論　文

# ソートと述語の二つの階層をもつ論理のホーン節計算の完全性

兼岩　憲[†]　　東条　敏[†]

# The Completeness of a Horn Clause Calculus with Sort and Predicate Hierarchies

Ken KANEIWA[†] and Satoshi TOJO[†]

**あらまし**　本論文では，構造的なデータを記述できる知識表現言語の拡張として，ソートと述語の別々の階層表現を導入したホーン節計算による完全な論理体系を提案する．特にソートとは性質の異なった述語の階層では，それぞれ固有の引数構成をもつ多項述語から上位述語を導出する場合，引数のずれを解消するために引数を補充/削除しなければならない．本論文では，引数を操作する手続きを含んだ階層に関する推論規則を追加する一方で，述語の階層関係と引数操作をモデルに反映させる制約を言語解釈に組み込む．その結果，柔軟な引数構成を許した推論体系に対する意味論を定義し，健全性・完全性を証明する．

**キーワード**　オーダーソート論理，ホーン節，述語の階層関係

## 1. まえがき

知識表現言語には，記述力や推論力の向上といった要件とともに，言語体系としての理論的な基礎が求められることから，論理をベースにした言語が多く提案されてきた．一方，純粋な論理の研究では，数学の異なる対象（実数，ベクトル空間，グラフ理論など）を扱うために多ソート論理 [5] やオーダーソート論理 [8] が形式化されている．そうした理論的な基礎を備えたソート論理は，ソート表現とその階層が構造的なデータを記述する候補として採用され，近年，知識表現の分野で再び注目されるようになった [9]．特に，導出原理に基づいたソート論理 [10], [11] は，ソート項への単一化によって計算機上での推論手続きを形式化した完全な体系である．

しかし，ソート表現を取り入れた多くの知識表現言語では，ソート階層と同様に多項述語に対して階層構造を構築することはできない．ソート表現は，論理式において項の中に含まれる表現であるため，論理体系では多項述語とは別の対象として認識される．この問題に対して，知識推論システムである New Helic II [6], [7] では名詞タイプと動詞タイプに区別された $H$ 項という表現が提案された．更に，イベント/プロパティを区別した型階層論理 [4] では，ソート階層とは別に述語の階層をもつ論理を形式化している．しかしながら，New Helic II は項と多項述語を明確に区別した論理体系の形式化にまで至っておらず，また，後者の論理は健全性が証明されているが完全性は研究課題として残されたままである．一方，Beierle [1] はソートを単項述語として論理式に記述できる完全な推論システムを提案しているが，多項述語の階層を扱うことはできない．

本論文の目的は，ソートと多項述語の二つの階層をもつ論理を構築し，その体系の完全性を証明することである．ソートと述語の二つの階層を導入する場合，ソート論理に述語の階層表現を追加して論理を拡張すればよいが，完全な推論体系を定義するためには，次に挙げる三つの形式化の要件を満たさなければならないと考える．

- 述語の階層関係の導入によるモデルへの影響
- 引数操作の意味付け
- 完全性を保証できる推論規則と引数操作

最初の二つは，ともに意味論に関するものである．拡張する論理では，ソート階層におけるサブソート関係 $s_1 \sqsubseteq_S s_2$ と同様に，述語の階層関係 $p_1 \sqsubseteq_P p_2$ も明記できるようにするが，その解釈はサブソート関係と全く異なる．ソートの解釈は個体の集合であるのに対して，述語は複数の個体による関係である．そのうえ，述語はソートとは違って原子論理式を構成し，その解

[†] 北陸先端科学技術大学院大学情報科学研究科，石川県
School of Information Science, Japan Advanced Institute of Science and Technology, Ishikawa-ken, 923–1292 Japan

釈が論理式の真偽値を与える．したがって，述語の階層関係に対する解釈がモデルに影響を与えることを考慮する必要がある．二つ目の要件は，述語の階層関係 $p \sqsubset_P q$ により多項述語 $p$ から異なる引数構成をもつ上位述語 $q$ を導出する場合，引数のずれを解消するために引数を補充/削除する操作を論理的に意味付けすることである．そのため，意味論において解釈上の引数操作，及び（複数の述語解釈を融合させる）解釈の共通部分という概念を新たに導入し，述語の階層と引数操作による制約を満たすモデル（$H$-モデル）を定義する．最後の要件は，述語の階層上の推論手続きを実現するために，引数操作を備えた適切な推論規則を導入することである．その際に，もとの言語には含まれない定数（補完定数と呼ぶ）により言語を拡張させて，引数の操作（補充/削除）に使用し，完全な推論を実現する．以上を踏まえて，本論文では論理プログラミングの基礎的体系であるホーン節による演繹計算を定義する．

本論文の構成は，次のとおりである．**2**. では，述語の階層関係をもつ論理の構文と意味論を定義する．**3**. では，**2**. の論理に基づいたホーン節計算を設計して，その健全性と完全性を証明する．最後に，**4**. で本研究のまとめと今後の課題を述べる．

## 2. 述語階層の論理

### 2.1 構　　文

最初に，述語階層の論理の構文を定義する．

［定義 2.1］　言語 $\mathcal{L}_H$ のアルファベットは次の記号から構成される．

(1)　$S$ は，最大ソート $\top$ と最小ソート $\bot$ を含むソート記号 $s_1, s_2, \ldots$ の集合である．

(2)　$F$ は，関数記号 $f_1, f_2, \ldots$ の集合である．

(3)　$P$ は，述語記号 $p_1, p_2, \ldots$ の集合である．

(4)　$V = \bigcup_{s \in S} V_s$ は，ソート変数の集合である．ただし，$V_s$ はソート $s$ の変数 $x{:}\, s, y{:}\, s, x{:}\, s, \ldots$ の集合とする．

(5)　$AL$ は，述語引数ラベル $a_1, a_2, \ldots$ の集合である．

(6)　$\neg, \wedge, \vee, \rightarrow, \forall, \exists$ は，結合子と量化記号である．

(7)　$(, ), \Rightarrow$ は，補助記号である．

［定義 2.2］（シグネチャ）　言語 $\mathcal{L}_H$ のシグネチャは，次の条件を満たす三つ組 $\Sigma = (D_S, D_F, D_P)$ である．

