一种求解 tableau 等式合一问题的算法*)

刘 全[1,2] 孙吉贵[2] 窦全胜[2]

(苏州大学计算机科学与技术学院 苏州 215006)[1] (吉林大学计算机科学与技术学院 长春 130012)[2]

摘 要 在增添扩展规则的 tableau 方法的基础上提出了一种新的含等词 tableau 方法——等式合一方法,并证明了它的可靠性和完备性。在该方法中,将 tableau 分成两个阶段,等词单独处理,通过提取等式合一问题并求解解替换封闭 tableau,进一步限制了 tableau 的搜索空间,提高了 tableau 的推理效率。同时,为了研究等式合一方法的有效性,在解替换求解方面,提出了提取不等式析取,并在启发式的帮助下计算等价类的方法。通过实例分析,结果表明,等式合一方法优于其它方法。

关键词 等式合一,不等式析取,等价类

An Algorithm to Solve Tableau with Equality Unification Question

LIU Quan[1,2] SUN Ji-Gui[2] DOU Quan-Sheng[2]

(Institute of Computer Science and Technology, Soochow University, Suzhou 215006)[1]

(Institute of Computer Science and Technology, Jilin University, Changchun 130012)[2]

Abstract A new method of tableau with equality based on the additional tableau expansion is proposed and its sound and complete is proved. We adopt the idea of separating the tableau expansion into two stages in the method. The equality handle is independent. It restricts the tableau search space and raises the inference efficiency by extracting the equality unification problems and solving the substitution to close tableau. At the same time, to verify the validity of the algorithm, we give a method based on the transformation into disjunctions of inequalities and the calculation of equivalence classes with the help of heuristics. Though the instance analysis and implementation of the EU-Tableau system, it shows that equality unification is superior to other methods.

Keywords Equality unification, Disjunctions of inequality, Equivalence classes

1 引言

自动推理研究的主要目标之一就是有效地处理含等词的逻辑。由于等词的存在,导致证明的搜索空间膨胀,使简单的定理证明变得非常复杂,甚至得不到证明。因此,多年来人们在含等词的推理方面,一直在寻求限制证明搜索空间的有效方法。在归结方面,最重要的方法是调节和 RUE 归结[1,2]。尽管这两种方法还不能有效避免冗余子句的产生,但一直被使用和实现。从 20 世纪 60 年代,tableau 方法产生以后,含等词的 tableau 也越来越受到重视,如 Jeffrey、Reeves[3] 以及 Fitting[4] 方法,这些方法都是增加扩展规则的 tableau 方法,很大程度上依赖于 tableau 的类型。冗余子句数目随着谓词元数的增加呈幂指数增长,不利于计算机的实现。本文在增添扩展规则的 tableau 方法的基础上提出一种新的含等词 tableau 算法——等式合一方法。在本方法中,将 tableau 分成两个阶段,等词单独处理,通过提取等式合一问题并求解解替换封闭 tableau,本法进一步限制了 tableau 的搜索空间,提高了 tableau 的推理效率[5]。

2 等式合一问题

定义 1 等式合一问题 $\langle E,s,t\rangle$ 包含形如 $(\forall x_1)\cdots(\forall x_n)(l\approx r)$ 的等式的有穷集合 E、项 s 和 t。一个替换 σ 为等式合一问题的解当且仅当 $E^{\sigma}\models(s^{\sigma}\approx t^{\sigma})$ 成立。

根据定义 1,合一问题 $\langle\{f(x)\approx x\},f(a),a\rangle$ 的解为 $\{a/x\}$;合一问题 $\langle\{(\forall x)f(x)\approx x\},g(f(a),f(b)),g(a,b)\rangle$ 的解为任意替换;合一问题 $\langle\{f(x)\approx x\},g(f(a),f(b)),g(a,b)\rangle$ 无解。

定义 2[6] 含有相同谓词和互补符号的原子公式 $p(s_1,\cdots,s_n)$ 和 $\neg p(t_1,\cdots,t_n)$ 称为潜在封闭原子对。

