第 6 卷第 5 期
1995 年 5 月

软 件 学 报
JOURNAL OF SOFTWARE

Vol. 6, No. 5
May 1995

带否定的 DATALOG 的语义的不动点特性*

周傲英　　施伯乐

(复旦大学计算机系,上海 200433)

摘要　带否定子目标的 DATALOG(称为 DATALOG~)是 DATALOG 的一种扩充.本文研究 DATALOG~语义的良基模型的不动点特性.首先,重新考察了稳定性变换,定义了新算子及其不动点.在此基础上,定义了振荡不动点模型论语义.然后,本文证明了振荡不动点模型和良基模型是等同的,说明前者可以看作是后者的构造性定义.

关键词　否定,DATALOG,模型论语义.

近年来,知识库语言方面的许多研究都是关于 DATALOG 的功能扩充的.目的在于扩充 DATALOG,增强其表达能力,例如,支持某种形式的非单调推理[1].本文研究 DATALOG 的一种扩充,记为 DATALOG~.DATALOG~中允许规则体中含有否定子目标.

对于 DATALOG~,如何定义其描述性语义是主要困难之一.这方面已有的工作主要是分 2 个方向开展的[2].一是"程序完整化"方法;另一是"正则模型"方法.第 1 种方法是 Clark 在 70 年代末提出的[3].这种方法的一个弱点是,在许多简单情况下,按这种方法给出的含义也会很不"自然",或者说不符合人们的直观理解[2].第 2 种方法是一个模型论方法,按照这种方法,程序中的规则都看作全称量化的蕴涵式,其语义是由某个正则模型确定的.通常,正则模型是从一组极小模型中选出的.人们提出了处理 DATALOG~程序语义的各种方案[4-7].其中的共同问题是:程序可能没有唯一的正则模型,并且,即使正则模型是存在的,也不清楚是否它能反映程序的主观含义.

人们研究了一些具有唯一正则模型的程序类.对于分层程序,分层模型语义是唯一的,可选来用于刻划程序的含义[4,5].对于局部分层程序,完美模型和程序的"自然"的主观含义是吻合的[7].然而,并非所有程序都是分层或局部分层的.有一些程序,尽管它们不是局部分层的,却表达了明确的意义,即存在自然的、无歧义的语义.这一事实激发了对更一般的程序及其语义的进一步研究.

稳定模型[6]和良基模型[2]语义是关于逻辑程序(无论局部分层与否)语义定义的两种提议.一个程序可能没有、或有一个、或有多个稳定模型.文献[6]中已经说明,当程序具有唯一的稳定模型时,这一模型可以考虑为程序的"自然的"主观含义.一般情况下,良基模型是一

* 本文 1993-06-22 收到,1993-09-14 定稿

本研究得到国家自然科学基金和国家教委博士点基金的资助.作者周傲英,1965 年生,副教授,主要研究领域为演绎数据库,面向对象数据库.施伯乐,1936 年生,教授,主要研究领域为数据库,知识库,面向对象数据库.

本文通讯联系人:周傲英,上海 200433,复旦大学计算机系

个部分模型.良基模型语义是局部分层程序的完美模型语义在一般程序中的推广.业已证明,完美模型语义的许多较为自然的性质,良基模型语义也具备.许多研究都表明了良基模型语义的合理性,提供了足够的证据说明良基语义和人们意识中的常识及直觉相一致[2].因此,本文也选择良基模型来表示 DATALOG~程序的主观含义.

本文着重研究 DATALOG~程序的良基模型语义的不动点特性.首先,我们重新考察了文献[6]中定义的稳定性变换.在此基础上,我们定义了一个新的算子.对于任一程序,反复使用这个算子结果会到达一个不动点,我们称之为振荡不动点.对于一般程序而言,振荡不动点模型和良基模型是等同的.因此,振荡不动点模型可以看作是良基模型的另一种构造性定义.

1 振荡不动点

这一节中,我们首先描述后续讨论中所必需的一些术语和记法.然后,重新考察稳定性变换,给出了说明振荡不动点的演示例子及振荡不动点的定义.