(1)　$D_S$ は，サブソート宣言 $s_1 \sqsubset_S s_2$ の集合である．

(2)　$D_F$ は，関数宣言 $f{:}\, s_1, \ldots, s_n \rightarrow s (n \geqq 0)$ の集合である．

(3)　$D_P$ は，述語の階層関係 $p_1 \sqsubset_P p_2$ と述語宣言 $p{:}\, \{(a_1, s_1), \ldots, (a_n, s_n)\}$ の集合である．ただし，次の関数が，$ARG(p) = \{a_1, \ldots, a_n\}$ 及び，$1 \leqq i \leqq n$ に対して $SCP(a_i) = s_i$ を満たすものとする．

– $SCP$ は，関数 $AL \rightarrow S$ である．

– $ARG$ は，関数 $P \rightarrow 2^{AL}$ である．

$SCP(a_i)$ は，引数ラベル $a_i$ のスコープとしてあるソートを示している．また，$ARG(p)$ は，述語 $p$ の固有の引数構成を引数ラベルの有限集合により示している．

［例 2.1］　言語 $\mathcal{L}_H$ とシグネチャ $\Sigma$ の例を以下に示す．言語 $\mathcal{L}_H$ は次の記号を含むとする．

$$\begin{aligned} S &= \{person, man, woman, \top, \bot\}, \\ F &= \{john, mary\}, \\ P &= \{p_1, p_2, q\}. \end{aligned}$$

シグネチャ $\Sigma$ は次により構成される．

$$\begin{aligned} D_S = &\{man \sqsubset_S person, woman \sqsubset_S person\} \cup \\ &\{s \sqsubset_S \top | s \in S\} \cup \{\bot \sqsubset_S s | s \in S\}, \\ D_F = &\{john{:} \rightarrow man, mary{:} \rightarrow woman\}, \\ D_P = &\{q \sqsubset_P p_1, q \sqsubset_P p_2\} \cup \\ &\{p_1{:}\, \{(agt, person), (coagt, person)\}, \\ &\ p_2{:}\, \{(agt, person), (obj, \top)\}, \\ &\ q{:}\, \{(agt, person), (obj, \top)\}\ \}. \end{aligned}$$

特に $D_P$ の宣言は，通常のオーダーソート論理のシグネチャにはなく，述語の階層関係と述語の引数構成を表している．述語の階層関係 $q \sqsubset_P p_1$ は，述語 $p_1$ が述語 $q$ の上位述語であることを示している．更に，$p_1{:}\, \{(agt, person), (coagt, person)\}$ は，述語 $p_1$ の引数は，引数ラベル $agt$ と $coagt$ からそれぞれ動作主と対象者の役割をもち，そのソートはともに $person$ である．

次に言語 $\mathcal{L}_H$ の表現：項と論理式を定義する．

［定義 2.3］（項）　シグネチャ $\Sigma$ が与えられたとき，ソート $s$ の項 は次により定義される．

(1)　変数 $x{:}\, s$ は，ソート $s$ の項である．

(2)　$c \in F$ かつ $c{:} \rightarrow s \in D_F$ であるならば，定数 $c{:}\, s$ は，ソート $s$ の項である．

(3)　$t_1, \ldots, t_n$ がそれぞれソート $s_1, \ldots, s_n$ の項，

$f \in F$ かつ $f: s_1, \ldots, s_n \rightarrow s \in D_F$ であるならば，$f(t_1, \ldots, t_n): s$ は，ソート $s$ の項である．

(4) $t$ がソート $s'$ の項，$s' \sqsubset_S s \in D_S$ であるならば，$t$ はソート $s$ の項である．

以上より定義されるソート $s$ の項の全体からなる集合を $TERM_s$ で表し，すべてのソートに対する項の集合 $\bigcup_{s \in S} TERM_s$ を $TERM$ で表す．関数 $Var: TERM \rightarrow 2^V$ を，次のように定義する．(i) $Var(x: s) = \{x: s\}$，(ii) $Var(f(t_1, \ldots, t_n): s) = Var(t_1) \cup \ldots \cup Var(t_n)$(特に，$Var(c: s) = \emptyset$)．$TERM_0$ は，変数を含まないすべての項の集合を表す．つまり，$TERM_0 = \{t \in TERM | Var(t) = \emptyset\}$ である．更に，変数を含まない項 $t \in TERM_0$ のことを基礎項と呼ぶ．

［定義 2.4］（論理式） シグネチャ $\Sigma$ が与えられたとき，論理式は次により定義される．

(1) $t_1, \ldots, t_n$ がそれぞれソート $s_1, \ldots, s_n$ の項，$p \in P$，$p: \{(a_1, s_1), \ldots, (a_n, s_n)\} \in D_P$ であるならば，$p(a_1 \Rightarrow t_1, \ldots, a_n \Rightarrow t_n)$ は，原子論理式（アトム）である．

(2) $A$，$B$ が論理式ならば，$\neg A$, $A \wedge B$, $A \vee B$, $A \rightarrow B$, $\forall x: sA$，及び $\exists x: sA$ は論理式である．

以上より定義される論理式の全体からなる集合を $FORM$ で表す．

［例 2.2］ 例 2.1 の言語とシグネチャにより，論理式の例を以下に示す．

$$p_1(agt \Rightarrow john: man, coagt \Rightarrow mary: woman),$$

$$p_2(agt \Rightarrow john: man, obj \Rightarrow wallet: \top)$$

これらの論理式は，前者が「動作主 $john$ が対象者 $mary$ に行為 $p_1$ をした」ことを明言している．後者は，「動作主 $john$ が対象物 $wallet$ に行為 $p_2$ を行った」ことを記述している．

［定義 2.5］ 関数 $FVar: FORM \rightarrow 2^V$ は次により定義される．

(1) $FVar(p(a_1 \Rightarrow t_1, \ldots, a_n \Rightarrow t_n)) = Var(t_1) \cup \cdots \cup Var(t_n)$,

(2) $FVar(\neg A) = FVar(A)$,

(3) $* \in \{\vee, \wedge, \rightarrow\}$ に対して，$FVar(A * B) = FVar(A) \cup FVar(B)$.

