

限制推理及其应用

金 芝 胡守仁 (国防科技大学计算机系)

摘 要

建立在标准逻辑基础上的推理系统是单调的,但是这种单调系统在处理不完全信息和不断变化的情况时,常导致不可避免的失败,而几乎所有的人工智能应用系统都涉及到上述问题。因此对非单调推理的研究越来越引起人们的普遍关注。在众多的非单调理论中,基于一阶逻辑的限界理论由于其良好的数学理论基础而成为实现非单调推理的形式化系统。本文主要介绍了限界理论和限界推理的实现方法,以及它在常识推理方面的应用。

建立在标准逻辑基础上的推理系统是单调的,若一个为真的命题加入理论 T 得理论 T' ,如果 $T \vdash P$,则必有 $T' \vdash P$ 。也就是说,可证明为真的命题数目随条件的增加而增加。这种系统具有以下优点:

- 不必检查新命题与原有知识间的不相容性。
- 加入新命题时,不需要寻求失效的命题。

但是,这种单调系统在处理不完全信息和不断变化的情况时,常导致不可避免的失败,而几乎所有的人工智能应用系统都涉及到上述问题。因为,严格地说,人的认识,人的推理本身就是非单调的。因此,对非单调推理理论和应用的研究,近年来成为人工智能界的一个热门课题,人们相继提出了许多非单调推理的形式系统。如McCarthy的限制理论,McDermott和Doyle的非单调逻辑,Reiter的省缺理论和Moore的直觉主义逻辑等,其中,基于经典谓词逻辑的限制理论,由于具备良好的数学理论基础,成为一个较引人注目的非单调逻辑形式化系统。

本文主要介绍限制推理理论,实现算法以及它在常识推理方面的应用。

一、限制推理理论

限制推理的思想由McCarthy于1980年首先提出^[1]。其基本含义是,使特定谓词的外延尽可能接近“false”的外延(“false”的外延为空集),也就是将这些特定谓词的外延最小化。

直观上说,谓词限制是指从有穷子句出发,推导出的满足特定谓词 P 的实体就是所有满足 P 的实体。对特定谓词的限制过程,实际上是一个最小化过程。假设 T 为语言 L 中由有限个子句组成的一阶理论,若 $P = \{P_1, P_2, \dots, P_m\}$ 为需要最小化的谓词集, $Z = \{Z_1, Z_2, \dots, Z_n\}$ 为在最小化过程中按 P 发生变化的谓词,称为变量(这里 $P_i (1 \leq i \leq m)$, $Z_j (1 \leq j \leq n)$ 都是 T 中的谓词符号,且 $P_i \neq Z_j$),则在 T 中对 P 的最小化过程将一阶理论 $T = T(P, Z)$ 转化为表达能力更强的二阶理论 $CIRC(T, P, Z)$ 。

定义2.1 带有变量 Z 的一阶理论 T ,对最小化谓词 P 的限制 $CIRC(T, P, Z)$ 可定义为:

$$T(P, Z) \wedge \forall P', Z' [T(P', Z') \wedge \{P' \rightarrow P\} \rightarrow P' = P]$$

其中, $P'_1 = \{P'_1, P'_2, \dots, P'_m\}$, $Z' =$

$\{Z_1', Z_2', \dots, Z_m'\}$, $P' \rightarrow P$ 表示对每一组变量 x 和所有 $i \leq m$, 有 $(P_i'(x) \rightarrow P_i)$ 。

该定义表明, 在 $T(P, Z)$ 成立且 Z 中谓词的外延在最小化过程中可以变化的情况下, P 中谓词具有最小可能的外延。下面我们利用模型论的观点进一步阐明限制推理的概念。