实际上,Reeves 方法就是对潜在封闭原子对的处理方法,这里我们将进一步引申为从 tableau 中提取等式合一问题。

定义 3 令 B 为一个 tableau 分枝,等式合一问题集合包含集合序对 $\langle E(B),P(B)\rangle$,其中,

(1) 集合 $E(B)$ 定义为:$\{s\approx t;s\approx t\in B\}\subset E(B)$;

(2) 对于 B 上的潜在封闭原子对 $p(s_1,\cdots,s_n)$ 和 $\neg p(t_1,\cdots,t_n)$(这里 $p\neq\approx$),存在某一替换,使得 $\{s_1^{\sigma}\neq t_1^{\sigma}\vee\cdots\vee s_n^{\sigma}\neq t_n^{\sigma}\}\subset P(B)$;

(3) 对于 B 上的每个不等式 $\neg(s\approx t)$,$\{(s^{\sigma},t^{\sigma})\}\subset P(B)$。

定义 4 令 B 为一个 tableau 分枝,对于 B 上的项 $[t]_B$,存在一个基替换 σ,通过集合 $E(B)$ 中的等式变换得到 $[s]_{B^{\sigma}}$,那么称 $[s]_{B^{\sigma}}$ 为项 $[t]_B$ 的等价类,记为 $[t]_B\sim[s]_{B^{\sigma}}$。

3 等式合一问题的实现算法

寻找等价类的方法可以看出,由于等式的对称性,可能导致等价类搜索树的无限增长,使机器的实现效率降低。

*)本课题得到国家自然科学基金(60073039,60273080,60473003)资助。**刘 全** 副教授,博士,研究方向为定理机器证明与自动推理,地理信息系统。**孙吉贵** 教授,博士导师,研究方向为人工智能、自动推理。

利用等式合一方法处理等词，对于基 tableau，可以直接通过寻找等价类，在 $P(B)$ 中计算出不可满足的公式。而对于自由变量 tableau，不同的替换可以导出不同的等价类，因此计算出一个可以反驳 $P(B)$ 中的所有不等式的替换是一个至关重要的问题。

计算等价类产生的集合序列可以用集合 $\langle t\rangle$ 表示，具体定义如下：

定义 5 对于项 t 和分枝 B，存在 $\langle t\rangle_B::\{s^{\sigma}$：存在替换 σ 使得 $s\in[t^{\sigma}]_{\sigma}\}$。

$\langle t\rangle_B$ 为在分枝 B 中，通过不同的替换得到的项 t 的所有等价类的集合。$\langle t\rangle_B$ 的元素 s_σ 为通过使用 $E(B)$ 中的不等式，从项 t 中得到的项，以替换为下角标。

如果通过替换 $\{a/x_1\}$ 使项 a 产生一个等价类 $g(f(a))$，那么 $g(f(a))_{\{a/x_1\}}$ 包含在 $\langle a\rangle_B$ 中。

通常，集合 $\langle t\rangle_B$ 为一无穷集合，但是可以通过某一算法计算出一个近似序列 $(\langle t\rangle_B^n)_{n\geqslant 0}$ 来表示 $\langle t\rangle_B$。

定义 6 s、s' 为项，σ、σ' 为替换，如果 s 和 s' 在 MGU τ 下是可合一的，使得 $s^{\tau}=s'$，σ 和 τ 都是比 σ' 更一般的合一，s_σ 包含 $s'_{\sigma'}$。记为：$s_\sigma\gg s'_{\sigma'}$。

表 1 几种符合包含关系的实例

s_σ	$s'_{\sigma'}$	τ
$f(x)_{\{a/y\}}$	$f(a)_{\{a/x,a/y\}}$	$\{a/x\}$
$a_{\{y/x\}}$	$a_{\{b/x,b/y\}}$	$\{b/y\}$
x_{id}	$a_{\{a/x\}}$	$\{a/x\}$

另外，$a_{\{f(y)/x\}}$ 不包含 $a_{\{f(b)/x\}}$。

定理 1 如果 $s_\sigma\gg s'_{\sigma'}$ 且 $s'_{\sigma'}\gg s''_{\sigma''}$，那么 $s_\sigma\gg s''_{\sigma''}$。