一个 DATALOG~程序是具有以下形式的 DATALOG~规则的一个有穷集合:

$$H:-B_1,B_2,\cdots,B_n.$$

其中 H 是一个原子,$B_1,\cdots,B_n$ 是(正的或否定的)文字.我们把 H 称为规则的首部;$B_1,\cdots,B_n$ 称为规则体,每个 $B_i(1\leqslant i\leqslant n)$ 称为子目标.规则体中的逗号代表合取,“:—”可以理解为蕴涵式中的“如果”.

给定一个程序 P,P 的 Herbrand 基,记为 H_P,是按以下方式所能构造的所有的基原子(即不含变量的原子)的集合.原子的谓词是 P 中出现的谓词符号;原子的参量是 P 中出现的常量.在无函词的情况下,P 中出现的所有常量组成的集合就是程序 P 的 Herbrand 空间,记为 U_P.程序 P 的 Herbrand 实例化结果 P_H 是对 P 中的规则按任何可能的方式用 U_P 中的常量替换变量得到的所有规则的集合.P_H 中的任一规则都称为实例化规则.

对于一个给定的程序 P,设 X 是 H_P 的一个子集,则我们用 $\sim.X$ 表示集合 $\{\sim A\mid A\in X\}$.程序 P 的一个部分解释 I 是 H_P 的一个划分,I 把 H_P 划分成3部分,分别记为 I^+,I^-,$I^?$.I^+ 代表解释 I 中为真的基原子的集合;I^- 代表在解释中为假的基原子的集合,换句话说,I^- 的基原子的补为真;$I^?$ 表示 H_P 中的在 I 中“未定义”的基原子的集合,即 $I^?$ 的基原子的真假值是未知的.对于一个解释 I 而言,如果 $I^?$ 为空,或者说,$H_P=I^+\cup I^-$,那么,I 就是一个完全解释.通常习惯把一个完全解释表示成集合 I^+(并且,上标往往省略),I^+ 中的基原子都是真的,H_P 中有而 I^+ 中没有的基原子都隐含假设属于 I^-,即都是假的.然而,为了表示一个部分解释,I^+ 和 I^- 都需要显式给出,$I^?$ 则可以看作是由 H_P 中不出现在 I^+ 和 I^- 中的原子构成的.

定义 1.1. 对一个程序 P 而言,P 的一个完全模型 M 是 P 的一个完全解释,它使得如果把属于 $M\cup\sim.(H_P-M)$ 的所有基文字(可能是正的、或是负的)都理解为真,而其它任何基文字都理解为假,则 P_H 中的每一个实例化规则都为成立. □

定义 1.2. 对程序 P 而言,P 的一个部分模型 $M=\langle M^+,M^-\rangle$ 是 P 的一个部分解释,它使得对于 M^- 中的每一个基原子“a”,P_H 中任何以“a”为规则首部的实例化规则的体部都至

少包括一个子目标"b",并且"b"是在 M 中是已知为假的,即 $b\in(\sim.M^{+}\cup M^{-})$. □

1.1 稳定性变换

稳定性变换是从一个完全解释到另一个的分 3 个阶段完成的变换,这一变换最初是在文献[6]中提出的,用于测试一个极小完全模型的稳定性.以下是其定义.

定义 1.3. 给定一个程序 P,给定其 Herbrand 实例化结果 P_H,以及 P 的一个完全模型 M.则 M 的稳定性变换,记为 $S_P(M)$,可以定义如下:

$S_P(M)$分以下 3 个步骤完成:对 P_H 中的所有实例化规则

①如果规则中有一否定子目标$\sim A$,而 $A\in M^{+}$,则删除这一规则.

②在第①步完成后所剩下的规则中,去掉规则体中的所有否定子目标(这些否定子目标必定是$\sim A$ 且 $A\in M^{-}$).

③对前 2 步结束后产生的结果程序(这时一定是一个 *Horn* 子句程序)使用标准的直接结果算子构造其最小(完全)模型.

以上最后形成的模型就是 $S_P(M)$. □

给定一个程序 P,假设 M 是 P 的一个最小(完全)模型.为了进行 M 的稳定性测试,用给定模型 M 与稳定性变换产生的结果模型 $S_P(M)$ 相比较.如果 $M=S_P(M)$,则称 M 是一个稳定模型.