定义2.2 设 M 和 N 是 T 的两个模型, 如果 M 和 N 有相同的论域, M 和 N 除 P 和 Z 中谓词外, 对其它所有谓词和常量都有相同的解释, 而且 M 中所有 P 谓词的外延都是 N 中 P 谓词外延的子集, 则有

$$M \leq N \text{ mod } (P, Z)$$

更形式化地有, 如果

$$(i) |M| = |N|$$

$$(ii) M[K] = N[K] \quad K \in P + Z$$

$$(iii) M[P_i] \subseteq N[P_i] \quad P_i \in P$$

则 $M \leq N \text{ mod } (P, Z)$

按定义, 关系 $\leq$ 是有序的, 因此用关系 $\leq$ 可定义 T 的最小模型。

定义2.3 T 的一个模型 M 是 T 的 (P, Z) -最小模型, 如果不存在 T 的 (P, Z) -最小模型 N , 使

$$N < M \text{ mod } (P, Z)$$

定义2.4^[1,2] 一个结构 M 是 $\text{CIRC}(T, P, Z)$ 的模型, 当且仅当 M 是 T 的 (P, Z) -最小模型。换句话说, 对 T 中任一公式 F 有

$$\text{CIRC}(T, P, Z) \models F$$

当且仅当对 T 的每一个 (P, Z) -最小模型 M 有 $M \models F$ 。

例2.5 设一个一阶理论 T 包括如下两个子句

$$\text{Bird}(\text{Tweety}).$$

$$\text{Bird}(\text{Tweety}) \wedge \text{Abnormal}(\text{Tweety}) \rightarrow \text{Fly}(\text{Tweety}).$$

其中, 第一个子句表示 Tweety 是鸟, 第二个子句表明 Tweety 只要没发生任何异常情况, 通常都会飞。

因为当前没有 Tweety 的异常信息, 按常识应该得出 Tweety 会飞的结论, 我们再来看

看限制推理的结论。设 $P = \{\text{Abnormal}\}$ (我们打算最小化异常性), $Z = \{\text{Fly}\}$ (Fly 作为变量, 它的外延依赖于谓词 Abnormal 的外延)。显然模型

$$M = \{\text{Bird}(\text{Tweety}), \text{Fly}(\text{Tweety})\}$$

为 T 的一个 (P, Z) -最小模型且包括在任一 T 的 (P, Z) -最小模型中, 则有

$$\text{CIRC}(T, P, Z) \models \text{Fly}(\text{Tweety})$$

二、限制推理的几个重要性质

Gelfond等^[5]提出并证明, 限制理论 $\text{CIRC}(T, P, Z)$ 且有下面几个重要的性质:

定理3.1^[3] 若公式 F 不含 Z 和 P^- 中的谓词, 则

$$\text{CIRC}(T, P, Z) \models F$$

当且仅当

$$T \models F$$

这里以及在以下的描述中 P^- 表示 P 中谓词的否定式。

定理3.2^[3] 若公式 F 为不含 Z 中谓词的基础公式, 则

$$\text{CIRC}(T, P, Z) \models F$$

当且仅当, 不存在基础子句 G , 使

$$i) G \text{ 不含 } Z + P^- \text{ 中的文字;}$$

$$ii) T \vdash \neg F \vee G \text{ 但 } T \not\models G.$$

定理3.3^[3] 若公式 F 是任一基础公式, 则

$$\text{CIRC}(T, P, Z) \models F$$

当且仅当, 要么 $T \vdash F$, 要么存在一个基础公式 G , 使

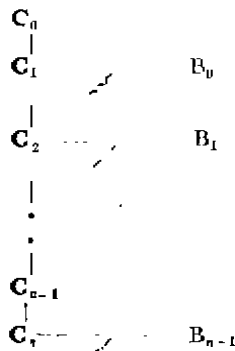
$$i) G \text{ 不含 } Z + P^- \text{ 中的谓词;}$$

$$ii) T \vdash F \vee G;$$

$$iii) \text{CIRC}(T, P, Z) \models \neg G$$

根据限制理论 $\text{CIRC}(T, P, Z)$ 的这些特性, 公式 F 在二阶理论 $\text{CIRC}(T, P, Z)$ 中是否成立的问题, 可以归结为一阶理论 T 中的证明问题。问题转化的关键在于寻找满足条件的基础公式 G 。为此, Przymusiński^[4]提出了最小模型归结 (Minimal-model-Linear-Ordered-resolution 或简称 $\text{MIL-Ordered-resolution}$) 的思想。