(4) $* \in \{\forall x: s, \exists x: s\}$ に対して，$FVar(*A) = FVar(A) - \{x: s\}$,

$\bar{\mu}$ により引数ラベルと項の対からなる有限集合 $\{a_1 \Rightarrow t_1, \ldots, a_n \Rightarrow t_n\} (n > 0)$ を略記する．これにより，特に引数の詳細を明記する必要のない場合は，$p(a_1 \Rightarrow t_1, \ldots, a_n \Rightarrow t_n)$ を $p(\bar{\mu})$ で表す．更に，$\bar{\mu}$ から引数ラベルだけを取り出す関数を，$LS(\bar{\mu}) = \{a_i \in AL | a_i \Rightarrow t_i \in \bar{\mu}\}$ と定義する．

## 2.2 意 味 論

意味論の定義では，述語の引数順序によって原子論理式の解釈に違いをもたらさないようにする．例えば，論理式：

$$p(a_1 \Rightarrow t_1, a_2 \Rightarrow t_2) \text{ と } p(a_2 \Rightarrow t_2, a_1 \Rightarrow t_1)$$

は，意味的に同等であるとみなされる．

更に，本論理の特色は引数の数や引数のラベルなど構造上違いのある二つの述語 $p$，$q$ に対して $p \sqsubset_P q$ なる階層関係があるとき，$p(\cdots)$ から $q(\cdots)$ を推論できるように $p$ のもともとの引数構成を $q$ の引数構成に変更して解釈する方法を導入することである．この操作の妥当性は後に **3**. にてホーン節計算の健全性・完全性として証明する．

［定義 2.6］（構造） 構造 $M = (U, I)$ は，次の条件を満たす．

- $U$ は，空でない集合である．
- $I$ は，以下を満たす関数である．
  - $s \in S$ のとき，$I(s) \subseteq U$ である．
  - $s_i \sqsubset_S s_j \in D_S$ のとき，$I(s_i) \subseteq I(s_j)$ である．
  - $f \in F$ かつ $f: s_1, \ldots, s_n \rightarrow s \in D_F$ のとき，

$$I(f): I(s_1) \times \ldots \times I(s_n) \rightarrow I(s)$$

である．

- $p \in P$ かつ $p: \{(a_1, s_1), \ldots, (a_n, s_n)\} \in D_P$ のとき，

$$I(p) \subseteq \{\varphi \in U^{ARG(p)} | \forall a_i \in ARG(p), \varphi(a_i) \in I(SCP(a_i))\}$$

である(注1)．このような関数$\varphi$ を引数解釈と呼ぶ．

一般に一階述語論理の意味論において，対象領域 $U$ の要素からなる順序 $n$ 組の集合（すなわち，$U^n$ の部分集合）で述語を解釈するように，述語の解釈 $I(p)$ を引数解釈の集合として一意に定義する．

一つの引数解釈は，ラベルと値（$U$ の要素）の対からなる集合：

(注1)：集合 $X$，$Y$ に対して，$X$ から $Y$ の関数の全体からなる集合を $Y^X$ で表す．

$$\varphi = \{(a_1, d_1), (a_2, d_2), \cdots, (a_n, d_n)\}$$

で表記できる．このため，引数解釈はラベルの集合 $AL$ と $U$ との直積 $\langle AL, U\rangle$ の部分集合，すなわち $\varphi \subseteq \langle AL, U\rangle$ であるといえる．通常，一階述語論理では，述語は順序 $n$ 組 $(d_1, d_2, ..., d_n)$ の集合として定義されるが，本言語では順序 $n$ 組 $(d_1, d_2, ..., d_n)$ の代わりに引数解釈 $\{(a_1, d_1), (a_2, d_2), ..., (a_n, d_n)\}$ で定義され，引数ラベルがあることで順序の制限のない集合で意味付けすることができる．更に注意すべきは，引数解釈 $\varphi$ は $U^{ARG(p)}$ の元であるので関数の性質をもち，各引数ラベル $a_i$ に対して一意 $(\varphi(a_i) = d_i)$ に $U$ の元が割り当てられ，一つの順序 $n$ 組 $(d_1, d_2, ..., d_n)$ のみに対応する．更に，順序 $n$ 組の集合（すなわち，$U^n$ の部分集合）による述語解釈と同様に，解釈関数 $\varphi$ の集合（引数解釈の全体に対する部分集合）として述語の解釈 $I(p)$ が定義される．

［例 2.3］ シグネチャ $\Sigma = (\emptyset, \emptyset, \{p\colon\{(agt, person), (obj, \top)\}\})$，集合 $U = \{d_1, d_2, d_3\}$ と 関数 $I$ が与えられたとする．述語 $p$ に対して $ARG(p) = \{agt, obj\}$ とし，$SCP(agt) = person$, $SCP(obj) = \top$, $I(person) = \{d_1\}$, $I(\top) = \{d_1, d_2, d_3\}$ とする．このとき例えば，$I(p)$ が $\varphi_1 = \{(agt, d_1), (obj, d_1)\}$, $\varphi_2 = \{(agt, d_1), (obj, d_2)\}$ であるような引数解釈の集合 $\{\varphi_1, \varphi_2\}$ であるならば，$(U, I)$ は定義 2.6 の構造の条件を満たす．

次に，以上の構造 $M$ の条件に加えて，述語の階層を導入したことによる制約を満たす構造の条件を定義する．そのための準備として，シグネチャ $\Sigma = (D_S, D_F, D_P)$ で $p \sqsubset_P q \in D_P$ のとき，述語 $p$ の解釈に含まれる引数構成を変更して 述語 $q$ の解釈の引数構成に合わせる操作を考える．この操作は，$p$ と $q$ が共通にもつ引数ラベルに対しては，$p$ のそのラベルと値を生かし，$q$ にあって $p$ にはない引数ラベルに対しては，そのラベルとある値を新たに付加するという 2 段階の操作からなる．

$p, q \in P$ とし，述語 $p$ の引数解釈の集合を $I(p)$ とすると，$\varphi \in I(p)$ に対して $p, q$ の引数解釈の共通部分は，

$$\varphi \cap \langle ARG(q), U\rangle$$

である．また付加部分は，$ARG(q) - ARG(p) = \{a_{c_1}, \ldots, a_{c_n}\}$ であるようなラベルの集合に対して，ある要素 $d_{c_i} \in I(SCP(a_{c_i}))$ が割り当てられていればよいので，

$$\{(a_{c_1}, d_{c_1}), \ldots, (a_{c_n}, d_{c_n})\}$$

と書くことができる．

ここで，引数解釈 $\varphi$ から引数ラベルだけを取り出す関数を，$LS^*(\varphi) = \{a_i \in AL | (a_i, d_i) \in \varphi\}$ と定義する．