首先,回忆一下变换 $T_P(I)$.T_P 称为 *Horn* 子句程序的直接结果算子.T_P 的输入是一个解释 I,其输出是如下的一个基原子集合:$P\in T_P(I)$ 当且仅当 P_H 中存在一个实例化规则,其规则首部是 P,而规则体部的所有子目标在解释 I 中都为真.这样一来,S_P 可以看作是 T_P 在更一般的情况下的翻版.而且,从上述定义可以看出,S_P 可以按照 T_P 的思想来理解.对于一个给定程序 P,如果 P 的规则体内有否定子目标,那么,每一个否定子目标$\sim p(X)$都可以被当作是一个带有特殊谓词"$\sim p$"的新的正子目标,这里"$\sim$"不再是一个否定操作符,"$\sim p$"放在一起代表一个谓词.从这个观点出发来理解,P 是一个 *Horn* 子句程序,给定一个完全解释 I,假定它是由 H_P 的一个子集表示,其中每一个原子都为真.则(H_P-I)中的原子都为假.换句话说,$\sim.(H_P-I)$中的每一个(否定)基文字都为真.因此,$\sim.(H_P-I)$可被看作是形如"$\sim P$"的谓词的事实的集合.如果把 P' 定义为 $P_H\cup\sim.(H_P-I)$,则 $S_P(I)=T_{P'}^{\infty}(\Phi)$.不难看出以下的紧密联系.

命题 1.1. 设 P 是一个给定的 DATALOG~程序,I 是 P 的一个完全解释.假定 P_I 是对 P 施用相对于 I 的 S_P 变换时前 2 步的结果,假设 P' 是根据程序 P 和 I 按上述方法构造出来的程序,则

$$S_P(I)=T_{P_I}^{\infty}(\Phi)=T_{P'}^{\infty}(\Phi).$$ □

$S_P(I)=T_{P'}^{\infty}(\Phi)$是本小节得出的主要结论.这个等式蕴涵的基本思想是:$S_P(I)$给出了按以下方式可导出的正事实的集合.如果把 P 中的否定子目标当作正的子目标,其谓词带有前缀"$\sim$",这些新谓词对应的事实的集合是$\sim.(H_P-I)$,则 $S_P(I)$是利用以上意义的 P 及相关谓词的事实可推导出来的正事实的集合.

1.2 振荡不动点

至此,我们可以讨论 S_P 的某些性质.首先,我们给出一个演示例子.利用这个例子,可以直观地例示 S_P 的某些性质,也说明取名"振荡不动点"的内在意图.

例 1.1. 考虑下面的程序 P，其中的规则都是实例化规则，规则中的原子缩写为单个字母. 实际上，可能会有几个字母表示具有相同谓词符号，而仅是参量值不同的基文字.

$b:-\sim a \qquad c:-\sim b \qquad c:-a,\sim p \qquad p:-\sim q \qquad q:-b,\sim p.$

这里，P 的 Herbrand 基 $H_P=\{a,b,c,p,q\}$.

我们从 $I_0=\Phi$ 开始. 假设 $I_1,I_2,\cdots$，是由 S_P 产生的完全解释的一个序列，且有 $I_{i+1}=S_P(I_i)$. 在上一节，我们给出了 S_P 的另一个利用算子 $T_{P'}$的定义. 当使用 T_P 时，"$\sim q$"被当作是一个单独的事实，而不是 q 的否定.

首先，$I_0=\Phi$，因此，按照 P' 的构造，可知

$$P_0'=P\cup\sim.(H-I_0)=P\cup\{\sim a,\sim b,\sim c,\sim p,\sim q\}.$$

则 $I_1=S_P(I_0)=T^{\infty}_{P_0'}=\{b,c,p,q\}$.

为了计算 I_2，P_1'可按以下方式构造

$$P_1'=P\cup\sim.(H-I_1)=P\cup\{\sim a\}.$$

则 $$I_2=S_P(I_1)=T^{\infty}_{P_1'}(\Phi)=\{b\}.$$

类似地 $$P_2'=P\cup\sim.(H-I_2)=P\cup\{\sim a,\sim c,\sim p,\sim q\}.$$

因而 $$I_3=S_P(I_2)=T^{\infty}_{P_2'}(\Phi)=\{b,p,q\}.$$

继续下去 $P_3'=P\cup\sim.(H-I_3)=P\cup\{\sim a,\sim c\}$，$I_4=S_P(I_3)=T^{\infty}_{P_3'}(\Phi)=\{b\}$.

可以看出 $$I_4=I_2,$$

因此 $$I_5=S_P(I_4)=S_P(I_2)=I_3=\{b,p,q\}.$$

这时，我们可得 $I_4=I_2$，$I_5=I_3$. 显然，没有必要再扩充这个序列，因为在这个序列的后继部分，必有 $I_6=I_4$，$I_7=I_5$，等等. 这个序列可按以下的方式分 2 列列出：

$I_0:\Phi$	$I_1:\{b,c,p,q\}$
$I_2:\{b\}$,	$I_3:\{b,p,q\}$
$I_4:\{b\}$,	$I_5:\{b,p,q\}$
……	……

对于这个序列，我们的第一印象是其中的解释是振荡的，如下所示：

$$\Phi\to\{b,c,p,q\}\to\{b\}\to\{b,p,q\}\to\{b\}\to\{b,p,q\}\to\cdots$$

沿着这个序列，振荡的幅度变得越来越小，最终变成一个等幅振荡，即从$\{b\}$到$\{b,p,q\}$和从$\{b,p,q\}$到$\{b\}$. 其次，序列中带有偶数下标的那些元素，如 $I_0,I_2,I_4,\cdots$，是单调递增的，并从下往上收敛到一个不动点. 与此相反，序列中带有奇数下标的元素，即 $I_1,I_3,I_5,\cdots$，是单调递减的，并且从上往下收敛到一个不动点. 最后，正如我们在下节将证明的一样，对于任一给定的程序 P，这 2 个不动点在 P 的良基模型 W 有着紧密的联系. 假设 W 表示为$\langle W^+,W^-,W^?\rangle$，则这 2 个不动点分别和 W^+和 $W^+\cup W^?$ 相同. 本例中，$W^+=\{b\}$，$W^+\cup W^?=\{b,p,q\}$. 因为 W^+、W^-和 $W^?$ 是两两不相交的，因此，$W^+=\{b\}$，$W^?=\{p,q\}$，$W^-=H-(W^+\cup W^?)=\{a,c\}$. 确实的，$\langle\{b\},\{a,c\},\{p,q\}\rangle$正是 P 的良基模型. □

正如上例中非形式化地说明的，带有偶数下标的元素组成的序列收敛到良构模型的真原子集合. 为了达到这个收敛点，我们定义如下的变换.

定义 1.4. 变换 O_P 定义为：

$$O_P(I)=S_P(S_P(I)).$$

这个变换是单调的(容易根据 S_P 的定义证明)，O_P 的最小不动点

$$O^+=O_P^\infty(\Phi).$$

称为振荡不动点，也就是期望到达的收敛点. □

定义 1.5. 设 O^+ 是以上定义的不动点，$O^? = S_P(O^+)-O^+$，$O^-=H_P-S_P(O^+)$，则 $\langle O^+, O^-, O^?\rangle$ 是振荡不动点(部分)模型. 如果 $O^? = \Phi$，则称之为振荡不动点完全模型. □

2　与良基模型的关系

首先，我们重新考察与良基模型相关的一些概念，如非基集(Unfounded Set)，U_P 变换和 W_P 变换等. 接着，我们给出振荡不动点模型和良基模型之间的关系. 贯穿本节，我们都假设给定的程序是 P，其 Herbrand 基是 H_P，其良基模型是 $W=\langle W^+, W^-, W^?\rangle$.

2.1　良基模型(well-founded model)

定义 2.1. 对 P 的一个部分解释 $I=\langle I^+, I^-, I^?\rangle$ 而言，我们说 $A\subseteq H_P$ 是 P 关于 I 的一个非基集，如果每一个原子 $p\in A$ 都满足以下条件：

对于 P 中规则首部为 p 的每一个实例化规则，下列条件中至少有一条成立：

①规则体内某个(正的或否定的)子目标 q 在 I 中为假，即 $q\in(\sim.I^+\cup I^-)$，

②规则体中某个正的子目标出现在 A 中.