定义3.4 设一阶理论 T 为有序子句集, C_0 是一个有序子句,以 C_0 为顶从 $T+C_0$ 到 C_n 的MILO 演绎为满足下面条件的一个线性演绎



i) 每个 B_i 是 T 中的子句;

ii) 若 C_i 是可约的, 则 C_{i+1} 是 C_i 的约化有序子句, 约化过程中允许删除不紧跟非框的 P^+Z 中文字的任意被框文字; 若 C_i 不可约, 则 C_{i+1} 是有信息有序子句 C_i 与 T 中有信息有序子句 B_i 对 C_i 中属于 P^+Z 中文字的有序归结式;

iii) 演绎中没有重言式

定义3.5 一个 $T+C$ 的MILO 演绎子句, 若该子句不含 P^+Z 中文字, 则称该子句为 $T+C$ 的MILO-叶; 所有 $T+C$ 的MILO-叶的合取称为MILO导出, 记作 $\text{Deriv}(T, C)$. 若不存在MILO-叶, 则 $\text{Deriv}(T, C)=\text{true}$.

定义3.6 设 $H=C_1 \wedge \dots \wedge C_m$ 为子句的合取, 则

$$\text{Deriv}(T, H) = \text{Deriv}(T_1, C_1) \wedge \dots \wedge \text{Deriv}(T_m, C_m)$$

其中 $T_1=T$, $T_{i+1}=T_i+C_i (i=1, \dots, m-1)$.

显然 $\text{Deriv}(T, C)$ 为不含 P^+Z 中文字的公式。根据前述定理3.1—定理3.3, 可以证明下面三个结论:

结论1 若 F 为不含 P^+Z 中文字的基础公式, 则

$$\text{CIRC}(T, P, Z) \models F$$

当且仅当

$$T \models F.$$

结论2 若 F 为不含 Z 中文字的基础公式, 则

$$\text{CIRC}(T, P, Z) \models F$$

当且仅当

$$T \models \text{Deriv}(T, F).$$

结论3 对于含 Z 中文字的基础公式 F , 可以证明^[4]

$$\text{CIRC}(T, P, Z) \models F$$

当且仅当

$$\text{CIRC}(T, P, Z) \models \neg \text{Deriv}(T, \neg F).$$

三、限制推理算法

从上节得出的三个结论, 我们可将任一基础公式 F 在二阶理论 $\text{CIRC}(T, P, Z)$ 中可否满足的问题, 转化为一阶理论 T 中的证明问题, 而一阶理论中的证明问题, 可采用任一种完备的归结算法实现, 从而得出限制推理的结论。

首先我们描述最小模型归结算法如下:

算法4.1(MILO归结算法)

步骤1. 找出 T 中可与 C_0 对 C_0 中 P^+Z 文字归结的所有子句, 设它们是 $B_0^1, \dots, B_0^r$, 并将它们与 C_0 组成对

$$[C_0 B_0^1], \dots, [C_0 B_0^r]$$

令 CLIST 是以任意顺序排列的上述对的表。

步骤2. 若 $\text{CLIST}=[]$, 则转步骤6。

步骤3. 设 $[\text{CB}]$ 是 CLIST 中第一对, 从 CLIST 中删除 $[\text{CB}]$ 。

步骤4. 归结 C 和 B , 设 $R_1, \dots, R_m$ 是 C 与 B 对 C 中 P^+Z 文字的所有有序归结式. 将它们组成表 CLIST_1 , 并对表中所有归结式 R_i , 检查

1) 若 R_i 可约, 则用其约化子句替代;

2) 若 R_i 是重言式, 则删除 R_i ;

3) 若 R_i 被 CLIST 中某个对 CB 中的 C 所包含, 则删除 R_i ;

4) 若 R_i 不含 P^+Z 文字, 则将其存入表 CLEAF 中, 并从 CLIST_1 中删除 R_i 。

步骤5. 若 $\text{CLIST}_1=[]$, 则转步骤2, 否则对 CLIST_1 中的每一归结式 R_q^* , 找出 T 中可与 R_q^* 对 R_q^* 中 P^+Z 文字归结的所有子句。若不存在, 则删除 R_q^* , 否则设 $B_1^1, \dots,$

B_i, i 是所有这些子句, 组成对

$$\{R_1 * B_1\}, \dots, \{R_n * B_n\}$$

以任意顺序排列在表CLIST的前部, 转步骤2。

步骤6. 若CLEAF=[], 则Deriv(T, C)=true; 否则令Deriv(T, C)为CLEAF中文字的合取式。

有了MILO-算法, 按上节所述三个结论, 我们很容易写出限制推理的算法如下。其中设已知子句集合T, 要证明的基础子句F。

算法4.2[CIRC-算法]

按F中涉及谓词的情况分别执行:

1°. 若F中谓词不含P+Z文字, 则

1) 按一般归结算法证明 $T \vdash F$;

2) 若 $T \vdash F$, 则回答Yes, 否则回答No。

2°. 若F中谓词仅含P+中文字, 则

1) 调用算法4.1求出Deriv(T, F);

2) 按一般归结算法证明 $T \vdash \text{Deriv}(T, F)$;

3) 若2)成功, 则回答Yes, 否则回答No。

3°. 若F中谓词含Z中文字, 则

1) 令 $F = G_1 \vee \dots \vee G_m$, 则 $\neg F = \neg G_1 \wedge \dots \wedge \neg G_m$;

2) $T_1 = T$;

3) For $i=1, m$ do

调用算法4.1求Deriv($T_i, \neg G_i$)

$T_{i+1} = T_i + \neg G_i$

4) 设 $\neg [\text{Deriv}(T_1, \neg G_1) \wedge \dots \wedge \text{Deriv}(T_m, \neg G_m)]$

$= (B_{11} \vee \dots \vee B_{1n}) \wedge \dots \wedge (B_{m1} \vee \dots \vee B_{mn})$;

5) For $i=1, n$ do

调用算法4.2证明CIRC(T, P; Z)
 $\models B_{11} \vee \dots \vee B_{1n}$

若不成功, 回答No, 算法停止

6) 回答Yes, 算法停止。

下面我们用一个例子说明算法的执行。

例4.3 设理论T包括子句

Bird(Tweety)

Bird(Tweety) $\wedge$ $\neg$ Abnormal

[Tweety] $\rightarrow$ Fly(Tweety)

令 $P = \{\text{Abnormal}\}$, $Z = \{\text{Fly}\}$. 证明 CIRC(T; P; Z) $\models$ Fly(Tweety)。

由于公式中含Z中文字, 用3°. 首先求Deriv(T, $\neg F$).

$\neg$ Fly(Tweety)

|

$\neg$ Bird(Tweety) $\vee$ Abnormal
[Tweety]

则Deriv(T, $\neg F$) = $\neg$ Bird(Tweety) $\vee$ Abnormal(Tweety)

$\neg$ Deriv(T, $\neg F$) = Bird(Tweety)

$\wedge$ $\neg$ Abnormal(Tweety)

则原问题转化为证明

Deriv(T; P; Z) $\models \neg$ Deriv(T, $\neg F$)

$\models$ Bird(Tweety) $\wedge$ $\neg$ Abnormal(Tweety)

先证明CIRC(T; P; Z) $\models$ Bird(Tweety), 由于有 $T \vdash$ Bird(Tweety)成立, 按1°, 显然有CIRC(T; P; Z) $\models$ Bird(Tweety)成立。

再证明CIRC(T; P; Z) $\models \neg$ Abnormal(Tweety). 按2°, 先求Deriv(T, $\neg$ Abnormal(Tweety)).