证明：如果 $s_\sigma\gg s'_{\sigma'}$ 且 $s'_{\sigma'}\gg s''_{\sigma''}$，那么根据定义 6，存在替换 τ 和 τ'，使得

$s\tau=s'$ 且 $s'\tau'=s''$

并且 $\sigma\phi\sigma'\phi\sigma''$，$\tau\phi\sigma'$ 且 $\tau'\phi\sigma''$。

设 $\tau''=(\tau'\cdot\tau)$，有 $s^{\tau''}=(s^{\tau})^{\tau'}=s'^{\tau'}=s''$。

并且 $\sigma\phi\sigma'\phi\sigma''$，$\tau\phi\sigma'$ 且 $\tau''\phi\sigma''$。

因此存在替换 μ 和 ν，使得

$\sigma''=\mu\cdot\tau=\nu\cdot\tau'$

并且 $\mu\cdot\tau''=\nu\cdot\tau'\cdot\tau=\mu\cdot\tau\cdot\tau=\mu\cdot\tau=\sigma''$ 成立。

根据定义 6，有，$s_\sigma\gg s''_{\sigma''}$。

证毕。

算法 1 （计算集合 $\langle t\rangle$ 的等价序列的近似算法）

输入：项 t、等式集合 $E(B)$、n 的最高次数
输出：项 t 的等价序列集合 $\langle t\rangle_B^n$

```
begin
    ⟨t⟩B = {t_id}
    n=1
    while n<=N do
    begin
        将⟨t⟩B^(n-1) 中不被另一元素包含的所有元素加入到 Θn 中
        sσ= H(⟨t⟩B)
        while flag do
        begin
            sσ^g 加入到集合 Θn 中
            σ 是 s 和等式 G∈E(B)一端的 MGU
            if σ 存在 then flag=true
            else flag=false
            n=n+1
        end
    end
end
```

例 1 表 2 给出了利用 $E(B)=\{g(f(x_1)\approx x_1,g(x_2)=f(x_2))$ 计算 $\langle a\rangle_B^n(n=0,1,2)$ 等价类的过程。

表 2 计算 $\langle a\rangle_B^n$ (n=0, 1, 2) 的等价类

n	$\langle a\rangle_B^n$	$H(\langle a\rangle_B^n)$	Θ_n
0	a_{id}	a_{id}	a_{id} $g(f(a))_{\{a/x1\}}$
1	a_{id} $g(f(a))_{\{a/x1\}}$	$g(f(a))_{\{a/x1\}}$	a_{id} $g(f(a))_{\{a/x1\}}$ $a_{\{a/x1\}}$ $g(f(g(f(a))))_{\{a/x1\}}$ $g(g(a))_{\{a/x1,a/x2\}}$ $f(f(a))_{\{a/x1,f(a)/x2\}}$
2	a_{id} $g(f(a))_{\{a/x1\}}$ $a_{\{a/x1\}}$ $g(f(g(f(a))))_{\{a/x1\}}$ $g(g(a))_{\{a/x1,a/x2\}}$ $f(f(a))_{\{a/x1,f(a)/x2\}}$		

定义 7(可满足启发式) 启发式 H 是可满足的，如果对于每个项 t 和元素 $s_\sigma\in\langle t\rangle_B^n(n\geqslant 0)$ 时，有 $H(\langle t\rangle_B^m)=s'_{\sigma'}(m\geqslant 0)$，并且有 $s'_{\sigma'}\gg s_\sigma$。

用来从启发式 $H(\langle t\rangle_B^m)$ 选择一个元素的启发式必须是可满足的。为了实现可满足的启发式，通常需要定义权重函数，然后选择权重最小的项作为可满足的项，这样的权重函数是任意的。下面给出一种选择可满足启发式的方法。

定义 8 $W(t)$ 是关于项 Term 的启发函数，那么(1)对于给定的 $\omega\geqslant 0$，存在集合 $\{t\in \text{Term}:W(t)\leqslant\omega\}$；(2) $W(t)$ 是关于项包含关系的递减函数，即如果 $t\gg t'$，那么 $W(t)\leqslant W(t')$。

定理 2 通过启发式函数 W 来选择(1) 权重最小的；(2) 没有被选择过的。项是可满足的项，指

证明：设任意 $s_\sigma\in\langle t\rangle_B^n$，对于所有的 $m\geqslant n$ 有 $s'_{\sigma'}\in\langle t\rangle_B^m$ 成立，并且 $s'_{\sigma'}\gg s_\sigma$。根据算法 1，元素 $\langle t\rangle_B^{n+k}$ 来自 $\langle t\rangle_B^{n+k+1}$，因此 $\langle t\rangle_B^{n+k+1}$ 包含 $\langle t\rangle_B^{n+k}$ 中的元素。因此，对于所有的 $s'_{\sigma'}\in\langle t\rangle_B^m$，$W(s')\leqslant W(s)$ 成立。

当 $m<n$ 时，假设 s_σ 已被选择，并且 $s'_{\sigma'}$ 也被选择，那么当 $m\geqslant 0$ 时，一定有 $\langle t\rangle_B^{n+m}$ 中的其它元素被选择，权重同样小于 $W(s)$。因为每个项只选择一次，因此这种情况是不可能的。证毕。

定义 9(利用近似算法封闭 tableau) 含分枝 $B_1,\cdots,B_k$ 的 tableau T 是 T_3 封闭的，如果

(1) 对于每一个分枝 $B_i(1\leqslant i\leqslant k)$ 存在一个析取 $D_i=(t_{i1}\neq s_{i1}\vee\cdots\vee t_{in_i}\neq s_{in_i})\in P(B_i)$，使得当 $1\leqslant j\leqslant n_i$，$l_{ij}$，$l_{ij}\geqslant 0$ 时，

$r^{ij}_{\rho_{ij}}\in\langle t_{ij}\rangle_{B_i}^{l_{ij}}$ 并且 $\underline{r}^{ij}_{\underline{\rho}_{ij}}\in\langle s_{ij}\rangle_{B_i}^{l_{ij}}$

这里 r^{ij} 和 $\underline{r}^{ij}$ 通过 MGU ρ_{ij} 是可合一的。

(2) 存在一个基替换 σ，使得替换 ρ_{ij}、ρ_{ij}、$\underline{\rho}_{ij}(1\leqslant i\leqslant k,1\leqslant j\leqslant n_i)$ 比 σ 更一般。

在 $P(B_1),\cdots,P(B_k)$ 中，对于每个项 t，一步一步地计算集合 $\langle t\rangle_{B_i}^0$，$\langle t\rangle_{B_i}^1$，$\cdots$，根据定义 4 判断 tableau T 是否为封闭的。

引理 1 如果对于 $n\geqslant 0$ 有 $s_\sigma\in\langle t\rangle_B$，那么对于任何一个比 σ 更一般的替换 τ，有 $(s^{\tau})_{\tau}\in\langle t\rangle_B$。

证明：关于 n 的归纳证明。

当 $n=0$ 时，$s_\sigma \in \langle t\rangle_B = \{t_{id}\}$。

那么，$s^\tau = t^\tau \in [t^\tau]_\tau$。

根据定义 4，得 $(s^\tau)_\tau \in \langle t\rangle_B$，引理 1 成立。

设 $n=k$ 时引理 1 成立。对于 $s_\sigma \in \langle t\rangle_B$ 和 $s_\sigma \in \Theta_n$ 分情况讨论。

当 $s_\sigma \in \langle t\rangle_B$ 时，$(s^\tau)_\tau \in \langle t\rangle_B$ 可直接推导出。

当 $s_\sigma \in \Theta_n$ 时，存在一个元素 $s'_{\sigma'} \in \langle t\rangle_B$ 可通过某一 s_σ 一步得到。在启发式下，通过 $s'\sigma$ 和等式 $G \in E(B)$ 得到项 s，并且有 $\tau' \not\phi \sigma$。

由于 $s'_{\sigma'} \in \langle t\rangle_B$ 且 τ 是一个替换，因此 $(s'^\tau)_\tau \in \langle t\rangle_B$，$s'^\tau \in [t^\tau]_\tau$。

由于 τ 存在更一般的替换 σ，s_τ 是在启发式的帮助下通过 s'^τ 和等式 $G \in E(B)$ 得到，因此 s^τ 在 $[t^\tau]_\tau$ 中，根据定义 4，得 $(s^\tau)_\tau \in \langle t\rangle_B$，引理 1 成立。证毕。