［定義 2.7］（解釈上の引数操作） シグネチャ $\Sigma = (D_S, D_F, D_P)$ と構造 $M = (U, I)$ が与えられたとする．引数解釈 $\varphi(\in I(p))$ に対する述語 $q$ への解釈上の引数操作とは，次のような引数解釈の集合への変換である．

$$\iota_q(I(p)) = \{(\varphi \cap \langle ARG(q), U\rangle) \cup \{(a_{c_1}, d_{c_1}), \ldots, (a_{c_n}, d_{c_n})\} \mid \varphi \in I(p)\}$$

ただし，$\{a_{c_1}, \ldots, a_{c_n}\} = ARG(q) - LS^*(\varphi)$（注 2）かつ $d_{c_i} \in I(SCP(a_{c_i}))$ とする．

［例 2.4］ 集合 $U = \{d_1, d_2, d_3, d_4, d', d''\}$ と関数 $I$ が与えられたとする．述語 $p, q \in P$ に対して $ARG(p) = \{agt, obj\}$, $ARG(q) = \{agt, coagt\}$ であり，引数解釈 $\varphi = \{(agt, d_1), (obj, d_2)\}$, $\psi = \{(agt, d_3), (obj, d_4)\}$ から，述語 $p$ の解釈を $I(p) = \{\varphi, \psi\}$ とすると，$\iota_q(I(p)) = \{\{(agt, d_1), (coagt, d')\}, \{(agt, d_3), (coagt, d'')\}\}$（ただし，$d', d'' \in I(SCP(coagt))$）となる．

次に複数の述語解釈を一つの述語とみなす解釈に統合する変換を定義する．複数の述語 $p_1, p_2, \cdots$ の引数解釈の共通部分とは，ラベルのセットの和 $ARG(p_1) \cup ARG(p_2) \cup \cdots$ に対して定義された引数解釈の集合である．各ラベルと値の対は $I(p_1), I(p_2), \cdots$ の要素の引数解釈から構成されるが，同一ラベルに不一致な値を与える引数解釈の組合せは排除する．

［定義 2.8］（解釈の共通部分） 構造 $M = (U, I)$ が与えられたとする．解釈の共通部分とは，以下のように二つの引数解釈の集合から一つの引数解釈の集合への変換である．

$$I(p_1) \sqcap I(p_2) = \{\varphi \oplus \psi \mid \varphi \in I(p_1),\ \psi \in I(p_2),\ \varphi \oplus \psi \neq \emptyset\}$$

ここで，二つの引数解釈の和集合から一つの引数解釈を作る操作を以下に定義する．

（注 2）：この場合，引数解釈 $\varphi$ は $I(p)$ の要素なので，定義 2.6 より $LS^*(\varphi)$ は，$ARG(p)$ と同じ集合といえる．

$$\varphi \oplus \psi = \begin{cases} \varphi \cup \psi & \mathit{if}\ \varphi(a) = \psi(a) \\ & (\forall a \in LS^*(\varphi) \cap LS^*(\psi)), \\ \emptyset & \mathit{otherwise}. \end{cases}$$

以上の定義は $n$ 個の引数解釈の集合に容易に敷衍(えん)できて，$\sqcap_{p_i \in \Pi} I(p_i)$ は各 $I(p_i)$ から一つずつ引数解釈を取り出してそれらを '$\oplus$' で結合した $\varphi_1 \oplus \varphi_2 \oplus \cdots$ の集合である．

［例 2.5］ $I(p_1) = \{\{(agt, d_1)\}\}$，$I(p_2) = \{\{(agt, d_2), (obj, d_1)\}, \{(agt, d_1), (obj, d_2)\}\}$ のとき，$I(p_1) \sqcap I(p_2) = \{\{(agt, d_1), (obj, d_2)\}\}$．

これまでの定義 2.7 と 2.8 を用いることで，述語の階層に関する構造を定義できる．まず，述語の階層関係 $\sqsubset_P$ において，隣接する関係$\sqsubset^1_P$ は，$\sqsubset^1_P := \{(p, q) \in \sqsubset_P \ |(p, r)$ or $(r, q) \notin \sqsubset_P\}$ により定義する．

［定義 2.9］ 構造 $M$ 上の述語 $q$ への解釈上の引数操作 $\iota_q$ とシグネチャ $\Sigma = (D_S, D_F, D_P)$ が与えられているとする．構造 $M = (U, I)$ が以下の条件を満たすとき，$H$-構造と呼ぶ．

(1) $p, q \in P$ かつ $p \sqsubset_P q \in D_P$ ならば，

$$\iota_q(I(p)) \subseteqq I(q)$$

である．

(2) $p_1, \ldots, p_n \in P$ は，$q \sqsubset^1_P p_i$ であるようなすべての述語とする．$q \sqsubset_P p_1, \ldots, q \sqsubset_P p_n \in D_P (n > 1)$ ならば，以下を満たす．

$$\iota_q(\sqcap_{p_i \in \Pi} I(p_i)) \subseteqq I(q)$$

ただし，$\Pi = \{p_1, \ldots, p_n\}$ とする．

［例 2.6］ 例 2.1 のシグネチャ $\Sigma$ に対して，集合 $U = \{d_1, d_2, d_3, d_4\}$ と以下を満たす関数 $I$ が与えられたとする．

$$I(p_1) = \{\varphi_1\},$$
$$I(p_2) = \{\varphi_2\},$$
$$I(q) = \{\psi\},$$

ただし，

$$\varphi_1 = \{(agt, d_1), (coagt, d_2)\},$$
$$\varphi_2 = \{(agt, d_1), (obj, d_3)\},$$
$$\psi = \{(agt, d_1), (obj, d_3)\}$$

とする．このとき，$I(p_1) \sqcap I(p_2) = \{(agt, d_1), (coagt, d_2), (obj, d_3)\}$ であり，$\iota_q(I(p_1) \sqcap I(p_2)) = \{(agt, d_1), (obj, d_3)\}$ となる．したがって，定義 2.9 (2) の条件を満たす．

構造 $M = (U, I)$ 上の変数割当てとは，すべての変数 $x{:}\,s$ に対して，$\alpha(x{:}\,s) \in I(s)$ であるような関数 $\alpha{:}\, V \to U$ である．解釈 $\mathcal{I}$ は，構造 $M$ と変数割当て $\alpha$ の対 $(M, \alpha)$ である．

［定義 2.10］ 解釈 $\mathcal{I} = (M, \alpha)$ が与えられたとすると，解釈関数 $[\![\ ]\!]_\alpha$ は以下のように定義される．

- $[\![x{:}\,s]\!]_\alpha = \alpha(x{:}\,s)$,
- $[\![c{:}\,s]\!]_\alpha = I(c)$,
- $[\![f(t_1, \ldots, t_n){:}\,s]\!]_\alpha = I(f)([\![t_1]\!]_\alpha, \ldots, [\![t_n]\!]_\alpha)$.

解釈 $\mathcal{I}$ が，$H$-構造 と変数割当てから構成されるとき，$H$-解釈という．以降では，本論文で扱う解釈をすべて $H$-解釈とする．

［定義 2.11］ $\mathcal{I} = (M, \alpha)$ を解釈とすると，充足関係 $\mathcal{I} \models A$ は，以下のように帰納的に定義される．

- $\mathcal{I} \models p(a_1 \Rightarrow t_1, \ldots, a_n \Rightarrow t_n)$ $\mathit{iff}$ $\{(a_1, [\![t_1]\!]_\alpha), \ldots, (a_n, [\![t_n]\!]_\alpha)\} \in I(p)$,
- $\mathcal{I} \models \neg A$ $\mathit{iff}$ $\mathcal{I} \not\models A$,
- $\mathcal{I} \models A \wedge B$ $\mathit{iff}$ $\mathcal{I} \models A$ かつ $\mathcal{I} \models B$,
- $\mathcal{I} \models A \vee B$ $\mathit{iff}$ $\mathcal{I} \models A$ または $\mathcal{I} \models B$,
- $\mathcal{I} \models A \to B$ $\mathit{iff}$ $\mathcal{I} \not\models A$ または $\mathcal{I} \models B$,
- $\mathcal{I} \models \forall x{:}\,sA$ $\mathit{iff}$ すべての $d \in I(s)$ に対して，$\mathcal{I}\{d/x{:}\,s\} \models A$,
- $\mathcal{I} \models \exists x{:}\,sA$ $\mathit{iff}$ ある $d \in I(s)$ に対して，$\mathcal{I}\{d/x{:}\,s\} \models A$.

論理式の集合 $\Gamma$ のすべての元 $A$ に対して，$\mathcal{I} \models A$ が成り立つとき，$\mathcal{I}$ を $\Gamma$ のモデルといい，$\mathcal{I} \models \Gamma$ で表す．特に，$\mathcal{I}$ が $H$-解釈のとき，$H$-モデルという．$\Gamma$ が少なくとも一つのモデルをもつとき，$\Gamma$ は充足可能であるといい，そうでないときは，充足不可能であるという．論理式の集合 $\Gamma$ の任意のモデルが，論理式 $A$ のモデルでもあるとき，$A$ は $\Gamma$ の論理的帰結といい，$\Gamma \models A$ と表す．特に，$\Gamma$ の元が一つのとき，$\{B\} \models A$ を $B \models A$ と表す．

## 3. ホーン節計算

本章では，前章で定義した述語階層の論理に基づいて，ホーン節計算 [2], [3] を定義し，その健全性・完全性を示す．その際，述語の階層上の導出において，異なる引数構成に対処するために補完される定数を補完定数といい，関数記号の集合 $F$ が，すべての引数ラ

ベル $a_i \in AL$ に対する補完定数 $c_{a_1}, \ldots, c_{a_n}$ を含むような言語 $\mathcal{L}_H^+$ を使用する．更に，言語 $\mathcal{L}_H^+$ のシグネチャ $\Sigma$ は，すべての引数ラベル $a_i \in AL$ に対して，$c_{a_i}\colon \to SCP(c_{a_i}) \in D_F$ を満たすものとする．

［定義 3.1］（ホーン節） $L, L_1, \ldots, L_n$ をそれぞれアトムとする．ゴール $G$ は，次のような形式である．

$$G\colon = L_1, \ldots, L_n.$$

更に，節 $C$ は，次のような形式である．

$$C\colon = L \leftarrow G.$$

$A$ が量化記号を含まない論理式であり，$FVar(A) = \{x_1\colon s_1, \ldots, x_m\colon s_m\}$ のとき，その全称閉包 $\forall x_1\colon s_1 \ldots \forall x_m\colon s_m A$ を $\forall A$ で表す．同様に，存在閉包を $\exists A$ で表す．それにより，節 $L \leftarrow G$ $(G = L_1, \ldots, L_n)$ は，その全称閉包 $\forall (L_1 \wedge \ldots \wedge L_n \rightarrow L)$ を表現しているとみなす．

［定義 3.2］（プログラム） プログラム $\mathcal{P}$ は，シグネチャ $\Sigma$ と補完定数を含まない節の有限集合 $\mathcal{C}$ の対 $(\Sigma, \mathcal{C})$ である．

［定義 3.3］（ソート代入） ソート代入は，$\theta(x\colon s) \in TERM_s$ であるような関数 $\theta\colon V \rightarrow TERM$ である．ソート代入は，$\{x\colon s \in Dom(\theta) | \theta(x\colon s) \neq x\colon s\}$ のような有限集合 $\{t_1/x_1\colon s_1, \ldots, t_n/x_n\colon s_n\}$（ただし，$1 \leqq i \leqq n$ に対して $t_i \in TERM_{s_i}$ とする）として表すことができる．

$\theta$ をソート代入とすると，すべての $x\colon s \in Dom(\theta)$ に対して $\theta(x\colon s)$ が基礎項，すなわち $Var(\theta(x\colon s)) = \emptyset$ ならば，$\theta$ をソート基礎代入という．

［定義 3.4］ $L, L_1, \ldots, L_n$ をそれぞれアトム，$G$ をゴール，$\theta$ をソート代入とする．$E\theta$ は，以下によって定義される．

- $E = x\colon s$ のとき，$E\theta = \theta(x\colon s)$，
- $E = f(t_1, \ldots, t_n)$ のとき，$E\theta = f(t_1\theta, \ldots, t_n\theta)$，
- $E = p(a_1 \Rightarrow t_1, \ldots, a_n \Rightarrow t_n)$ のとき，$E\theta = p(a_1 \Rightarrow t_1\theta, \ldots, a_n \Rightarrow t_n\theta)$，
- $E = L_1, \ldots, L_n$ のとき，$E\theta = L_1\theta, \ldots, L_n\theta$，
- $E = L \leftarrow G$ のとき，$E\theta = L\theta \leftarrow G\theta$，
- $E = \{E_1, \ldots, E_n\}$ のとき，$E\theta = \{E_1\theta, \ldots, E_n\theta\}$．

$E_1$ と $E_2$ をそれぞれ表現とすると，$E_1\theta = E_2\theta$ が成り立つとき，ソート代入 $\theta$ を $E_1$ と $E_2$ の単一化子という．

［補題 3.1］ 解釈 $\mathcal{I}$，節 $L \leftarrow G$，ソート基礎代入 $\theta$ が与えられたとする．このとき，$\mathcal{I} \models L \leftarrow G$ ならば，$\mathcal{I} \models (L \leftarrow G)\theta$ が成り立つ．

（証明）解釈 $\mathcal{I} = (M, \alpha)$，量化記号を含まない論理式 $A$ に対して，$\mathcal{I} \models \forall A$ とする．定義 2.11 により，$A$ のすべての変数割当て $\beta$ に対して，$\mathcal{I}\beta \models A$ である．一方，$\theta$ を $A$ のソート基礎代入とするならば，$\beta$ は $\beta(x\colon s) = [\![\theta(x\colon s)]\!]_\alpha$ と書ける．すると，$\mathcal{I} \models A\theta$ が成り立つ．

ここで，$\mathcal{I} \models L \leftarrow G (G = L_1, \ldots, L_n)$ のとき，$\mathcal{I} \models \forall (L_1 \wedge \ldots \wedge L_n \rightarrow L)$ である．したがって，以上の証明により $\mathcal{I} \models (L_1 \wedge \ldots \wedge L_n \rightarrow L)\theta$ が導ける． □

関数 $CVar\colon \mathcal{C} \rightarrow 2^V$ は，$CVar(L \leftarrow L_1, \ldots, L_n) = FVar(L_1 \wedge \ldots \wedge L_n \rightarrow L)$ により定義する．$\theta$ をソート代入，$C$ を節とすると，$C\theta$ を $C$ の例（instance）という．すべての $C$ の例からなる集合を $ground(C)$ で表し，$\bigcup_{c \in \mathcal{C}} ground(C)$ を $ground(\mathcal{C})$ で表す．

$\Delta(AL)$ は，$AL$ の有限部分集合の族である．$ATOM (\subset FORM)$ は，すべての原子論理式の集合であり，

$$ATOM^* = \{p(a_1 \Rightarrow t_1, \ldots, a_n \Rightarrow t_n) \mid \{a_1, \ldots, a_n\} \in \Delta(AL), p \in P, t_i \in TERM\}$$

は，未定義な引数を含んだ原子論理式の集合への拡張である．

［例 3.1］ $ARG(p) \neq \{a_1, a_2\}$ のとき，$p(a_1 \Rightarrow t_1, a_1 \Rightarrow t_2)$ は未定義な引数を含んだ論理式である．

推論規則を導入する前に，定義 2.7 と 2.8 による（述語 $p$ から述語 $q$ への）引数解釈の変換に相当する述語引数の補充/削除の手続きを定義する．

［定義 3.5］（引数操作） 原子論理式 $A$ を $ATOM^*$ の要素とすると，述語引数の操作 $\sigma$ は，以下のように定義される $ATOM^*$ から $ATOM$ への関数である．

$$\sigma(A) = ADD^m(DEL^n(A))$$

ただし，$ADD^m$ は $m$ 個の $ADD$ からなる合成関数であり，$DEL^n$ は $n$ 個の $DEL$ からなる合成関数である．このとき，$m$ は $ADD^m = ADD^{m+1} (m > 0)$ であるような最小 の自然数であり，$n$ は $DEL^n = DEL^{n+1} (n > 0)$ であるような最小の自然数である．引数の補充 $ADD$ は次のように定義される．

$$ADD(A) = \begin{cases} q(\bar{\mu} \cup \{a \Rightarrow c_a\colon s\}) & \textit{if } A = q(\bar{\mu}) \ \& \\ & \quad a \notin LS(\bar{\mu}), \\ A & \textit{otherwise.} \end{cases}$$

ただし，$a \in ARG(q) - LS(\bar{\mu})$，$SCP(a) = s$ かつ $c_a$ は引数ラベル $a$ の補完定数である．更に，引数の削除 $DEL$ は次のように定義される．

$$DEL(A) = \begin{cases} q(\bar{\mu}) & if \quad A = q(\bar{\mu} \cup \{a \Rightarrow t\}) \ \& \\ & \quad a \notin ARG(q), \\ A & otherwise, \end{cases}$$

ただし，$SCP(a) = s$ かつ $t \in TERM_s$ である．

［補題 3.2］ $\Sigma = (D_S, D_F, D_P)$ をシグネチャ，$\sigma$ を引数操作，$p_1, \ldots, p_n, p, q \in P$ とする．そのとき，次の命題が成り立つ．

(1) $p \sqsubset_P q \in D_P$ ならば，$ARG(p) = \{a_1, \ldots, a_n\}$ かつ $t_1, \ldots, t_n \in TERM_0$ による$\bar{\mu} = \{a_1 \Rightarrow t_1, \ldots, a_n \Rightarrow t_n\}$ に対して，$p(\bar{\mu}) \models \sigma(q(\bar{\mu}))$ である．

(2) $q \sqsubset_P p_1, \ldots, q \sqsubset_P p_n \in D_P (n > 1)$（ただし，$p_1, \ldots, p_n$ は，$q \sqsubset_P^1 p_i$ となるすべての述語）ならば，$\bigcup_{1 \leqq i \leqq n} ARG(p_i) = \{a_1, \ldots, a_m\}$ かつ $t_1, \ldots, t_m \in TERM_0$ による $\bar{\mu}_{p_i} = \{a_j \Rightarrow t_j | a_j \in ARG(p_i) \cap \{a_1, \ldots, a_m\}\}$ に対して，$\{p_1(\bar{\mu}_{p_1}), \ldots, p_n(\bar{\mu}_{p_n})\} \models \sigma(q(\bar{\mu}_q))$ である．

（証明）(1) $p(\bar{\mu}) \models \sigma(q(\bar{\mu}))$ であることを，定義 2.9 $\iota_q(I(p)) \subseteqq I(q)$ を用いて示す．(2) も同様に証明できる．証明の詳細は，付録 1. を参照． □

ホーン節計算の推論規則を次のとおり定義する．

［定義 3.6］（代入規則） $L \leftarrow G$ を節とする．

$$\frac{}{(L \leftarrow G)\theta}$$

ただし，$\theta$ は $L \leftarrow G$ のソート基礎代入とする．

［定義 3.7］（カット規則） $L_1 \leftarrow G_1 \cup \{L\}$，$L \leftarrow G_2$ を基礎節とする．

$$\frac{L_1 \leftarrow G_1 \cup \{L\} \quad L \leftarrow G_2}{L_1 \leftarrow G_1 \cup G_2}$$

［定義 3.8］（抽象化規則） $p(\bar{\mu}) \leftarrow G$ を基礎節，$\sigma$ を引数操作とする．

$$\frac{p(\bar{\mu}) \leftarrow G}{\sigma(q(\bar{\mu})) \leftarrow G}$$

ただし，$p \sqsubset_P q$ である．

［定義 3.9］（具体化規則） $t_1, \ldots, t_m \in TERM_0$，$\bigcup_{1 \leqq i \leqq n} ARG(p_i) = \{a_1, \ldots, a_m\}$ とする．$1 \leqq i \leqq n$ に対して $\bar{\mu}_{p_i} = \{a_j \Rightarrow t_j | a_j \in ARG(p_i) \cap \{a_1, \ldots, a_m\}\}$ からなる $p_1(\bar{\mu}_{p_1}) \leftarrow G_1, \ldots, p_n(\bar{\mu}_{p_n}) \leftarrow G_n$ を基礎節，$\sigma$ を引数操作とする．

$$\frac{p_1(\bar{\mu}_{p_1}) \leftarrow G_1 \quad \ldots \quad p_n(\bar{\mu}_{p_n}) \leftarrow G_n}{\sigma(q(\bar{\mu}_q)) \leftarrow G_1 \cup \ldots \cup G_n}$$

ただし，$1 \leqq i \leqq n (n > 1)$ に対して，$q \sqsubset_P^1 p_i$ かつ $\bar{\mu}_q = \{a \Rightarrow t \in \bigcup_{1 \leqq i \leqq n} \bar{\mu}_{p_i} | a \in ARG(q)\}$ とする．

［定理 3.1］ それぞれの推論規則の結論 $C$ は，その前提 $\{C_1, \ldots, C_n\}$ の論理的帰結である．すなわち，$\{C_1, \ldots, C_n\} \models C$ である．

（証明）それぞれの推論規則について，$\{C_1, \ldots, C_n\} \models C$ であることを示す．

1. （代入規則）：補題 3.1 によって，容易に導ける．

2. （カット規則）：基礎節 $L_1 \leftarrow G_1 \cup \{L\}$，$L \leftarrow G_2$ に対して，$\mathcal{I} \models L_1 \leftarrow G_1 \cup \{L\}$ かつ $\mathcal{I} \models L \leftarrow G_2$ とする．$L_{G_i} = \wedge_{L_i \in G_i} L_i$ のとき，$\mathcal{I} \models L_{G_1} \wedge L \rightarrow L_1$ かつ $\mathcal{I} \models L_{G_2} \rightarrow L$ である．$\mathcal{I} \models L_{G_1} \wedge L$ ならば，$\mathcal{I} \models L_1$ が成り立つ．$\mathcal{I} \not\models L_{G_1} \wedge L$ ならば，$\mathcal{I} \not\models L_{G_1}$ または $\mathcal{I} \not\models L_{G_2}$ である．よって，$\mathcal{I} \not\models L_{G_1} \wedge L_{G_2}$．したがって，$\mathcal{I} \models L_{G_1} \wedge L_{G_2} \rightarrow L_1$ である．

3. （抽象化規則）：$\mathcal{I} \models G$ のとき，$\mathcal{I} \models p(\bar{\mu})$ であるので，補題 3.2 より $\mathcal{I} \models \sigma(q(\bar{\mu}))$ である．したがって，$\mathcal{I} \models \sigma(q(\bar{\mu})) \leftarrow G$．

4. （具体化規則）：$\mathcal{I} \models L_{G_1} \wedge \ldots \wedge L_{G_n}$とすると，$\mathcal{I} \models p_1(\bar{\mu}_{p_1})$，$\ldots$，$\mathcal{I} \models p_n(\bar{\mu}_{p_n})$． よって，補題 3.2 から，$\mathcal{I} \models \sigma(q(\bar{\mu}_q))$ となる．ゆえに，$\mathcal{I} \models L_{G_1} \wedge \ldots \wedge L_{G_n} \rightarrow \sigma(q(\bar{\mu}_q))$ である．したがって，$\mathcal{I} \models \sigma(q(\bar{\mu}_q)) \leftarrow G_1 \cup \ldots \cup G_n$ が成り立つ． □

［定義 3.10］（導出） $\mathcal{P} = (\Sigma, \mathcal{C})$ をプログラム，$C, C', C_1, C_2$ をそれぞれ節，$\theta$ をソート基礎代入とすると，導出 $\mathcal{P} \vdash C$ は次のように定義される．

(1) $C \in \mathcal{C}$ ならば，（代入規則）の結論 $C\theta$ に対して，$\mathcal{P} \vdash C\theta$ である．

(2) （カット規則）の前提 $C_1$，$C_2$ に対して，$\mathcal{P} \vdash C_1$ かつ $\mathcal{P} \vdash C_2$ ならば，その規則の結論 $C'$ に対して，$\mathcal{P} \vdash C'$ である．

(3) （抽象化規則）または（具体化規則）の前提 $C$ に対して，$\mathcal{P} \vdash C$ ならば，その規則の結論 $C'$ に対して，$\mathcal{P} \vdash C'$ である．

$\mathcal{P} = (\Sigma, \mathcal{C})$ をプログラムとすると，$\mathcal{P} \vdash L \leftarrow G$ のとき，$L \leftarrow G$ は $\mathcal{P}$ より導出可能であるという．特に，$\mathcal{P} \vdash L \leftarrow$ を $\mathcal{P} \vdash L$ で表す．$\Sigma$ が与えられ，かつ $\mathcal{C} \models C$ ならば，$\mathcal{P} \models C$ と表す．

［定理 3.2］（導出の健全性） $\mathcal{P}$ をプログラム，$L$ をアトムとすると，$\mathcal{P} \vdash L$ ならば，$\mathcal{P} \models L$ である．

（証明）定義 3.10 と定理 3.1 から，導出の長さに関する帰納法で証明できる． □

［定義 3.11］ エルブラン構造 $M_H = (I_H, U_H)$ は，次

の条件を満たす構造である．

(1)　$U_H = TERM_0$ である．

(2)　$c \in F$ かつ $c\!: \rightarrow s \in D_F$ のとき，$I_H(c) = c$ である．

(3)　$f \in F$ かつ $f\!: s_1, \ldots, s_n \rightarrow s \in D_F$ のとき，$I_H(f)(t_1, \ldots, t_n) = f(t_1 \ldots, t_n)$ である．

エルブラン解釈 $\mathcal{I}_H$ は，その構造 $M_H$ がエルブラン構造である解釈をいう．

［補題 3.3］ $\mathcal{I}_H$ をエルブラン解釈，$L \leftarrow G$ を節とすると，$\mathcal{I}_H \models L \leftarrow G$ ならば，かつそのときに限り $\mathcal{I}_H \models ground(L \leftarrow G)$ である．

（証明）

(⇒) $\mathcal{I}_H \models L \leftarrow G$ とする．（代入規則）と補題 3.1 により，$L \leftarrow G$ のすべての例に対して，$(L \leftarrow G)\theta \in ground(L \leftarrow G)$ となる．

(⇐) $CVar(L \leftarrow G) = \{x_1\!: s_1, \ldots, x_n\!: s_n\}$ のとき，$\mathcal{I}_H \models ground(L \leftarrow G)$ を仮定する．$1 \leqq i \leqq n$ に対して，任意の $d_i \in I(s_i)$ で，$\mathcal{I}_H\{d_1/x_1\!: s_1, \ldots, d_n/x_n\!: s_n\} \models L \leftarrow G$ となることを証明する．定義 3.11 より，$d_i \in I(s_i) = TREM_{s_i}$ なので，$L \leftarrow G$ への任意のソート基礎代入 $\theta$ に対して，$(L \leftarrow G)\theta \in ground(L \leftarrow G)$ である．したがって，$(x_i\!: s_i)\theta = \alpha(x_i\!: s_i)$ と定義すると，$\mathcal{I}_H \models L \leftarrow G$ となる．　□

続いて，完全性の証明を与える．まず，プログラム $\mathcal{P}$ から導出可能なアトムすべてを充足する解釈 $\mathcal{I}_\mathcal{P}$ を構築し，その解釈がプログラム $\mathcal{P}$ のモデルであることを示す．それにより，任意のプログラムに対して解釈 $\mathcal{I}_\mathcal{P}$ が与えられ，そこで充足するアトムは，すべて導出可能であることから完全性を証明する．

［定義 3.12］ $\mathcal{P}$ をプログラム，$L$ をアトムとする．導出解釈 $\mathcal{I}_\mathcal{P}$ は，以下より構成されるエルブラン解釈である．

$$\mathcal{I}_\mathcal{P} \models L \ \mathit{iff}\ \mathcal{P} \vdash L.$$

［補題 3.4］ $\mathcal{P}$ をプログラムとすると，導出解釈 $\mathcal{I}_\mathcal{P}$ は $H$-モデルである．

（証明）付録 2. を参照．　□

［定理 3.3］（導出の完全性） $\mathcal{P}$ をプログラム，$L$ をアトム，$\theta$ を $L$ のソート基礎代入とすると，$\mathcal{P} \models L\theta$ ならば，$\mathcal{P} \vdash L\theta$ である．

（証明）$\mathcal{I}$ を解釈，$\mathcal{P} = (\Sigma, \mathcal{C})$ をプログラムとする．このとき，$\mathcal{P} \models L\theta$ （すなわち，$\Sigma$ が与えられて，$\mathcal{C} \models L\theta$）を仮定する．補題 3.4 によって，$\mathcal{I}_\mathcal{P}$ は $\mathcal{P}$ の $H$-モデル，つまり $\mathcal{I}_\mathcal{P} \models L\theta$ である．ゆえに，$\mathcal{I}_\mathcal{P}$ の定義から $\mathcal{P} \vdash L\theta$ が導かれる．　□

## 4. む　す　び

本論文では，ホーン節計算によるソートと述語の二つの階層を導入した完全な論理体系を提案した．階層表現を備えた論理は，オーダーソート論理に代表されるように，ソート表現による階層を備えている．対して，本論文の新規性は異なる引数構成をもつ多項述語による階層をもつ論理に基づいて，その演繹計算を設計し完全性を証明したことにある．特に，述語の階層に関する導出には，包含関係によって疑似的に表した階層関係とは違って，述語間の引数のずれを調整する操作を実現した．述語階層を伴う論理は，もともと自然言語の意味論や法的推論など多様かつ複雑な知識表現を必要とする分野からの要求仕様であるが，本研究で示した健全性・完全性により，この拡張された論理が知識表現に対しても有用な基盤を与えることが期待される．

## 文　　献

[1] C. Beierle, U. Hedtsuck, U. Pletat, P.H. Schmitt, and J. Siekmann, “An order-sorted logic for knowledge representation systems,” Artif. Intell., vol.55, pp.149–191, 1992.

[2] M. Hanus, “Logic programming with type specifications,” in Types in Logic Programming, ed. F. Pfenning, MIT Press, 1992.

[3] W. Hodges, “Logical feature of horn clauses,” in Handbook of Logic in Artificial Intelligence and Logic Programming vol.1, ed. D.M. Gabby, C.J. Hogger, and J.A. Robinson, Oxford University Press, 1992.

[4] K. Kaneiwa and S. Tojo, “Event, property, and hierarchy in order-sorted logic,” Proc. 1999 Int. Conf. on Logic Programming, pp.94–108, MIT Press, 1999.

[5] M. Manzano, “Introduction to many-sorted logic,” in Many-sorted Logic and its Applications, pp.3–86, John Wiley and Sons, 1993.

[6] K. Nitta, M. Shibasaki, T. Sakata, T. Yamaji, W. Xianchang, H. Ohsaki, S. Tojo, and I. Kokubo, “New helic II: A software tool for legal reasoning,” Proc. 5th Int. Conft. on Artificial Intelligence and Law, pp.287–296, ACM Press, 1995.

[7] K. Nitta, M. Shibasaki, T. Sakata, T. Yamaji, H. Ohsaki, S. Tojo, I. