完全／不完全なルールに対する推論手続き

井戸 譲治[†] 馬場口 登[†]

† 大阪大学工学部

本稿では，ルール形式で書かれた完全／不完全な知識を対象とする推論手続きについて述べる．知識表現は自己認識論理に基づいたものであり，推論手続きもその意味論・構文論に従っている．本手続きは論理式が拡張世界に含まれるか否かを SLD 導出による反駁の有無により決定する．様相リテラルに対しては副次的な推論木を生成し，また負リテラルに対しては部分計算を行うことにより SLD 導出機構を適用できるようにしている．さらに，多重拡張を持つような場合に対しても適切に動作する．

A Reasoning Procedure for Complete/Incomplete Rules

George IDO[†] Noboru BABAGUCHI[†]

† OSAKA University

In this paper,we discuss a reasoning procedure for complete/incomplete knowledge represented by rule forms. Knowledge representation is based on autoepistemic logic(AEL) and its reasoning procedure obeys semantics and syntactics of AEL. This procedure decides whether a formula belongs to a chosen expansion by attempting to construct its refutation. Subsidiary reasoning trees for modal literals and a partial computation process for negative literals are introduced to apply SLD-resolution mechanism. Moreover,this procedure can work appropriately where there are multiple expansions.

1 はじめに

近年，様々な角度からデータベース技術・理論への検討が加えられているが，一階述語論理を数学的基礎としてデータモデルや問い合わせ処理の意味を明確にし，さらに表現力，質問応答能力を向上させようとする試みがある．このようなアプローチによるものは演繹データベースと呼ばれる [1].

演繹データベースは，一階述語論理に基づく言語を用いて表現された事実と規則とから構成される．最大の特徴は，単なる事実の検索のみならず，演繹推論によりデータベースに陽に記述されていない事実をも導き出すことができることであろう．従って，これまでのデータベースよりも柔軟な質問応答処理が可能であり，知識処理システムの一種として位置づけることができる．しかし演繹推論は，前提となる知識が常に正しいことに立って行われる推論であり，例外を含む知識や仮説的な知識など，常に成り立つとは限らない，いわゆる不完全な知識を扱うことはできない．現実に存在する知識は不完全なものが多く，このような不完全な知識を扱う推論機構の実現は重要な課題といえる．

さて，このような不完全な知識による推論のモデルとして非単調推論があり，これを定式化する論理体系は非単調論理と呼ばれる．非単調論理の一つである自己認識論理 [2] は,「信じている」という意味の様相記号 L を通常論理の体系に導入することにより，内省的なエージェントが自身の信念に基づいて行う推論をモデル化するもので,「αならば通常βである」といったデフォルト型の知識も自然に表現できる．近年盛んに研究され，意味論が整備されてきており，他の非単調論理との関係についても明らかにされつつあるが，一階計算上での自己認識論理は非決定的であることが知られており，自己認識論理に基づく推論機構をそのままの形で計算機上に実現することは困難である．

本稿では，自己認識論理に基づく知識表現に対する推論手続きについて，演繹データベースの高度化という観点に立って検討する．知識表現は，実用上の観点から，ルール形式のもののみに制限する．即ち，演繹データベースにおける事実と規則に加え,「αならば通常βである」といったデフォルト型の知識を対象とする．推論手続きは，基本的にはSLD導出を用いて反駁を試みるものであるが，負リテラルを扱う必要から，部分計算を行いプログラム節の変換を行う．また，多重拡張による無限ループを回避するために，節を拡張世界で指標付けしており，多重拡張の際には，どのような拡張世界で真となるのかを明示している．以下，まず論理的基盤となる自己認識論理について概観したあと，推論手続きについて説明していく．

2 自己認識論理

自己認識論理 [2] は,「信じている」という意味を持たせた様相記号 L を導入することにより，内省的なエージェントが自身の信念に基づいて行う推論をモデル化するものである．自己認識論理の解釈は，次のように定義される．

