 $T \models D_1(C_2) \sqcap D_2(C_2) \sqsubseteq D_1(C_1) \sqcap D_2(C_1)$, 即 $T \models D(C_2) \sqsubseteq D(C_1)$.

如果 $D(X) = D_1(X) \sqcup D_2(X)$, $\exists R.D'(X)$ 或 $\forall R.D'(X)$, 类似可证.

如果 $D(X) = \lambda Y.D'(X)$, 其中 $Y \neq X$, 由 $\lambda X.C(X)$ 的定义可知, $D'(X)$ 中变量 Y 的每次自由出现在偶数个否定构造算子的辖域中, 从而由归纳假设可知, $T \models D'(C_1) \sqsubseteq D'(C_2)$. 又由定理 1 可知 $T \models \lambda Y.D'(C_1) \sqsubseteq \lambda Y.D'(C_2)$, 即 $T \models D(C_1) \sqsubseteq D(C_2)$.
证毕.

4 $\mu\text{ALC}\text{IO}$ 的推理

为了给出 $\mu\text{ALC}\text{IO}$ 的可满足性推理算法, 下面首先给出 $\mu\text{ALC}\text{IO}$ 与混合 μ -演算之间的对应关系.

因为 $\mu\text{ALC}\text{IO}$ 的 TBox 中只包含封闭概念, 因而下面仅给出封闭概念的可满足性推理问题.

4.1 $\mu\text{ALC}\text{IO}$ 与混合 μ -演算

下面给出 $\mu\text{ALC}\text{IO}$ 的概念与混合 μ -演算的公式之间的对应关系, 该对应关系是通过下列函数 u 来归纳定义的:

$$\begin{aligned} u(\top) &= \text{true}; \\ u(\perp) &= \text{false}; \\ u(A) &= A; \\ u(X) &= X; \\ u(\{o\}) &= \{o\}; \\ u(C_1 \sqcap C_2) &= u(C_1) \wedge u(C_2); \\ u(C_1 \sqcup C_2) &= u(C_1) \vee u(C_2); \\ u(\neg C) &= \neg u(C); \\ u(\mu X.C) &= \mu X.u(C); \\ u(\nu X.C) &= \nu X.u(C); \\ u(\exists R.C) &= \langle R \rangle u(C); \\ u(\forall R.C) &= [R]u(C). \end{aligned}$$

可以看出, 对 $\mu\text{ALC}\text{IO}$ 的任意概念 C , 将概念 C 转化为混合 μ -演算的公式 $u(C)$ 是多项式时间复杂的, 复杂度为 $O(|C|)$. 下面证明函数 u 的正确性.

定理 3. 给定 $\mu\text{ALC}\text{IO}$ 的任意概念 C , $u(C)$ 是通过函数 u 转化后得到的混合 μ -演算的公式, 则 C 是可满足的, 当且仅当 $u(C)$ 是可满足的.

证明. 先证明 \Rightarrow . 如果 C 是可满足的, 则存在一个解释 $I = (\Delta^I, \cdot^I)$ 以及 I 上的赋值 ρ , 使得 $C_\rho^I \neq \emptyset$. 由解释 $I = (\Delta^I, \cdot^I)$ 和 I 上的赋值 ρ 可以如下得到混合 μ -演算的 Kripke 结构 $K = (W^K, R^K, L^K)$ 和赋值 V^K , 其中假设 RS 是 $\mu\text{ALC}\text{IO}$ 的关系集合, AS 是 $\mu\text{ALC}\text{IO}$ 的原子概念集合, OS 是 $\mu\text{ALC}\text{IO}$ 的枚举概念集合:

$$\begin{aligned} W^K &= \Delta^I; \\ R^K &= \{R^I : RS \rightarrow 2^{\Delta^I \times \Delta^I}\}; \\ L^K &= \{A^I : AS \cup OS \rightarrow 2^{\Delta^I}\}; \\ V^K &= \rho. \end{aligned}$$

对 $\mu\text{ALC}\text{IO}$ 的任意概念 C , 对 C 的结构进行归纳证明.

如果 $C = \top$, 则 $u(C) = \text{true}$, 从而 $(u(C))^K = (\text{true})^K = L^K(\top) = \top^I = \Delta^I = W^K$. 所以对任意 $w \in W^K$, 有 $w \in (u(C))^K$, 即 K 是 $u(C)$ 的一个模型, 从而 $u(C)$ 是可满足的.

因为 C 是可满足的, 所以 $C \neq \perp$.

如果 $C = X$, 则 $u(C) = X$, $(u(C))^K (V^K) = X^K (V^K) = \rho(X) = X_\rho^I \subseteq \Delta^I$, 因为 $C_\rho^I \neq \emptyset$, 所以存在

$w \in W^K$, 有 $w \in (u(C))^K(V^K)$, 即 K 是 $u(C)(V^K)$ 的一个模型, 从而 $u(C)$ 是可满足的.

如果 C 是原子概念 A , 即 $u(C) = A$, 则 $(u(C))^K(V^K) = (A)^K(V^K) = L^K(A) = A^I$. 因为 $C_\rho^I \neq \emptyset$, 所以 $A^I \neq \emptyset$. 从而存在 $w \in W^K$, 有 $w \in (u(C))^K(V^K)$, 即 K 是 $u(C)(V^K)$ 的一个模型, 从而 $u(C)$ 是可满足的.

如果 $C = \neg D$, 则 $u(C) = u(\neg D) = \neg u(D)$, 从而 $(u(C))^K(V^K) = (\neg u(D))^K(V^K) = W^K \setminus (u(D))^K(V^K)$. 由归纳假设可知, $(u(D))^K(V^K) = D_\rho^I \subseteq \Delta^I$. 因为 $C_\rho^I \neq \emptyset$, 所以 $(\neg D)_\rho^I \neq \emptyset$, 即 $\Delta^I \setminus D_\rho^I \neq \emptyset$. 从而有 $W^K \setminus (u(D))^K(V^K) \neq \emptyset$. 因此存在 $w \in W^K$, 有 $w \in (u(C))^K(V^K)$, 即 K 是 $u(C)(V^K)$ 的一个模型, 从而 $u(C)$ 是可满足的.

如果 $C = \{o\}$, 则 $u(C) = \{o\}$, $(u(C))^K(V^K) = (\{o\})^K(V^K) = L^K(\{o\}) = \{o\}_\rho^I = \{o_\rho^I\} = \{o\}$. 因为 $C_\rho^I \neq \emptyset$, 所以 $\{o\}_\rho^I = \{o_\rho^I\} = \{o\} \in \Delta^I$. 从而存在 $w \in W^K$, 有 $w \in (u(C))^K(V^K)$, 即 K 是 $u(C)(V^K)$ 的一个模型, 从而 $u(C)$ 是可满足的.

如果 $C = C_1 \sqcap C_2$, 则 $u(C) = u(C_1 \sqcap C_2) = u(C_1) \wedge u(C_2)$. 由归纳假设可知, $(u(C_1))^K(V^K) = (C_1)_\rho^I \subseteq \Delta^I$, $(u(C_2))^K(V^K) = (C_2)_\rho^I \subseteq \Delta^I$. 因为 $C_\rho^I \neq \emptyset$, 所以 $(C_1 \sqcap C_2)_\rho^I \neq \emptyset$, 从而 $(u(C_1))^K(V^K) \cap (u(C_2))^K(V^K) \neq \emptyset$, 因此 $(u(C))^K(V^K) = (u(C_1 \sqcap C_2))^K(V^K) = (u(C_1))^K(V^K) \cap (u(C_2))^K(V^K) \neq \emptyset$, 即存在 $w \in W^K$, 有 $w \in (u(C))^K(V^K)$, 即 K 是 $u(C)(V^K)$ 的一个模型, 从而 $u(C)$ 是可满足的.

如果 $C = C_1 \sqcup C_2$, 类似可证.

如果 $C = \exists R.D$, 则 $u(C) = u(\exists R.D) = \langle R \rangle u(D)$. 由归纳假设可知, $(u(D))^K(V^K) = D_\rho^I \subseteq \Delta^I$. 因为 $C_\rho^I \neq \emptyset$, 所以 $(\exists R.D)_\rho^I \neq \emptyset$, 即 $\{s \in \Delta^I \mid \exists s'. (s, s') \in R_\rho^I \wedge s' \in D_\rho^I\} \neq \emptyset$. 因为 $(\langle R \rangle u(D))^K(V^K) = \{s \in W^K \mid \exists t \in W^K. (s, t) \in R^K(R) \wedge t \in u(D)^K(V^K)\} = \{s \in \Delta^I \mid \exists t \in \Delta^I. (s, t) \in R^K(R) \wedge t \in D_\rho^I\} = \{s \in \Delta^I \mid \exists t \in \Delta^I. (s, t) \in R_\rho^I \wedge t \in D_\rho^I\} \neq \emptyset$, 即存在 $w \in W^K$, 有 $w \in (u(C))^K(V^K)$, 即 K 是 $u(C)(V^K)$ 的一个模型, 从而 $u(C)$ 是可满足的.

如果 $C = \forall R.D$, 类似可证.

如果 $C = \mu X.D$, 则 $u(C) = u(\mu X.D) = \mu X.u(D)$. 由归纳假设可知, $(u(D))^K(V^K) = D_\rho^I \subseteq \Delta^I$. 因为 $C_\rho^I \neq \emptyset$, 所以 $(\mu X.D)_\rho^I \neq \emptyset$, 即 $\bigcap \{\varepsilon \subseteq \Delta^I \mid D_{\rho[X/\varepsilon]}^I \subseteq \varepsilon\} \neq \emptyset$. 因为 $(\mu X.u(D))^K(V^K) = \bigcap \{W' \subseteq W^K \mid u(D)^K(V^K[X/W']) \subseteq W'\} = \bigcap \{W' \subseteq \Delta^I \mid$

$L(D)^K(V^K[X/W']) \subseteq W'\} = \bigcap \{W' \subseteq \Delta^I \mid D_{\rho[X/\varepsilon]}^I \subseteq W'\} \neq \emptyset$, 即存在 $w \in W^K$, 有 $w \in (u(C))^K(V^K)$, 即 K 是 $u(C)(V^K)$ 的一个模型, 从而 $u(C)$ 是可满足的.

如果 $C = \nu X.D$, 类似可证.

再证明 \Leftarrow . 如果 $u(C)$ 是可满足的, 则存在 Kripke 结构 $K = (W^K, R^K, L^K)$ 和赋值 V^K , 使得存在 $w \in W^K$, 有 $w \in (u(C))^K(V^K)$. 由混合 μ -演算的 Kripke 结构 $K = (W^K, R^K, L^K)$ 和赋值 V^K 可以如下得到 μ ALCIO 的解释 $I = (\Delta^I, \cdot^I)$ 和 I 上的赋值 ρ , 其中假设 AP 是原子命题的集合, Var 是命题变量的集合, Nom 是枚举个体的集合, $Prog$ 是原子程序的集合:

$$\Delta^I = W^K;$$

$$R^I = \{R^K; Prog \rightarrow 2^{W \times W}\};$$

$$A^I = \{L^K; AP \cup Nom \rightarrow 2^W\};$$

$$\rho = V^K.$$

对混合 μ -演算的任意公式 $u(C)$, 对 $u(C)$ 的结构进行归纳证明.

如果 $u(C) = \text{true}$, 则 $C = \top$, 从而 $(C)^I = (\top)^I = L^K(\text{true}) = \text{true}^K = W^K = \Delta^I$. 所以存在一个解释 I , 使得 $C^I \neq \emptyset$, 从而 C 是可满足的.

因为 $u(C)$ 是可满足的, 所以 $u(C) \neq \text{false}$.

如果 $u(C) = X$, 则 $C = X$, $C_\rho^I = \rho(X) = X_\rho^I = X^K(V^K) = (u(C))^K(V^K)$, 因为 $u(C)$ 是可满足的, 所以存在 $w \in W^K$, 使得 $w \in (u(C))^K(V^K)$. 因此, 存在一个解释 $I = (\Delta^I, \cdot^I)$ 以及 I 上的赋值 ρ , 使得 $C_\rho^I \neq \emptyset$, 即 C 是可满足的.

如果 $u(C) = \neg u(D)$, 则 $C = \neg D$, $C_\rho^I = (\neg D)_\rho^I = \Delta^I \setminus D_\rho^I$. 由归纳假设可知, $D_\rho^I = (u(D))^K(V^K)$, 因而 $C_\rho^I = \Delta^I \setminus (u(D))^K(V^K)$. 因为 $u(C)$ 是可满足的, 所以存在 $w \in W^K$, 使得 $w \in (u(C))^K(V^K)$. 因此 $(u(C))^K(V^K) = (\neg u(D))^K(V^K) = W^K \setminus (u(D))^K(V^K) \neq \emptyset$, 从而 $C_\rho^I \neq \emptyset$, 即 C 是可满足的.

如果 $u(C) = \{o\}$, 则 $C = \{o\}$, $C_\rho^I = \{o\}_\rho^I = \{o_\rho^I\} = \{o\}$. 因为 $u(C)$ 是可满足的, 所以存在 $w \in W^K$, 使得 $w \in (u(C))^K(V^K)$. 又因为 $(u(C))^K(V^K) = (\{o\})^K(V^K) = L^K(\{o\}) = \{o\}$, 从而有 $\{o\}_\rho^I = \{o_\rho^I\} = \{o\} \in \Delta^I$, 因此 $C_\rho^I \neq \emptyset$, 即 C 是可满足的.

如果 $u(C) = u(C_1) \wedge u(C_2)$, 则 $C = C_1 \sqcap C_2$, $C_\rho^I = (C_1 \sqcap C_2)_\rho^I = (C_1)_\rho^I \cap (C_2)_\rho^I$. 由归纳假设可知, $(C_1)_\rho^I = (u(C_1))^K(V^K)$, $(C_2)_\rho^I = (u(C_2))^K(V^K)$. 因为 $u(C)$ 是可满足的, 所以存在 $w \in W^K$, 使得 $w \in (u(C))^K(V^K)$. 因为 $(u(C))^K(V^K) = (u(C_1 \sqcap C_2))^K(V^K) = (u(C_1))^K(V^K) \cap (u(C_2))^K(V^K) = (C_1)_\rho^I \cap (C_2)_\rho^I = C_\rho^I \neq \emptyset$, 即 C 是可满足的.

$(C_2)^K(V^K) = (u(C_1))^K(V^K) \cap (u(C_2))^K(V^K) = (C_1)_\rho^I \cap (C_2)_\rho^I$, 所以 $(C_1)_\rho^I \cap (C_2)_\rho^I \neq \emptyset$, 即 $C_\rho^I \neq \emptyset$, 从而 C 是可满足的.

如果 $u(C) = u(C_1) \vee u(C_2)$, 类似可证.

如果 $u(C) = \langle R \rangle u(D)$, 则 $C = \exists R.D$, $C_\rho^I = (\exists R.D)_\rho^I = \{s \in \Delta^I \mid \exists s'.(s, s') \in R_\rho^I \wedge s' \in D_\rho^I\}$. 由归纳假设可知, $D_\rho^I = (u(D))^K(V^K)$. 因为 $u(C)$ 是可满足的, 所以存在 $w \in W^K$, 使得 $w \in (u(C))^K(V^K)$. 又因为 $C_\rho^I = (\exists R.D)_\rho^I = \{s \in \Delta^I \mid \exists s'.(s, s') \in R_\rho^I \wedge s' \in D_\rho^I\} = \{s \in \Delta^I \mid \exists t \in \Delta^I. (s, t) \in R^K(R) \wedge t \in D_\rho^I\} = \{s \in W^K \mid \exists t \in W^K. (s, t) \in R^K(R) \wedge t \in u(D)^K(V^K)\} = (\langle R \rangle u(D))^K(V^K) = (u(C))^K(V^K)$, 所以 $C_\rho^I \neq \emptyset$, 从而 C 是可满足的.

如果 $u(C) = [R]u(D)$, 类似可证.

如果 $u(C) = \mu X.u(D)$, 则 $C = \mu X.D$, $C_\rho^I = (\mu X.D)_\rho^I = \cap \{\epsilon \subseteq \Delta^I \mid D_{\rho[X/\epsilon]}^I \subseteq \epsilon\}$. 由归纳假设可知, $D_\rho^I = (u(D))^K(V^K)$. 因为 $u(C)$ 是可满足的, 所以存在 $w \in W^K$, 使得 $w \in (u(C))^K(V^K)$. 又因为 $C_\rho^I = (\mu X.D)_\rho^I = \cap \{\epsilon \subseteq \Delta^I \mid D_{\rho[X/\epsilon]}^I \subseteq \epsilon\} = \cap \{W' \subseteq \Delta^I \mid L(D)^K(V^K[X/W']) \subseteq W'\} = \cap \{W' \subseteq W^K \mid u(D)^K(V^K[X/W']) \subseteq W'\} = (\mu X.u(D))^K(V^K) = (u(C))^K(V^K)$, 所以 $C_\rho^I \neq \emptyset$, 从而 C 是可满足的.

如果 $u(C) = \nu X.u(D)$, 类似可证. 证毕.

由定理 3 可以看出, $\mu\text{ALC}\text{IO}$ 的概念与混合 μ -演算的公式之间具有一一对应关系, 即对 $\mu\text{ALC}\text{IO}$ 的任意概念 C , 可以等价地将概念 C 转化为混合 μ -演算的公式 $u(C)$; 反之也成立. 由于混合 μ -演算的可满足性推理问题可以转化为 2ATA 自动机的空问题推理^[14], 从而可以利用 2ATA 自动机给出 $\mu\text{ALC}\text{IO}$ 的可满足性推理算法.

4.2 $\mu\text{ALC}\text{IO}$ 的推理

因为 $\mu\text{ALC}\text{IO}$ 包含枚举构造算子, 从而 $\mu\text{ALC}\text{IO}$ 没有树模型性质. 与混合 μ -演算^[14]一样, $\mu\text{ALC}\text{IO}$ 的概念的任何一个模型可以展开为一棵森林(即树的集合), 通过选择函数来实现可满足性推理. 下面首先给出几个相关定义和两个预备定理(定理 4 和定理 5).

给定 $\mu\text{ALC}\text{IO}$ 的任意封闭概念 C , C 的闭包 $cl(C)$ 是满足下列规则的最小封闭概念的集合:

- (1) $C \in cl(C)$;
- (2) 如果 $C' \in cl(C)$, 则 $\neg C' \in cl(C)$;
- (3) 如果 $C_1 \sqcap C_2 \in cl(C)$ 或 $C_1 \sqcup C_2 \in cl(C)$, 则 $\{C_1, C_2\} \subseteq cl(C)$;

(4) 如果 $\exists R.C' \in cl(C)$ 或 $\forall R.C' \in cl(C)$, 则 $C' \in cl(C)$;

(5) 如果 $\lambda X.C' \in cl(C)$, 则 $C'[X/\lambda X.C'] \in cl(C)$.

可以看出, $cl(C)$ 中元素的个数是随 $|C|$ 线性增长的.

给定 $\mu\text{ALC}\text{IO}$ 的任意封闭概念 C , C 的一个原子 at 是 C 的闭包 $cl(C)$ 中的公式的集合, 并且 at 满足下列性质:

- (1) 如果 $A \in AS \cup OS$, $A \in cl(C)$, 则 $A \in at$ 或 $\neg A \in at$, 并且 $\{A, \neg A\} \not\subseteq at$;
- (2) 如果 $C_1 \sqcap C_2 \in cl(C)$, 则 $C_1 \sqcap C_2 \in at$, 当且仅当 $\{C_1, C_2\} \subseteq at$;
- (3) 如果 $C_1 \sqcup C_2 \in cl(C)$, 则 $C_1 \sqcup C_2 \in at$, 当且仅当 $\{C_1, C_2\} \cap at \neq \emptyset$;
- (4) 如果 $\lambda X.C' \in cl(C)$, 则 $\lambda X.C' \in at$, 当且仅当 $C'[X/\lambda X.C'] \in at$.

概念 C 的所有原子的集合记为 $at(C)$.

可以看出, C 的原子 at 是 $cl(C)$ 中的一个一致子集.

给定 $\mu\text{ALC}\text{IO}$ 的任意封闭概念 C , C 的一个准模型 (I, π) 由解释 $I = (\Delta^I, \cdot^I)$ 和映射 $\pi: \Delta^I \rightarrow at(C)$ 组成, 并且满足下列性质:

- (1) 存在一个 $a_0 \in \Delta^I$, 使得 $C \in \pi(a_0)$;
- (2) 对任意 $A \in AS \cup OS$, 如果 $A \in \pi(a)$, 则 $a \in A^I$; 如果 $\neg A \in \pi(a)$, 则 $a \notin A^I$; 如果 $\{o\} \in \pi(a)$, 则 $\{o\}^I = \{a\}$;
- (3) 如果 $\exists R.C' \in \pi(a)$, 则存在 $b \in \Delta^I$, $(a, b) \in R^I$, 并且 $C' \in \pi(b)$;
- (4) 如果 $\forall R.C' \in \pi(a)$, 则对任意 $b \in \Delta^I$, $(a, b) \in R^I$, 有 $C' \in \pi(b)$.

可以看出, 概念 C 的准模型 (I, π) 对含有不动点构造算子的概念不解释, 对其它概念的解释与解释 I 相同.

给定封闭概念 C 的准模型 (I, π) , (I, π) 的选择函数 $ch: \Delta^I \times cl(C) \rightarrow \Delta^I \cup cl(C)$ 定义如下, 对任意的 $a \in \Delta^I$:

- (1) 如果 $C_1 \sqcup C_2 \in \pi(a)$, 则 $ch(a, C_1 \sqcup C_2) \in \{C_1, C_2\} \cap \pi(a)$;
- (2) 如果 $\exists R.C' \in \pi(a)$, 则存在 $b \in \Delta^I$, $(a, b) \in R^I$, $C' \in \pi(b)$, 使得 $ch(a, \exists R.C') = b$.

可以看出, 选择函数用来表示含有构造算子 \sqcup

或 \exists 的概念如何被满足.

一个修饰准模型 (I, π, ch) 由一个准模型 (I, π) 和一个选择函数 ch 组成.

给定 μ ALCIO 的任意封闭概念 C 以及 C 的修饰准模型 $(\Delta^I, \cdot^I, \pi, ch)$, 则可以如下定义修饰准模型中封闭概念之间的推导关系 $\vdash \subseteq (cl(C), \Delta^I)^2$:

(1) 如果 $C_1 \sqcup C_2 \in \pi(a)$, 则 $(C_1 \sqcup C_2, a) \vdash (ch(C_1 \sqcup C_2), a)$;

(2) 如果 $C_1 \sqcap C_2 \in \pi(a)$, 则 $(C_1 \sqcap C_2, a) \vdash (C_1, a)$ 和 $(C_1 \sqcap C_2, a) \vdash (C_2, a)$;

(3) 如果 $\exists R.C' \in \pi(a)$, 则 $(\exists R.C', a) \vdash (C', ch(\exists R.C', a))$;

(4) 如果 $\forall R.C' \in \pi(a)$, 则对任意 $b \in \Delta^I, (a, b) \in R^I$, 有 $(\forall R.C', a) \vdash (C', b)$;

(5) 如果 $\lambda X.C' \in \pi(a)$, 则 $(\lambda X.C', a) \vdash (C'[X/\lambda X.C'], a)$.

给定 μ ALCIO 的任意封闭概念 C 以及 C 的修饰准模型 $(\Delta^I, \cdot^I, \pi, ch)$, $a, b \in \Delta^I, \mu X.C'$ 是修饰准模型 $(\Delta^I, \cdot^I, \pi, ch)$ 中的最小不动点封闭概念, 如果存在一个序列 $(\beta_1, a_1), \dots, (\beta_k, a_k)$, 其中 $k \geq 1$, 使得 $\beta_1 = \beta_k = \mu X.C', a_1 = a, a_k = b$, 并且对任意 $1 \leq i < k$, $(\beta_i, a_i) \vdash (\beta_{i+1}, a_{i+1})$ 以及 $\mu X.C'$ 是每个 β_i 的子概念, 则称 $\mu X.C'$ 在修饰准模型 $(\Delta^I, \cdot^I, \pi, ch)$ 中从 a 到 b 所产生.

给定 μ ALCIO 的任意封闭概念 C 以及 C 的修饰准模型 $(\Delta^I, \cdot^I, \pi, ch)$, 如果不存在最小不动点封闭概念 $\mu X.C' \in cl(C)$ 以及无限序列 a_1, a_2, \dots , 使得对任意 $i \geq 0, \mu X.C'$ 从 a_i 到 a_{i+1} 所产生, 则称修饰准模型 $(\Delta^I, \cdot^I, \pi, ch)$ 是良构造的.

定理 4. 给定 μ ALCIO 的任意封闭概念 C , 则 C 有模型 $I = (\Delta^I, \cdot^I)$, 当且仅当 C 有良构造修饰准模型 $(\Delta^I, \cdot^I, \pi, ch)$.

证明. 由定理 3 可知 μ ALCIO 的概念 C 与混合 μ -演算的公式 $u(C)$ 具有一一等价关系, 即 C 是可满足的, 当且仅当 $u(C)$ 是可满足的. 又由定理 3 的证明过程可知, C 的模型 $I = (\Delta^I, \cdot^I)$ 与 $u(C)$ 的模型 $K = (W^K, R^K, L^K)$ 可以相互转化以及 C 的良构造修饰准模型 $(\Delta^I, \cdot^I, \pi, ch)$ 与 $u(C)$ 的良构造修饰准模型 (W^K, R^K, L^K, π, ch) 可以相互转化. 因此, C 有模型 $I = (\Delta^I, \cdot^I)$, 当且仅当 $u(C)$ 有模型 $K = (W^K, R^K, L^K)$; 以及 C 有良构造修饰准模型 $(\Delta^I, \cdot^I, \pi, ch)$, 当且仅当 $u(C)$ 有良构造修饰准模型 (W^K, R^K, L^K, π, ch) . 根据文献[14]引理 1 可知,

$u(C)$ 有模型 $K = (W^K, R^K, L^K)$, 当且仅当 $u(C)$ 有良构造修饰准模型 (W^K, R^K, L^K, π, ch) . 因此, C 有模型 $I = (\Delta^I, \cdot^I)$, 当且仅当 C 有良构造修饰准模型 $(\Delta^I, \cdot^I, \pi, ch)$. 证毕.

给定 μ ALCIO 的含有枚举个体 $\{o_1, \dots, o_m\}$ 的任意封闭概念 $C(o_1, \dots, o_m)$ 以及 $C(o_1, \dots, o_m)$ 的修饰准模型 $(\Delta^I, \cdot^I, \pi, ch)$, 则 $(\Delta^I, \cdot^I, \pi, ch)$ 的松弛由映射 R^r 和 π^r 组成, 其中 R^r 和 π^r 定义如下:

$R^r: RS \rightarrow 2^{\Delta^I \times \Delta^I}$, 其中 RS 是 μ ALCIO 的关系的集合;

$R^r: R \rightarrow R^I \setminus \{(a, b) \mid \text{存在 } 1 \leq i \leq m, \text{ 使得 } \{o_i\}^I = \{b\}\}$;

$\pi^r: \Delta^I \rightarrow \{G \mid G = G_1 \cup G_2, G_1 \in at(C), G_2 \subseteq \{\xrightarrow{R} o_i \mid R \text{ 在 } C \text{ 中出现}, R \neq UR, 1 \leq i \leq m\}\}$;

$\pi^r: a \rightarrow \pi(a) \cup \{\xrightarrow{R} \{o\} \mid (a, b) \in R^I, R \neq UR, \{o\}^I = \{b\}\}$.

定理 5. 给定 μ ALCIO 的任意封闭概念 C , 如果 C 是可满足的, 则 C 有一个良构造修饰准模型 $(\Delta^I, \cdot^I, \pi, ch)$, 该良构造修饰准模型 $(\Delta^I, \cdot^I, \pi, ch)$ 的松弛是一棵森林, 并且 C 标注该森林中某棵树的根节点.

证明. 由定理 3 可知 μ ALCIO 的概念 C 与混合 μ -演算的公式 $u(C)$ 具有一一等价关系, 因而直接可以由文献[14]引理 2 得到. 证毕.

下面给出如何利用 2ATA 自动机对 μ ALCIO 的可满足性问题进行推理的算法.

给定 μ ALCIO 的含有枚举个体 $\{o_1, \dots, o_m\}$ 的任意封闭概念 $C(o_1, \dots, o_m)$, 假设 OS 表示 μ ALCIO 的枚举概念的集合, RS_C 是概念 C 中关系的集合, C 的一个猜想 $GU = (G, f, CO)$, 其中猜想列表 $G = (r_1, \dots, r_m)$, 连接 $CO \subseteq OS \times RS_C \times OS$, 猜想映射 $f: \{1, \dots, m\} \rightarrow \{1, \dots, m\}$, 对任意 $1 \leq i, j \leq m$, $\emptyset \not\subseteq r_i \subseteq cl(C)$ 或 $r_i = \perp$, 对任意 $j \neq f(i), o_i \in r_{f(i)}, o_i \notin r_j$, 如果 $OS \cap r_i = \emptyset$, 则 $r_i = \perp$, 并且 $(o_i, R, o_j) \in CO$, 当且仅当 $(o_j, R^-, o_i) \in CO$.

定理 6. 给定 μ ALCIO 的任意封闭概念 C , 对 C 的任意猜想 GU , 存在一个 2ATA 自动机 $A(C, GU)$, 满足下列性质:

(1) 如果 C 是可满足的, 则存在 C 的一个猜想 GU' , 使得 2ATA 自动机 $A(C, GU')$ 接受的语言不是空集;

(2) 如果一棵树 T 被 2ATA 自动机 $A(C, GU)$

接受,则去除 T 的根节点得到的森林是 C 的一个良构造修饰准模型的松弛;

(3) 2ATA 自动机 $A(C, GU)$ 的状态数目随 $|C|$ 线性增长.

证明. 由定理 3 可知 μ ALCIO 的概念 C 与混合 μ -演算的公式 $u(C)$ 具有一一等价关系,因而直接可以由文献[14]定理 1 可得. 证毕.

由定理 4、定理 5 和定理 6 可知, μ ALCIO 的概念可满足性推理可以通过多项式过程转化为 2ATA 自动机的空问题推理,因为 2ATA 自动机的空问题推理是指数时间复杂的^[14,16],因此有下列定理.

定理 7. μ ALCIO 的概念可满足性判定是指数时间复杂的.

5 结束语

分析了描述逻辑循环术语集的研究现状和存在的问题,基于混合 μ -演算将不动点构造算子引入到含有枚举构造算子的描述逻辑 ALCIO 中,提出了描述逻辑 μ ALCIO,给出了 μ ALCIO 的语法和语义,证明了 μ ALCIO 的可满足性推理等价于混合 μ -演算的可满足性推理,并利用树自动机给出了 μ ALCIO 的可满足性推理算法,给出了推理算法正确性证明和复杂性定理. 进一步工作主要是研究描述逻辑 μ ALCIO 的概念包含推理算法及其优化策略.

参 考 文 献

- [1] Jiang Yun-Cheng, Wang Ju, Deng Pei-Min, Tang Yong. Semantics and reasoning of terminological cycles in description logic FL^- . Chinese Journal of Computers, 2008, 31(2): 185-195(in Chinese)
(蒋运承, 王驹, 邓培民, 汤庸. 描述逻辑 FL^- 循环术语集的语义及推理. 计算机学报, 2008, 31(2): 185-195)
- [2] Giacomo G D, Lenzerini M. A uniform framework for concept definitions in description logics. Journal of Artificial Intelligence Research, 1997, 6(1): 87-110
- [3] Buchheit M, Donini F M, Nutt W, Schaerf A. A refined architecture for terminological systems; Terminology= schema+ views. Artificial Intelligence, 1998, 99(2): 209-260
- [4] Baader F, Calvanese D, McGuinness D, Nardi D, Patel-Schneider P. The Description Logic Handbook: Theory, Implementation and Applications. Cambridge: Cambridge University Press, 2003
- [5] Cao Fa-Sheng, Yu Quan, Wang Ju, Jiang Yun-Cheng. Condition of cyclic ALCN-Tbox exists model. Chinese Journal of Computers, 2008, 31(1): 16-23(in Chinese)
(曹发生, 余泉, 王驹, 蒋运承. 循环 ALCN-Tbox 具有模型的条件. 计算机学报, 2008, 31(1): 16-23)
- [6] Baader F. Using automata theory for characterizing the semantics of terminological cycles. Annals of Mathematics and Artificial Intelligence, 1996, 18(2-4): 175-219
- [7] Kusters R. Characterizing the semantics of terminological cycles in ALN using finite automata//Cohn A G, Schubert L, Shapiro S C eds. Proceedings of the 6th International Conference on Principles of Knowledge Representation and Reasoning. San Francisco: Morgan Kaufmann Publishers, 1998: 499-511
- [8] Nebel B. Terminological cycles: Semantics and computational properties//Sowa J F ed. Proceedings of the Principles of Semantic Networks. San Francisco: Morgan Kaufmann Publishers, 1991: 331-362
- [9] Nebel B. Reasoning and revision in hybrid representation systems//LNAI 422. Berlin: Springer-Verlag, 1990
- [10] Baader F. Terminological cycles in a description logic with existential restrictions//Gottlob G, Walsh T eds. Proceedings of the 18th International Joint Conference on Artificial Intelligence (IJCAI 2003). San Francisco: Morgan Kaufmann Publishers, 2003: 325-330
- [11] Schild K. Terminological cycles and the propositional μ -calculus//Doyle J, Sandewall E, Torasso P eds. Proceedings of the 4th International Conference on the Principles of Knowledge Representation and Reasoning. San Francisco: Morgan Kaufmann Publishers, 1994: 509-520
- [12] Calvanese D, Giacomo G D, Lenzerini M. Reasoning in expressive description logics with fixpoints based on automata on infinite trees//Dean T ed. Proceedings of the 16th International Joint Conference on Artificial Intelligence (IJCAI 1999). San Francisco: Morgan Kaufmann Publishers, 1999: 84-89
- [13] Bonatti P A, Peron A. On the undecidability of logics with converse, nominals, recursion and counting. Artificial Intelligence, 2004, 158(1): 75-96
- [14] Sattler U, Vardi M Y. The hybrid μ -calculus //Gore R, Leitsch A, Nipkow T eds. Proceedings of the 1st International Joint Conference on Automated Reasoning. LNCS 2083. London: Springer-Verlag, 2001: 76-91
- [15] Kozen D. Results on the propositional μ -calculus. Theoretical Computer Science, 1983, 27(3): 333-354
- [16] Vardi M Y. Reasoning about the past with two-way automata//Larsen K G, Skyum S, Winskel G eds. Proceedings of the 25th International Colloquium on Automata, Languages and Programming. LNCS 1443. London: Springer-Verlag, 1998: 628-641

[17] Thomas W. Languages, automata and logic//Rozenberg G, Salomaa A eds. Handbook of Formal Languages. New York: Springer-Verlag. 1997: 389-455

[18] Schmidt-Schauß M, Smolka G. Attributive concept descriptions with complements. Artificial Intelligence, 1991, 48 (1): 1-26



JIANG Yun-Cheng, born in 1974, Ph. D. , professor. His main research interests include description logics, semantic Web and Web intelligence.

WANG Ju, born in 1950, Ph. D. , professor. His main research interests include description logics, separation logic

and artificial intelligence.

DENG Pei-Min, born in 1950, professor. His main research interests include automata theory and its applications.

TANG Yong, born in 1964, Ph. D. , professor, Ph. D. supervisor. His main research interests include temporal database, knowledge engineering and CSCW.

ZHOU Sheng-Ming, born in 1956, M. S. , associate professor. His main research interests include mathematical logic and artificial intelligence.

Background

Description logics are a logical reconstruction of the frame-based knowledge representation languages, with the aim of providing a simple well-established declarative semantics to capture the meaning of structured representation of knowledge. Terminological cycles (or cyclic definitions) have been a difficult spot in the study of description logics for quite a few years. The basic problems, such as their semantics and reasoning mechanisms, have not been reasonably well settled. However, terminological cycles may greatly extend the expressive capability of the language. In some applications, terminological cycles are indispensable. Also, by using terminological cycles, one can establish a description logic knowledge base more conveniently because it would give the people an intuitive understanding of the axioms, sentences of the knowledge base. If we do not accept terminological cycles in the knowledge base, when a cyclic concept occurs in the application, it would make the system specification overly complex and hard to understand by a user. For these reasons, it is significant to study the semantics and the

algorithmic nature of terminological cycles. Based on hybrid μ -calculus, description logic μ ALCIO which may include terminological cycles is presented, and μ ALCIO is derived from description logic ALCIO which includes nominal constructors by adding least and greatest fixpoint constructors. The syntax and semantics of description logic μ ALCIO are given. The equality between satisfiability of description logic μ ALCIO and satisfiability of hybrid μ -calculus is proved. The satisfiability reasoning algorithm of description logic μ ALCIO is presented by using tree automata. The correctness of the satisfiability reasoning algorithm is proved, and the complexity property of the reasoning algorithm is given. The work is supported by the National Natural Science Foundation of China under grant Nos. 60663001, 60573010, 60673135 and 60373081, the Open Foundation of the State Key Laboratory of Computer Science of Chinese Academy of Sciences under grant No. SYSKF0904, and the Natural Science Foundation of Guangxi Province of China under grant Nos. 0640030, 0991100 and 0832103.