$\neg$ Abnormal(tweety)

|

$\neg$ Bird(Tweety) $\vee$ Fly(Tweety)

由于归结树无叶, 则Deriv(T, $\neg$ Abnormal(Tweety)) = true. 显然有CIRC(T; P; Z) $\models$ true成立。

所以算法回答Yes, 即有

CIRC(T; P; Z) $\models$ Fly(Tweety)

可以证明, 算法4.2是可终止的, 它的执行效率与要证明的基础子句中含P+Z中文字的个数以及理论T中P+Z文字出现的频率有密切的关系。

四、限制推理的应用

第一代知识系统性能脆弱的原因之一, 就是缺乏对常识性知识的表示和推理能力,

在处理常识问题时显得极端笨拙和不可理解,甚至导致彻底的失败。限制推理对知识系统的最大贡献,就是它提供了一种形式化常识性知识的手段和一种常识推理的方法。与仅仅通过一阶逻辑公理来描述知识的系统相比,限制理论这种二阶理论更有能力表达象“一般”,“例外”等的情形。

人类的知识中,特别是经验知识中,经常出现“一般”,“例外”的情形,这里就涉及到可能出现的异常情况。用一阶逻辑表示和处理这类知识时,必须将所有异常情况排除,才能保证知识处理的正确性,但是,通常异常情况是非常琐碎和繁杂的,有时甚至是意想不到的,比如,一般情况下,鸟是会飞的,但死鸟和病鸟不能飞,折断了翅膀的鸟也不能飞。这段知识用一阶逻辑可表示如下:

$$\text{Bird}[x] \wedge \neg \text{Dead}[x] \rightarrow \text{Fly}[x]$$

$$\text{Bird}[x] \wedge \neg \text{Ill}[x] \rightarrow \text{Fly}[x]$$

$$\text{Bird}[x] \wedge \neg \text{further}[x] \rightarrow \text{Fly}[x]$$

另外还有很多其它意想不到的情况出现,使得鸟不能飞。这样就给这类常识知识的表示和推理带来困难,有时使知识处理走向不可避免的失败。

用限制理论我们很容易形式化这类常识知识。如,一般情况下鸟是会飞的这个断言,我们可以表示成:

$$\text{Bird}[x] \wedge \neg \text{Abnormal}[x] \rightarrow \text{Fly}[x]$$

用不出现异常 $\neg \text{Abnormal}[x]$ 表示通常情况,限制这个断言可用的范围,然后通过异常属性 Abnormal 的最小化,保证通常情况推出结论的正确性。这里的最小化就是指,没有陈述的异常信息就是不存在的,只有明确已知的才是该实体的异常属性。这种处理方法显然反映了人类对异常属性的处理。

比如,在人们谈论鸟的时候如果没有特别说明这只鸟不会飞,那么大家都会认为这只鸟会飞,给它做笼子时就必须考虑怎样不让它飞走。只有在特别说明这只鸟的异常情况时,如这是一只死鸟,大家才会想到它是不会飞的,做笼子的时候就不必考虑会不会飞走的问题。因此,限制理论和限制推理为解决知识系统中的常识知识表示和推理问题奠定了基础。

由于限制推理问题必须转化成一阶逻辑中的证明问题,才能得到解决,故推理效率很低,这就限制了它在知识系统中的应用。在目前情况下,限制推理仅适用于低层次的,面向小型常识知识库的推理,而对大型知识系统的描述和处理却难以实现。当然,对可划分的大型知识系统,借助于任务划分方法,在多处理单元的支撑下,采用分布式问题求解方法,可以部分解决推理的效率问题。

参考文献

- [1] McCarthy, J., Circumscription—A form of non-monotonic reasoning, *Artificial Intelligence* 13 (1985), 27—39.
- [2] Lifschitz, V., Computing Circumscription, *Proceedings IJCAI-85, Los Angeles, CA (1985)*, 121—127.
- [3] Gelfond, M., Przymusinska, H. and Przymusinski, T., The Extended Closed World Assumption and Its Relationship to Parallel Circumscription, *Proceedings ACM SIGACT-SIGMOD Symposium on Principles of Database Systems, Cambridge, MA (1986)*, 133—139.
- [4] Przymusinski, T., An Algorithm to Compute Circumscription, *Artificial Intelligence* 38 (1989), 49—73.