定義 2.1 (解釈) [Moore85][Konolige88]
Iを一階論理の解釈，Tを理論とする．以下を満たす真理値の割当てを解釈と呼び，2字組 $\langle I,T\rangle$ で表す．
(1) ϕ を通常式（様相記号を含まない式）とすれば，$I \models \phi$ のとき，またそのときに限り $\langle I,T\rangle$ でϕ が真．
(2) $\phi \in T$ のとき，またそのときに限り $\langle I,T\rangle$ で $L\phi$ が真． ■

また，推論結果に相当する，即ちエージェントが信じている式の集合は拡張世界と呼ばれる．Moore の定義した拡張世界は人間の直観にそぐわないことがあり，その後 Konolige [3] や Marek と Truszczynski[4] がいくつかの拡張世界の定義を与えている．ここでは，Marek らによる反復的拡張世界 (iterative expansion) の定義に従う．

定義 2.2 (反復的拡張世界) [Marek and Truszczynski89] まず，オペレータΓを次のように定義する．

$$\Gamma(S) = Th(S \cup \{L\phi \mid \phi \in S\})$$

次に

$$\Gamma_0^T(F) = Th(F \cup \{\neg L\phi \mid \phi \notin T\})$$

$$\begin{aligned}\Gamma_{n+1}^T(F) &= \Gamma(\Gamma_n^T(F)) \\ &= Th(\Gamma_n^T(F) \cup \{L\phi \mid \phi \in \Gamma_n^T(F)\})\end{aligned}$$

さらに

$$\Gamma^T(F) = \bigcup_{n=0}^{\infty} \Gamma_n^T(F)$$

このとき，

$$T = \Gamma^T(F)$$

を満たす理論 Tを前提 Fの反復的拡張世界 *(iterative expansion)* という． ■

この定義は，$L\phi$が導かれるためにはまずϕが導かれなければならないことを示しており，より人間の直観に沿ったものとなっている．以後，反復的拡張世界を単に拡張世界と呼ぶ．

3 知識表現と推論の枠組み

はじめに述べたように，ここで対象とする式はルール形式のものに制限している．「αならばβである」という常に成り立つルールを完全ルールと呼び，一階論理を用いて$\beta \leftarrow \alpha$と表記する．不完全な知識としては，「αならば通常βである」というデフォルト型のもののみを考え，これを不完全ルールと呼び，$\beta \Leftarrow \alpha$と表記する．この式の意図された意味は「αが成り立つとき，βが矛盾しなければβを結論する」というものであり，自己認識論理では$\beta \leftarrow L\alpha \wedge \neg L \neg \beta$と表される．

さて，計算効率を考えると，見かけ上負リテラルが現れない確定節を対象とするのが良いが，確定節集合からは負リテラルは演繹されないため，自己認識論理による上のような不完全ルールは無意味なものとなってしまう．そこで本稿では，「αとβが同時に成り立つことは矛盾する」という意味の一階論理式$\perp \leftarrow \alpha, \beta$を用いることにする．これを制約と呼ぶ．

定義 3.1 (プログラム) 以下のような論理式をプログラム節と呼ぶ．

(1) ファクト：$p(t_1, \cdots, t_n)$
(2) 完全ルール：$A \leftarrow B_1, \cdots, B_m$
(3) 不完全ルール：$A \Leftarrow B_1, \cdots, B_m$
(4) 制約：$\perp \leftarrow B_1, \cdots, B_m$

ここで，p は n 引数述語記号，各 t_iは定数，A及び各 B_jは原子式で，関数を含まないものとする．また，$\perp$は恒偽命題を表す記号である．プログラム節の集合をプログラムという． ■

(3) は自己認識論理式 $A \leftarrow LB_1, \cdots, LB_m, \neg L \neg A$ と等価である．ここで以後の議論で用いる用語について説明しておく．ϕ をリテラルとするとき，$L\phi, \neg L\phi$ の形の式をそれぞれ正の様相リテラル，負の様相リテラルという．正の様相リテラルは様相原子式とも呼ばれる．また，$\leftarrow l_1, \cdots, l_m$ の形の式をゴールという．ここで各 l_i はリテラルまたは様相リテラルである．以後，A, B, Cは原子式を，Mは様相リテラルを，lはリテラルまたは様相リテラルを表すものとする．