引理 2 如果 $r_\sigma \in \langle t\rangle_B$，那么对于 $n \geqslant 0$，存在一个元素 $r'_{\sigma'} \in \langle t\rangle_B^n$ 并且 $r'_{\sigma'} \gg r_\sigma$。

证明：如果 $r_\sigma \in \langle t\rangle_B$，那么根据定义 4，有 $r \in [t^\sigma]_\sigma$。

对于 $m \geqslant 0$，通过 $E(B)$ 中的等式序列 $E_1, \cdots, E_m$，从 $t\sigma$ 开始，经过中间步骤 $u^0, \cdots, u^m$ 推导出 r，即

$$t^\sigma = u^0 \xrightarrow{E_1} u^1 \xrightarrow{E_2} \cdots \xrightarrow{E_m} u^m = r = t_{id}$$

关于 $0 \leqslant k \leqslant m$ 归纳证明。

当 $k=0$ 时，存在唯一元素 $r^0_{\sigma_0} = t_{id} \in \langle t\rangle_B^n$ 包含 $u^0_\sigma = (t\sigma)_\sigma$，即 $r^0_{\sigma_0} \gg u^0_6$。

假设当 $k=n$ 时，对于 $l_k \geqslant 0$，存在一个元素 $r^k_{\sigma_k} \in \langle t\rangle_B^{l_k}$ 并且 $r^k_{\sigma_k} \gg u^k_\sigma$。

当 $k=n+1$ 时，在启发式 H 下，必然存在一个 $r'^{k}_{\sigma'_k} \gg r^k_{\sigma_k}$，也必然存在 $H(\langle t\rangle_B^{l_{k+1}-1}) = r'^{k}_{\sigma'_k} \gg r^k_{\sigma_k} \gg u^k_\sigma$。

另外存在一个替换 σ，使得 $r'^k_\sigma = r^{k\sigma} = u^{k\sigma} = u^k \xrightarrow{E_{k+1}} u^{k+1}$。

$\sigma'_k \varnothing \sigma$ 成立。根据算法 1，存在 $\sigma'_{k+1} \not\phi \sigma$，且满足 $u^{k+1}_{\sigma'_{k+1}} \in \Theta_{k+1}$ 成立。

同理，存在元素 $r^{k+1}_{\sigma'_{k+1}} \in \langle t\rangle_B^{k+1}$，且 $u^{k+1}_{\sigma'_{k+1}} \gg u^{k+1}_\sigma$。

当 $m=k$ 时，有 $r'_{\sigma'} = r^m_{\sigma_m} \in \langle t\rangle_B^n = \langle t\rangle_B^{l_m}$ 且 $r'_{\sigma'} \gg u^m_\sigma = r_\sigma$。证毕。

定理 3 T 是关于公式 ϕ 的 tableau，q 是 γ-规则应用的次数。

可靠性：如果 T 是 T_3 封闭的，那么在规范模型中，ϕ 是不可满足的。

完备性：如果 ϕ 在规范模型中是不可满足的，q 的限定次数足够高，那么 T 是 T_3 封闭的。

证明：（可靠性证明）假设基替换 τ 是一个比 σ 更一般的替换，根据定义 8，类似 τ 的所有替换均包含在 $\{\rho_{ij}, \varrho_{ij}, \underline{\varrho}_{ij} : 1 \leqslant i \leqslant k, 1 \leqslant j \leqslant n_i\}$ 中。这样，对于任意 $1 \leqslant i \leqslant k, 1 \leqslant j \leqslant n_i$ 有

$r^{ij}_{\rho_{ij}} \in \langle t_{ij}\rangle_{B_i}^{l_{ij}}$ 并且 $\underline{r}^{ij}_{\varrho_{ij}} \in \langle s_{ij}\rangle_{B_i}^{l_{ij}}$

根据引理 1，有

$(r^{ij}_\tau)_\tau \in \langle t_{ij}\rangle_{B_i}$ 且 $(\underline{r}^{ij}\tau)_\tau \in \langle s_{ij}\rangle_{B_i}$

这里 τ 是 r^{ij} 和 $\underline{r}^{ij}$ 通过 MGU ϱ_{ij} 的合一。因此有 $r^{ij}\tau = \underline{r}^{ij}\tau$ 成立，即 $\langle t_{ij}\rangle_{B_i}$ 和 $\langle s_{ij}\rangle_{B_i}$ 存在公共元素。根据定理 1 可知，公式 ϕ 是不可满足的。

（完备性证明）如果根据定理 1，ϕ 在规范模型中是不可满足的，q 的限定次数足够高。根据定义 3，存在一个基替换 σ，对于 T 上的每一个分枝 B，存在一个析取 $(t_1 \neq s_1 \vee \cdots \vee t_n \neq s_n) \in P(B)$，使得 $1 \leqslant i \leqslant n$，$[t_i^\sigma]_B = [s_i^\sigma]_B$。对于 $1 \leqslant i \leqslant k, 1 \leqslant j \leqslant n_i$，存在一个元素

$r^{ij}_\sigma \in \langle t_{ij}\rangle_{B_i} \cap \langle s_{ij}\rangle_{B_i}$，根据引理 2，$r_{\rho_{ij}} \in \langle t_{ij}\rangle_{B_i}^{l_{ij}}$ 且 $\underline{r}_{\varrho_{ij}} \in \langle s_{ij}\rangle_{B_i}^{l_{ij}}$，满足 $r_{\rho_{ij}} \gg r^{ij}_\sigma$ 且 $\underline{r}_{\varrho_{ij}} \gg r^{ij}_\sigma$。

同时存在替换 μ 和 ν，使得 $r^{ij}\mu = \underline{r}^{ij}\nu$，这里 μ、ν、ρ_{ij}、ϱ_{ij}、$\underline{\varrho}_{ij}$ $(1 \leqslant i \leqslant k, 1 \leqslant j \leqslant n_i)$ 是比 σ 更一般的替换。

有 $r^{ij\sigma} = \underline{r}^{ij\sigma}$ 成立，r^{ij} 和 $\underline{r}^{ij}$ 通过 MGU ϱ_{ij} 的可合一的，即 T 能被封闭。

4 算法举例

通过一个实例，详细地描述等式合一 tableau 及等价类计算方法，从中可以显示出方法的优点。

例 2 证明公式 $(\forall x)[g(f(x)) \approx c \wedge (f(x) \approx g(x) \vee \phi)] \rightarrow g(g(a)) \approx g(f(b))$ 为重言式。其中 ϕ 为不可满足的任意公式。$All(x, (g(f(x)) \approx c)\ \&\ (f(x) \approx g(x)\ v\ \phi)) \Rightarrow g(g(a)) \approx g(f(b))$

tableau 第一阶段的扩展如下：

$F\ (\forall x)[g(f(x)) \approx c \wedge (f(x) \approx g(x) \vee \phi)] \rightarrow g(g(a)) \approx g(f(b))$ √

$T\ (\forall x)[g(f(x)) \approx c \wedge (f(x) \approx g(x) \vee \phi)]$ √

$F\ g(g(a)) \approx g(f(b))$

$T\ g(f(x)) \approx c \wedge (f(x) \approx g(x) \vee \phi)$ √

$T\ g(f(x)) \approx c$

$T\ \ f(x) \approx g(x) \vee \phi$

$T\ \ f(x) \approx g(x) \qquad\qquad T\ \ \phi$

$*$

后面标注"√"的为不再使用的公式，第一阶段完成后，tableau 扩展已经完成，但 tableau 没有封闭。设未封闭分枝为 B，B 转换成集合 $E(B)$ 和 $P(B)$。

$E(B) = \{g(f(x)) \approx c, f(x) \approx g(x)\}$

$P(B) = \{g(g(a)) \neq g(f(b))\}$

Tableau 的第二阶段转换成计算 $\langle g(g(a))\rangle_B^n$ 和 $\langle g(f(b))\rangle_B^n$ 的合一元素。

在图 1 中，第一步 $P(B)$ 分成两部分。第二步由 $g(g(a))$ 利用等式②得到 $f(g(a))$ 和 $g(f(a))$，分别使用了替换 $\{g(a)/x\}$ 和 $\{a/x\}$。第三步由 $\langle g(f(b))\rangle_B^1$ 利用等式②得到 $f(f(a))$ 和 $g(g(b))$，分别使用了替换 $\{f(b)/x\}$ 和 $\{b/x\}$，由 $g(f(b))$ 利用等式①得到 c。第四步由 $g(f(a))$ 利用等式①得到 c。这样，由于 $c_{\{a/x\}} \in \langle g(g(a))\rangle_B^2$，$c_{id} \in \langle g(f(b))\rangle_B^1$，因此 tableau 是封闭的。

在例 2 中，如果使用 Fitting 方法证明同一公式为重言式，必须提高 γ 公式的使用次数。在同一启发搜索算法下，证明不等式 g(g(a))≠g(f(b)) 的不可满足性，将产生 15 个中间等式，并需要几次的自由变量替换。如果使用 Jeffrey 方法证明同一公式为重言式，等式规则的应用次数依赖于使用等式的次序，最好的情况下只需要 2 个分枝即封闭。然而，在较差的情况下，可以产生几百个分枝。

结论 本文提出利用等式合一处理含等词 tableau 方法，将 tableau 扩展分成两个阶段[8]。第一阶段是在处理等词之前，应用 tableau 扩展规则，完成所有的 tableau 扩展，在这个过程中要求对 γ 规则应用次数进行限制。第二阶段是处理等词，通过提取等式合一问题并求解解替换封闭 tableau，在启

发式的帮助下计算等价类的方法，限制等词的使用，以避免无关公式的产生，在效率和实现上都有较大的提高。

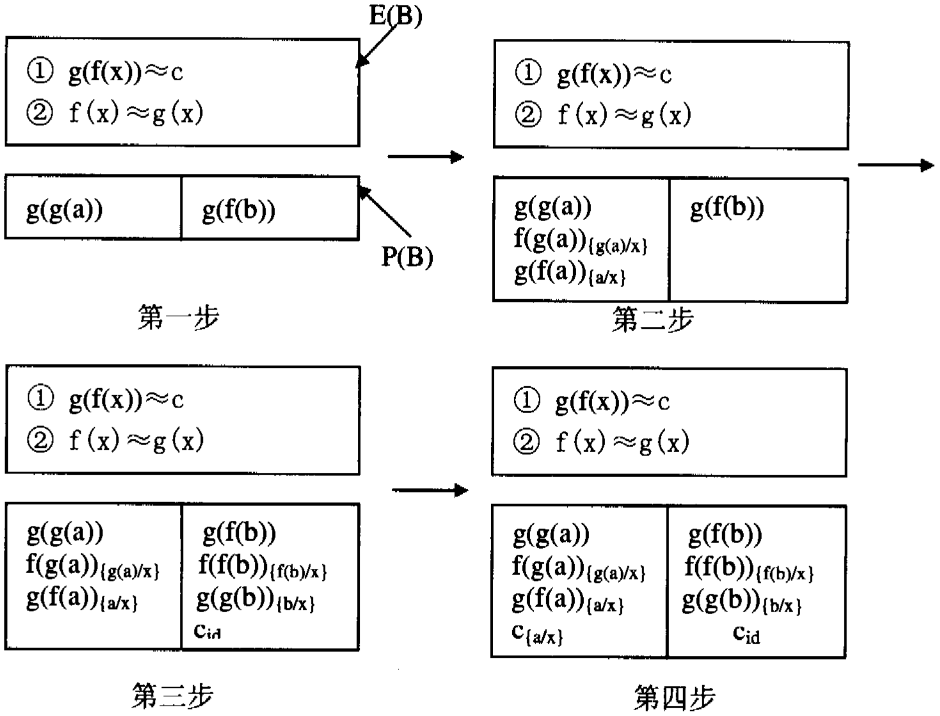

图 1　tableau 扩展的第二阶段

在机器实现时，可以在以下两方面进行改进，以提高效率：一是改进启发式算法，提高计算等价类的速度；二是将计算等价类的结果以树的结构进行存储，利用树的特点，进行搜索的优化，提高推理效率。

含等词的分阶段 tableau 方法很容易从经典逻辑扩充到非经典逻辑[9]，并适合于自动推理，因此具有很广阔的应用前景。

参 考 文 献

1　刘叙华．基于归结方法的自动推理．北京：科学出版社，1994

2　刘叙华，孙吉贵．NC-RUE-NRF 归结．软件学报，1995，6(2)：65～68

3　Fitting M. First-Order Logic and Automated Theorem Proving. New York：Springer Verlag，1996

4　Fitting M. Types and Tableau. New York：Springer Verlag，2000

5　刘全，孙吉贵．提高一阶多值逻辑 tableau 推理效率的布尔剪枝方法．计算机学报，2003，26(9)：1165～1170

6　Beckert B. Semantic Tableaux with Equality. Journal of Logic and Computation，1999(1)：39～58

7　Beckert B，Hähnle R. Analytic tableau. In：Bibel W，Schmitt P H，eds. Automated Deduction - A Basis for Application，colume 1. Kluwer，1998

8　Sun Ji-Gui，Liu Quan. An Algorithm for Two Separation Tableau with Equality. ICIIT 2002. 320～325

9　刘全，孙吉贵．非经典逻辑的语义 tableau 方法．计算机科学，2002，29(5)：72～75

(上接第 193 页)

是针对开发商而设计的，因此，用户能够在 E2Q 平台上自行设计管理信息系统，也就是说用户可以自己调整改进系统而不需要请专业软件公司开发。

E2Q 推行的“零编程”理念可以帮助不同行业、不同企业的用户快速地部署业务流程应用系统，并可以便捷地对业务流程进行更新。与传统的软件开发模式相比，使用 E2Q 平台可以将软件开发的周期缩短 50%以上。

E2Q 平台支持分布式开发模式，同一个 E2Q 应用系统的设计开发可以通过多个设计人员同时在网络上进行。

(2)快速适应需求(业务规则)的变更　E2Q 平台既是开发平台，也是运行支撑平台，E2Q 平台提供高效的业务建模工具和各种应用包，可以由用户自己完成动态调整与生成企业的业务模型，实现零编程。E2Q 的业务规则不是以程序的方式形成的，而是以数据的形态存放于数据库中，一旦用户系统的业务规则需要改变，通过 E2Q 设计器就可以轻松、快速实现。E2Q 设计器提供了记录表设计、过程设计、图表设计、报表设计等设计工具，完全能胜任用户级的修改。修改后的业务规则可以立即通过设计器上传功能动态上传到应用系统中，而不需停止应用系统的运行。同样，用户还能够通过 E2Q 平台不断地增加新的应用子系统，新的业务规则，大大延长应用软件系统的生命周期。

(3)E2Q 平台是针对中小规模应用而设计的　E2Q 平台由于是针对中小规模应用而设计的，因此整个平台是集成化的，在部署与实施过程中，实施成本远远低于以上所提及的业务流程系统。

结论　业务流程管理平台的出现使软件平台多了一个具有革命性意义的战略层级，为降低大型复杂系统的实现难度提供了新的途径。这也就是说，它使大型软件复杂应用系统的体系结构出现了新的变化，也会对复杂应用系统本身产生重大的影响。

E2Q 平台是业务流程管理平台的一个具体实例。用 E2Q 平台设计业务系统与传统的开发过程相比要少很多时间，E2Q 平台在设计实现一气呵成，返工修改的代价也很小；E2Q 的测试主要在业务流程上，不会在一些技术问题上花费时间。通过统一构架可以实现：面向全业务领域的信息化，成为集成化信息底层平台；支持业务再造；支持业务应用灵活重构；支持多层次的二次开发，提高系统可扩展性；分布式的 Web 构架模式。

参 考 文 献

1　王欣．管理信息系统．北京：中国水利水电出版社，2004

2　徐萍，毛江华．软件平台革命．计算机世界，2003-03-03(A15-A17)

3　CusumanoM A，张云涛，龚玲，等译．软件业的生存之道．北京：电子工业出版社，2005

4　Hohmann L，蓝莉，曾永和译．超越软件架构：创建和维护优秀解决方案．北京：中国电力出版社，2005