程序 P 关于 I 的所有非基集的并，记为 $U_P(I)$，也是一个非基集，称为 P 关于 I 的最大非基集. □

定义 2.2. 对于一个部分解释 $I=\langle I^+, I^-, I^?\rangle$，变换 T_P，U_P 和 W_P 分别定义如下：

- $p\in T_P(I)$ 当且仅当有 P 的某个实例化规则 r，满足下述条件：r 的规则首部是 p，r 的规则体中的每一个子目标在 I 中都为真(即属于 $(I^+\cup\sim.I^-)$).
- $U_P(I)$ 是 P 关于 I 的最大非基集
- $W_P(I)=T_P(I)\cup\sim.U_P(I)$. □

根据定义可以直接得知：T_P，U_P 和 W_P 都是单调变换.

定义 2.3. P 的良基模型定义如下：W_P 的最小不动点中每个正文字是 W^+ 中的一个元素，每个否定文字所对应的正文字是 W^- 中的一个元素，不动点中没出现的原子都是 $W^?$ 中的元素. □

2.2　振荡不动点与良基模型

在 1.2 节中，我们非形式化地断言 $O_P^\infty(\Phi)=W^+$，$S_P(O_P^\infty(\Phi))=W^+\cup W^?$. 这里给出证明，然后说明振荡不动点模型 $O=\langle O^+, O^-, O^?\rangle$ 和良基模型 $W=\langle W^+, W^-, W^?\rangle$ 是相同的.

引理 2.1. $S_P(W^+)=H_P-W^-$.

证明：首先回忆一下，S_P 是从一个到另一个完全解释的变换，并且完全解释是用其真的基文字表示的. 这里，$S_P(W^+)$ 代表一个完全解释 W^+ 上的稳定性变换. 这个完全解释的真文字集合为 W^+，H_P 中其它的文字都为假.

正如 1.1 节中所述，S_P 起初在文献[6]中是定义为经过前 2 步约减后所剩下的程序 P_I 的最小模型. 为简单起见，我们在证明中使用原始定义，结果产生的最小模型 $S_P(W^+)$ 是 H_P 的一个子集.

本节中，$W=\langle W^+,W^-,W^?\rangle$是$P$的良基模型，即$W$是变换$W_P$的不动点．因此，$W^+=T_P(W)$，$W^-=U_P(W)$．为了证明$S_P(W^+)=H-W^-$，只要证明，对每一个文字$p$，$p\notin S_P(W^+)\Leftrightarrow p\in U_P(W)$就足够了．

首先，我们证明$p\in U_P(W)\Rightarrow p\notin S_P(W^+)$．根据变换$U_P$的定义，对给定程序$P$的每一实例化规则$r$，其规则首部为$p$，以下情形之一成立：

①规则体中某个子目标q在W中为假，即$q\in\sim.(W^+\cup\sim.W\ \ )=\sim.W^+\cup W^-$，或者

②某个正子目标属于$U_P(W)=W^-$．

显而易见，情形②被①包含，因而是不必要的．因此，这里p的非基性的条件可以描述为：对于每一个规则首部为p的实例化规则，规则体中存在某个子目标q，使得$q\in\sim.W^+$或$q\in W^-$．

再来看$S_P(W^+)$，在这个稳定性变换中，在约减的第1步，那些含有属于$\sim.W^+$的否定子目标的实例化规则被去掉．然后在第2步，规则中所有的否定子目标（它们在$\sim.(H_P-W^+)$中为真）被删除．在上一段中已经说明，关于W^-的每一个元素p，每一个规则首部为p的实例化规则的体中至少有一个否定子目标$q\in\sim.W^+$，或至少有一个正子目标$q\in W^+$．在P_H中，p的规则，如果其体内包含有属于$\sim.W^+$的子目标，就会在稳定性变换的第1步被去掉，因而就对$S_P(W^+)$中p的导出没有贡献．关于$p\in W^-$的余下的规则中必定在每一个规则体中含有一个子目标$q\in W^-$．因此，在P的关于W^+约减结果程序P_I中，每一个规则首部在W^-中的规则一定至少有一个子目标是在W^-中的．从文字间依赖的观点来看，W^-的元素间的依赖是有回路的．其中的基原子没有一个是可以从$T_{PI}^{\infty}(\Phi)=S_P(W^+)$中导出的，也就是说，$P_I$中余下的规则首部为$p$的规则对$p$的导出没有任何贡献．

从以上讨论，我们可以得到$p\in U_P(W)\Rightarrow p\notin S_P(W^+)$．

现在，我们来证明$p\notin S_P(W^+)\Rightarrow p\in U_P(W)$．

这等价于证明$p\in(H_P-S_P(W^+))\Rightarrow p\in U_P(W)$．$U_P(W)$是$P$关于$W$的最大非基集，因此，只要证明$(H_P-S_P(W^+))$是$P_H$关于$W$的一个非基集就足够了．假定某个$p\in(H_P-S_P(W^+))$不满足非基集的条件，则$P_H$中必有一个规则

$$p:-a_1,\cdots,a_m,\sim b_1,\cdots,\sim b_n,$$

使得以下事实成立：

①没有a_i在W中为假，即$a_i\notin W^-$，

②没有b_j在W中为真，即$b_j\notin W^+$，

③没有a_i在$(H_P-S_P(W^+))$中为真，即$a_i\notin(H_P-S_P(W^+))$，所以，$a_i\in S_P(W^+)$．

因为W是P的一个良基模型，由以上第②个事实可知，形如$\sim b_i$的文字是不假的．因此，在P关于用W^+所表示的完全解释的约减结果程序P_I中，有一条规则

$$p:-a_1,\cdots,a_k.$$

从事实③，我们可知对$i=1,\cdots,k$，$a_i\in S_P(W^+)$．因此，我们有$p\in S_P(W^+)$．这样一来，就发生了矛盾．所以，我们可以得出结论$(H_P-S_P(W^+))$是一个非基集．　□

引理2.2. W^+是O_P的一个不动点．

证明：回忆变换O_P的定义，对一个完全解释I，$O_P(I)=S_P(S_P(I))$．本引理的目标是证

明 $O_P(W^+)=W^+$，即 $S_P(S_P(I))=W^+$. 根据引理 2.1，$S_P(W^+)=H_P-W^-=W^+\cup W^?$. 这样，只要证明 $S_P(W^+\cup W^?)=W^+$，即 $S_P(W^+\cup W^?)=T_P(W)$就足够了.

首先，我们证明 $T_P(W)\subseteq S_P(W^+\cup W^?)$. 根据定义 2.2 中给出的变换 T_P 的定义可知，$p\in T_P(W)$当且仅当 P_H 中有某个实例化规则 r，其规则首部是 p，规则体内每一个子目标都为真，即属于$(W^+\cup W^-)$. 当 P_H 相对于$(W^+\cup W^?)$约减时，关于 p 的这样的规则在第 1 个约减阶段后被保留下来. 在第 2 个阶段，仅有其中的否定子目标被删除. 结果产生的程序可以直接导出 p. 因此，有 $p\in S_P(W^+\cup W^?)$.

现在，我们来证明 $S_P(W^+\cup W^?)\subseteq T_P(W)$，即对于每个 $p\in S_P(W^+\cup W^?)$，有 $p\in T_P(W)$. 我们在 $S_P(W^+\cup W^?)$的构造的步骤上施行归纳法来予以证明. 对第 0 步，要证的结论平凡成立，因为 $S_P(W^+\cup W^?)$为空. 假定，对 $k>0$，p 是在 $S_P(W^+\cup W^?)$构造的第 k 步导出的，则 P_I 中存在一条规则

$$p:-a_1,\cdots,a_m.$$

使得 $a_i(1\leqslant i\leqslant m)$是在小于 k 的步骤中已经导出的并且属于 W^+. 这条规则对应于 P_H 中具有以下形式的某条规则：

$$p:-a_1,\cdots,a_m,\sim b_1,\cdots,\sim b_n.$$

并且这条规则在约减的第 1 阶段保留下来. 这里，$b_j\in H_P-(W^+\cup W^?)=W^-$.

根据归纳假设，$p\in T_P(W)$. 因此，对每个 $p\in S_P(W^+\cup W^?)$，必有 $p\in T_P(W)$. □

引理 2.3. $O_P^\infty(\Phi)\subseteq W^+$.

证明：O_P 是单调递增算子. $O_P^\infty(\Phi)$是最小不动点，根据引理 2.2，W^+是 O_P 的一个不动点. 所以，$O_P^\infty(\Phi)\subseteq W^+$. □

引理 2.4. $O_P^\infty(\Phi)\supseteq W^+$.

证明：用反证法. 假定不成立. 设文字集 $O^+=O_P^\infty(\Phi)$. 进一步，设文字集 $O^-=(H_P-S_P(O^+))$. 因为我们是在不动点上，$S_P(H_P-O^-)=O^+$. 设 $Q=W_P(O)$是由 Q^+和 Q^-组成，我们必定有 $Q^+=O^+$，否则就有 $S_P(O^+)$比(H_P-O^-)要小. 因为 W 是 W_P 的最小不动点，W_P 又是单调的，由此可推 $Q^-\supseteq O^-$. 但是，对 $p\in(Q^--O^-)$而言，因为 p 是在关于 O 的非基集之中的，这隐含着对于 P_H 中任何给定的 p 的规则，要么它有一个正子目标 $q\in Q^-$，要么它有一个否定子目标$\sim r$，使得 $r\in O^+$. 由此可推出 $p\notin S_P(O^+)$，因此，$p\in(H_P-S_P(O^+))=O^-$. 这是矛盾的. □

定理 2.1. 对于一般程序，振荡不动点模型和良基模型是等同的.

证明：根据引理 2.3 和 2.4 可知，$O^+=W^+$. 根据引理 2.1 和定义 1.5 可得，$O^-=W^-$. 所以我们有，$\langle O^+,O^-,O^?\rangle$等同于$\langle W^+,W^-,W^?\rangle$. □

3　结论和未来的工作

①本文重新考察了稳定性变换. 定义了一个新的算子 O_P 及其不动点. 基于这一点，定义了一个不动点语义，称为振荡不动点模型语义.

②本文证明，对任何 DATALOG~ 程序，振荡不动点模型和良基模型是一样的. 所以，振荡不动点模型可以看作是良基模型的构造性定义.

值得进一步研究的问题有：

①DATALOG~程序自底向上计值和优化.对 DATALOG 程序，人们已经提出许多有效的方法.将这些思想或技术推广应用到 DATALOG~程序语义的计值优化是我们下一阶段的工作.

②研究新的描述性语义及其有效的计算.从本节前一部分的讨论可知，良基模型语义会带来语义与主观愿望不相符合，以及有用信息丢失等问题.我们拟从实用的角度研究一般程序上的语法及语义限制和合适的不动点语义.

参考文献

1 Przymuskinski T C. Every logic program has a natural stratification and an iterated fixed point model. Proc. of 8th ACM Symp. on PODS, 1989. 11—21.

2 Gelder A V, Ross K A, Schlipf J S. The well—founded semantics for general logic programs. J. of ACM, 1991, **38(3)**: 620—650.

3 Clark K L. Negation as failure. In: Gallaier H, Minker J eds. Logic and Database, Plenum Press, 1978. 293—322.

4 Gelder A V. Negation as failure using tight derivation for general logic programs. In: Minker J ed. Foundations of Deductive Databases and Logic Programming, Morgan Kaufmann Publishers, 1988. 149—176.

5 Przymusinski T C. On declarative semantics of deductive databases and logic programs. In: Minker J ed. Foundations of Deductive Databases and Logic Programming, Morgan Kaufmann Publishers, 1988. 193—216.

6 Gelfond M, Lifschitz V. The stable model semantics for logic program. In: Proc. of Intl. Conf. on Logic Programming, 1988. 1070—1080.

7 Apt K R, Blair H A, Walker A. Towards a theory of declarative knowledge. In: Minker J ed. Foundations of Deductive Databases and Logic Programming, Morgan Kaufmann Publishers, 1988. 89—148.

THE FIXPOINT CHARACTERISTIC OF SEMANTICS OF DATALOG WITH NEGATION

Zhou Aoying Shi Baile

(*Department of Computer Science, Fudan University, Shanghai* 200433)

Abstract Datalog with negation (called Datalog~) is an extension of Datalog. The fixpoint characteristic of well—founded model semantics of Datalog~ is discussed in this paper. At first, the stability transformation is revisited, and a new operator O_P and its fixpoint are defined. Based on this, the oscillating fixpoint model semantics is defined. Then, it is shown that for any datalog~ program the oscillating fixpoint model is identical to its well—founded model, and so, the former can be viewed as a constructive definition of the latter.

Key words Negation, DATALOG, model—theoretical semantics.