次にこの枠組みでの推論を規定する．自己認識論理において推論結果に相当するのは拡張世界であり，拡張世界に含まれる式を，前提から導かれる事実と見なすことができる．しかし拡張世界は複数存在する場合があるため，注意が必要となる．全ての拡張世界に含まれる式を推論結果とすると，結論として得られることが少なすぎる．例えば，次のような例 [5] を考えよう．

$$block(a),\ \ block(b),\ \ block(c),$$
$$heavy(a),\ \ heavy(b),\ \ heavy(c),$$
$$ontable(x) \Leftarrow block(x), heavy(x)$$
$$\perp \leftarrow ontable(a), ontable(b)$$
$$(a, b, c \text{ は定数})$$

このプログラムの拡張世界は次の 2 つとなる．

$$T_1 = Th(\{ontable(a), \neg ontable(b), ontable(c)\})$$

$$T_2 = Th(\{\neg ontable(a), ontable(b), ontable(c)\})$$

この場合，全ての拡張世界に含まれる式を推論結果とすると，$ontable(c)$ しか得られない．しかし，「$ontable(a)$ が真のときには $ontable(b)$

が偽，$ontable(a)$ が偽のときには $ontable(b)$ が真」という推論結果が得られるのが望ましいであろう．一般に多重拡張は，「αを成り立たせるとするとβが成り立たなくなり，βを成り立たせるとするとαが成り立たなくなる」という状況を示している．そこで，ある拡張世界に含まれる式とともに，そのために拡張世界から除外された式を提示するのが良いと思われる．以上の議論から，推論を次のように定める．

定義 3.2 (推論) 与えられたプログラム Pと入力 A（原子式の連言）に対し，Aのある基礎例 $A\theta$ を含む$T \in \mathrm{Ex}(P)$, について，$\langle A\theta, PB, NB\rangle$ を返し，そうでないとき *no* を返す手続きを，推論という．但し，*PB,NB* はそれぞれ Tに含まれるリテラル，含まれないリテラルの集合であり，$\mathrm{Ex}(P)$ は Pの拡張世界の集合である． ■

4　推論手続き

本推論手続きは，基本的には，SLD導出*に基づく導出によって論理式の反駁が生成されることにより，その論理式がプログラムの拡張世界に含まれるとするものである．様相リテラルの導出は，その解釈及び拡張世界の定義から，その様相リテラルの対象式が反駁を持つか否かによって行われる．この際，負リテラルの証明が必要となるが，SLD 導出を適用できるように，部分計算を行ってプログラム節を変換している．また，このように様相リテラルの真偽を対象式の反駁の有無で決定する方法の場合，多重拡張の際に無限ループに陥ってしまうため，それを回避するための制御を行っている．以下，これらについて詳述する．

4.1　部分計算

様相リテラルを通常のリテラルと同様に導出に用いることを考える．本稿では対象とする式を定義 3.1 のように制限するため，プログラムそのままでは様相リテラルの導出は起こり得ない．しかし，拡張世界の定義より，正負の様相リテラル $L\phi, \neg L\phi$はそれぞれ，ϕが帰結されること，されないことにより帰結されるため，ϕの導出による証明の結果によって様相リテラルを新たに生成し，それを用いて導出を行うことになる．

プログラムは見かけ上負リテラルを含まないので，SLD 導出の使用が考えられる．しかし，不完全ルールの本体中の負の様相リテラルは様相記号のスコープ内に負リテラルを持つため，$P \cup \{\leftarrow \neg A\}$ という導出を行う必要がある．従って，SLD 導出をそのまま適用することはできない．

ところが，確定節集合からは負リテラルは演繹され得ないので，$\neg A$ の反駁には必ず制約を用いることになる．従って，制約$\bot \leftarrow A_1, \cdots A_m$ の本体$\leftarrow A_1, \cdots, A_m$を頂上節とする線形導出がゴール$\leftarrow A$ に到ることにより，$\neg A$ を証明することができる．この場合，線形導出の中心節は常に負リテラルの選言なので，SLD 導出を用いることができる．

しかし，この方法ではひとつの負リテラルの証明のために全ての制約について導出を行う必要があり，推論のつどこれを行うのは負担が大きすぎる．また，どのような負リテラルであるかに関わらず制約からの導出は同じであるので，負リテラルの証明ごとにこれを行うのは意味がない．そこで，この導出については，プログラムが与えられた時点で，部分計算という形で行っておくのが良いと思われる．これを次のように形式化する．

定義 4.1 (計算規則) 計算規則とは，以下のようなゴールから原子式または様相リテラルを選択する規則である．

*R1:*原子式のみを左から選択する．
*R2:*原子式または様相原子式を左から選択する．
*R3:*原子式または正負の様相リテラルを左から選択する． ■

定義 4.2 (負のルール) 頭部が負リテラル，本体が様相リテラルの連言である式を負のルールという． ■

定義 4.3 (部分計算) 以下のような手順で負のルールを得る手続きを部分計算という．

*ここでは，節の頭部のみを導出に用いる手続きを指すものとする．

(1) 制約$\perp \leftarrow A_1, \cdots, A_j$ について，$\leftarrow A_1, \cdots, A_j$ をトップゴールとするSLD導出を，計算規則を *R1* とし，不完全ルールを用いずに行う．最終ゴール$\leftarrow B_1, \cdots, B_k$ のどの原子式も，プログラム中のファクトあるいは完全ルールの頭部と単一化できない．
(2)(1) で得られたゴール$\leftarrow B_1, \cdots, B_k$ について，計算規則を *R1* とし，不完全ルールを用いてSLD導出を行う．最終ゴールのうち，
$\leftarrow LC_1, \cdots, LC_l, \cdots, \neg L \neg B_1, \cdots, \neg L \neg B_k$
という形のもののみ残し，他は除外する．
(3)(2) で得られたゴール
$\leftarrow LC_1, \cdots, LC_l, \cdots, \neg L \neg B_1, \cdots, \neg L \neg B_k$
に対しトップゴールを$\leftarrow C_1, \cdots, C_l$ とし，計算規則を *R2* としてSLD導出を行う．最終ゴールのうち，空節あるいは$\leftarrow \neg L \neg R_1, \cdots, \neg L \neg R_m$ という形のもののみ残し，他は除外する．
(4)(3) で得られたゴールに対し，それまでの導出で行った代入をθとする．この代入θを，(1) で得たゴール$\leftarrow B_1, \cdots, B_k$ に施し，$\leftarrow (B_1, \cdots, B_k)\theta$ を得る．
(5)(1)〜(4) を各制約式について行い，得られたゴール$\leftarrow (B_1, \cdots, B_k)\theta$ の包含関係を調べ，冗長なものは除外する．
(6)(5) で得られたゴールを
$\neg B_i \leftarrow B_1, \cdots, B_{i-1}, B_{i+1}, \cdots, B_k$
の形（全ての対偶を含む）に変形する．このうち本体が空でないものは，不完全ルールとの導出形
$\neg B_i \leftarrow LC_1, \cdots, LC_l, \cdots, \neg L \neg B_1, \cdots, \neg L \neg B_k$
を生成する． ■

この部分計算では，頂上節及び全ての側節をプログラム節とするSLD導出を行っているため，最終的に得られる節はプログラムの論理的帰結となっている．これが単一節である場合には即座に負リテラルが導かれるが，本体が空でない場合，多重拡張によって無限ループがの生じるため，注意が必要となる．これについては次節で述べる．

4.2 多重拡張による無限ループ

制約$\perp \leftarrow A_1, \cdots, A_l$ とファクト，完全ルールの導出形として，節

$$\leftarrow B_1, \cdots, B_k \tag{1}$$

が得られたとする．$B_1, \cdots, B_k$ が同様に得られた他の節には含まれず，かつ $B_1, \cdots, B_k$ を頭部に持つ不完全ルールが存在すると仮定する．いま，$\neg B_i$ をSLD導出により証明することを考える．確定節は負リテラルを演繹し得ないということと，上述の仮定から，式 (1) を変形した

$$\neg B_i \leftarrow B_1, \cdots, B_{i-1}, B_{i+1}, \cdots, B_k \tag{2}$$

のみを考えれば良い．この式と不完全ルールの導出形として

$$\begin{aligned} \neg B_i \leftarrow LC_{i1}, \ \cdots \ , LC_{im}, \\ \neg L \neg B_1, \ \cdots \ , \neg L \neg B_{i-1}, \\ \neg L \neg B_{i+1}, \ \cdots \ , \neg L \neg B_k \end{aligned} \tag{3}$$

を得る．ここで，導出に用いた不完全ルールの本体が成立しているとすれば，

$$\begin{aligned} \neg B_i \leftarrow \neg L \neg B_1, \ \cdots \ , \neg L \neg B_{i-1}, \\ \neg L \neg B_{i+1}, \ \cdots \ , \neg L \neg B_k \end{aligned} \tag{4}$$

が得られる．負の様相リテラル$\neg L\phi$ の真偽をϕの反駁の有無に帰着させる場合，ϕ の証明を副次的に行なう必要があるが，今の場合，$\neg L \neg B_j (j \neq i)$ を証明するためにゴール$\leftarrow \neg B_j$ から導出を始めると，上述の仮定から，式 (2) の対偶より得られる節

$$\begin{aligned} \neg B_j \leftarrow \neg L \neg B_1, \ \cdots \ , \neg L \neg B_{j-1}, \\ \neg L \neg B_{j+1}, \ \cdots \ , \neg L \neg B_k \end{aligned} \tag{5}$$

を用いることになるため，無限ループに陥ってしまう．この無限ループの理由を考える．

式 (1) と $B_1, \cdots, B_k$ の不完全ルールの導出形として

$$\begin{aligned} \leftarrow \neg L \neg B_1, \ \cdots \ , \neg L \neg B_k \\ (= L \neg B_1 \vee \ \cdots \ \vee L \neg B_k) \end{aligned} \tag{6}$$

が得られる（各不完全ルールの本体が成立しているという仮定をしていることに注意）が，これはプログラムの論理的帰結となっている．様相リテラルの真理値は2章で述べたように拡張世界

によって決定されることから，式(6)は拡張世界が$\neg B_1, \cdots, \neg B_k$のうちの少なくともひとつを含むことを要請している．$\neg B_i$を含む拡張世界を考えると，$\neg B_i$は，上述の仮定から式(4)から導かれなければならず，この本体が成立していなければならない．つまり，$\neg B_i$を導くためには，拡張世界が$\neg B_1, \cdots, \neg B_{i-1}, \neg B_{i+1}, \cdots, \neg B_k$を含まないことが必要となる．従って式(4)と(6)から，拡張世界は$\neg B_1, \cdots, \neg B_k$のうちいずれか一つを含み他は含まないといえる．これは拡張世界が複数存在することを意味しており，適切に推論を進めるには，このうちの一つを選び出す必要がある．先の無限ループは，導出によるトップダウン的な推論では拡張世界を選び出すことができないために生じたものであることがわかる．

以上のことから，多重拡張の際には，導出を中断して，この時点で考えられる全ての拡張世界（上の例ではk通り）について場合分けして証明を進めるか，あるいは何らかの基準を用いて拡張世界を選択するかしなければならない．また，どちらの場合も，複数あるうちの特定の拡張世界についての証明であることを明示する必要がある．そこで，次のように節の指標付けを行う．

定義 4.4 (拡張世界で指標付けされた節) 三字組$\langle G, PB, NB\rangle$を拡張世界で指標付けされた節という．ここでGは$M_1, \cdots, M_m \leftarrow l_1, \cdots, l_n$という形の節であり，$PB, NB$はそれぞれ拡張世界が含む式，含まない式の集合である．PBを正の信念，NBを負の信念と呼ぶ．■

例えば，拡張世界で指標付けされた節$\langle \neg Lp, \{q\}, \{r\}\rangle$は$q$を含み$r$を含まない拡張世界で$\neg Lp$が真となることを示している．

多重拡張であるか否かは，上述の仮定(1) $B_1, \cdots, B_k$が他の節に含まれない(2)これらを頭部に持つ不完全ルールが存在する(3)これら不完全ルールの本体が成立している，を満たしているかどうかにより判断するが，(1)は説明を簡単にするための強すぎる条件で，実際には，節をリテラルの集合とみなし，どの節も他の節を包含しなければ良い．

なお，現時点では拡張世界を選択するための適切な基準が得られなかったため，場合分けして証明を続けるものとして推論手続きを定義する．

4.3 推論木

次に推論木を定義する．

定義 4.5 (推論木) プログラムをP，ゴールをGとする．$P \cup G$の推論木とは，以下の条件を満たす木である．

(1) 木の各節点は信念集合で指標付けされた節でラベル付けされ，各枝はプログラム節または信念集合で指標付けされた節でラベル付けされる．

(2) 木の根節点は$\langle G, \{\}, \{\}\rangle$である．

(3) 木のある節点を$\langle M_1, \cdots, M_m \leftarrow l_1, \cdots, l_n, PB, NB\rangle$とし，$l_1$が$A$または$LA$ (Aは原子式)という形の式であるとする．

a)AがファクトA'と単一化代入θにより単一化可能であるとき，子節点を持つ．子節点は$\langle (M_1, \cdots, M_m \leftarrow l_2, \cdots l_n)\theta, PB, NB\rangle$

b)Aが完全ルール$A' \leftarrow B_1, \cdots, B_l$の頭部$A'$と単一化代入$\theta$により単一化可能であるとき，子節点を持つ．子節点は$\langle (M_1, \cdots, M_m \leftarrow B_1, \cdots, B_l, l_2, \cdots l_n)\theta, PB, NB\rangle$

c)Aが不完全ルール$A' \Leftarrow B_1, \cdots, B_l$の頭部$A'$と単一化代入$\theta$により単一化可能であるとき，子節点を持つ．子節点は$\langle (M_1, \cdots, M_m \leftarrow LB_1, \cdots LB_l, \neg L\neg A, l_2, \cdots, l_n)\theta, PB, NB\rangle$

(4) 木のある節点を$\langle M_1, \cdots, M_m \leftarrow l_1, \cdots, l_n, PB, NB\rangle$とし，$l_1$が$\neg L\neg A$ (Aは原子式) という形の式であるとする．

a)$\neg A$がNBに含まれるかまたは信念集合で指標付けされた節$\langle \neg L\neg A, PB', NB'\rangle$が存在するとき，子節点を持つ．子節点は$\langle M_1, \cdots, M_m \leftarrow l_2, \cdots, l_n, PB \cup PB', NB \cup NB'\rangle$

b)$\neg A$がPBに含まれるかまたは信念集合で指標付けされた節$\langle L\neg A, PB', NB'\rangle$が存在するとき，子節点を持つ．子節点は$\langle M_1, \cdots, M_m \leftarrow l_1, \cdots, l_n, PB \cup PB', NB \cup NB'\rangle$ただし，この子節点は子節点を持たない．

(5) 木のある節点を$\langle \leftarrow \neg A, PB, NB\rangle$とする．$\neg B$が負のルール$\neg B' \leftarrow LC_1, \cdots, LC_l, \neg L\neg B_1,$

$\cdots, \neg L\neg B_k$ の頭部$\neg B'$と単一化代入 θにより単一化可能であるとき，子節点を持つ．子節点は $\langle (L\neg B_1,\cdots,L\neg B_k \leftarrow LC_1,\cdots,LC_l)\theta, PB, NB\rangle$

(6) 節点 $\langle G, PB, NB\rangle$ で，G が空節であるものを成功節点，G の本体が空であるものを多重拡張節点，G が空でなく (2)〜(5) に当てはまらないものを失敗節点という．これらの節点は子節点を持たない． ■

様相リテラル $L\phi$，$\neg L\phi$がϕの反駁の有無により導出に用いられることはすでに述べたが，SLD 導出では部分木が反駁を形成する（SLD 木のあるゴール$\leftarrow A, B$の子孫ゴールに$\leftarrow B\theta$が存在する場合，この間の部分木は解をθとする A の反駁となる）ので，正の様相リテラルについては，上のようにトップレベルの推論木に埋め込んでもよい．従って，副次的な推論木の生成は負の様相リテラルに対してのみ行われる．この副次的な推論木によって，様相リテラルの導出のための節が生成される．

どのような節を生成するかは，推論木の葉節点の状態によって決まる．$P\cup\{\leftarrow\neg A\}$ の推論木において，成功節点は$\neg A$ の反駁の成立を示しているから $L\neg A$ が生成され，失敗節点しか持たない場合は$\neg L\neg A$ が生成される．さて，多重拡張であることは前節に挙げた 3 つの仮定を満足することにより判断されるが，部分計算を終えた時点で，仮定 (1)(2) は成立している．多重拡張節点は仮定 (3) の成立をチェックするためのもので，定義からわかるように $P\cup\{\leftarrow\neg A\}$ の推論木では最初の導出の直後に節の本体に現れるのは$\neg A$ の証明に必要な不完全ルールの本体の連言であり（負のルール中の負の様相リテラルを導出節の頭部に置くのはこの不完全ルールの本体からの導出に現れる負の様相リテラルと区別するためである），多重拡張節点に到達した時点で仮定 (3) が成立することになる．これを形式化する．

定義 4.6 (拡張世界で指標付けされた節の生成) 根節点を $\langle\leftarrow\neg A, PB, NB\rangle$ とする，$P\cup\{\leftarrow\neg A\}$ の推論木において

(1) 失敗節点のみを持つとき，節 $\langle\neg L\neg A, PB, NB\rangle$ を生成する．

(2) 成功節点 $\langle\Box, PB', NB'\rangle$ を持つとき，節 $\langle L\neg A, PB', NB'\rangle$ を生成する．

(3) 多重拡張節点 $\langle L\neg B1,..,L\neg Bn\leftarrow, PB', NB'\rangle$ を持つとき，節

$\langle L\neg A, PB'\cup\{\neg A\}, NB'\cup\{\neg B_1,\cdots,\neg B_n\}\rangle$

$\langle\neg L\neg A, PB'\cup\{\neg B_1\}, NB'\cup\{\neg A,\neg B_2,\cdots\neg B_n\}\rangle$

$\langle\neg L\neg A, PB'\cup\{\neg B_2\}, NB'\cup\{\neg A,\neg B_1,\cdots\neg B_n\}\rangle$

$\vdots$

$\langle\neg L\neg A, PB'\cup\{\neg B_n\}, NB'\cup\{\neg A,\neg B_1,\cdots\neg B_{n-1}\}\rangle$

を生成する． ■

4.4 推論手続きの実行例

ここでは，2 章で挙げた例に対して推論手続きを行った様子を示す．まず部分計算の流れを追うと．プログラム中の制約に手続き (1),(2) を施すと，$\leftarrow Lblock(a), Lheavy(a), \neg L\neg ontable(a), Lblock(b), Lheavy(b), \neg L\neg ontable(b)$ を得る．手続き (3) により$\leftarrow block(a), heavy(a), block(b), heavy(b)$ をトップゴールとして導出を行なうと，空節に至る．この間の代入は恒等代入であり，手続き (4) により$\leftarrow ontable(a), ontable(b)$ が得られ，(5),(6) を行なった結果，最終的に$\neg ontable(a) \leftarrow Lblock(b), Lheavy(b), \neg L\neg ontable(b)$ と$\neg ontable(b) \leftarrow Lblock(a), Lheavy(a), \neg L\neg ontable(a)$ が得られる．

このプログラムに対しゴールを$\leftarrow ontable(x)$ とした推論木を図 1 (a) に示す．左側の枝について説明すると，4 段目において負の様相リテラル$\neg L\neg ontable(a)$ がサブゴールとなるため，$\leftarrow\neg ontable(a)$ を根とする推論木（図 1 (b)）を生成する．この推論木の葉節点が多重拡張節点となるため，ここでは 2 つの節を生成している．この推論結果は

- $ontable(c)$ が真
- $\neg ontable(b)$ を含み$\neg ontable(a)$ を含まない拡張世界で $ontable(a)$ が真
- $\neg ontable(a)$ を含み$\neg ontable(b)$ を含まない拡張世界で $ontable(b)$ が真

となる．

5 おわりに

本稿では，自己認識論理に基づく完全／不完全なルールに対する推論手続きを提案した．本手続きでは，推論実行時における負リテラル導出のための計算の手間を軽減するために部分計算という形で前もってプログラム節の変換を行っているが，実際には，部分計算の段階で負のルールを多重拡張節にまで変換できると思われる．また，多重拡張の全ての場合について推論を行うのは，計算の手間という観点からは好ましいものではなく，拡張世界選択のための何らかの基準を設定するか，あるいは，質問として与えられた式を成り立たせるような拡張世界についてのみ推論を行うなどの方策が必要となろう．

6 謝辞

本研究を遂行するにあたり御指導賜わった手塚慶一大阪大学名誉教授に感謝する．なお，本研究の一部は文部省科学研究費（重点領域研究（知識科学）No.04229211）の補助による．

参考文献

[1] J.Minker. Foundations of Deductive Databases and Logic Programming. *Morgan Kaufmann*, (1987)

[2] R.C.Moore. Semantical Considerations on Non-Monotonic Logic. *Artificial Intelligence*, vol25,pp75-94 (1985)

[3] K.Konolige. On the Relation between Default and Autoepistemic Logic. *Artificial Intelligence*, vol35,pp343-382 (1988)

[4] W.Marek and M.Truszczynski. Relating Autoepistemic and Default Logics. In *Proceedings,1st International Conference on Principle of Knowledge Representation and Reasoning*, pp276-288 (1989)

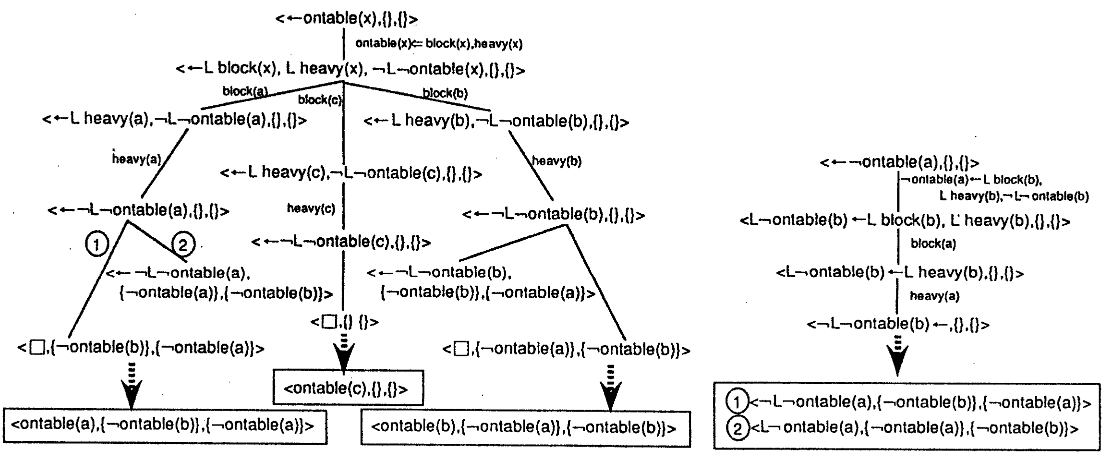
[5] V,Lifschitz. Benchmark Problem for Formal Nonmonotonic Reasoning. In *Proceedings,2nd Workshop on Non-Monotonic Reasoning*, pp.202-219 (1988)

[6] J.W.Lloyd. Foundations of Logic Programming 2nd,Extended Edition. *Springer-Verlag* (1987)

[7] 櫟 粛之，石田 亨．デフォルト論理に基づく知識プログラミングシステムとそのプログラム変換の理論的枠組み．人工知能学会誌 vol.7 No.2 , pp.280-291 (1992)

[8] 森 有一，馬場口登，手塚慶一．自己認識的データベースについて．情報処理学会研究報告 91-AI-74-2 (1991)

[9] 井戸譲治，馬場口登，手塚慶一．完全／不完全な知識に対する証明手続き．情報処理学会第45回全国大会予稿集 5G-2 (1992)



(a) (b)

図1 実行例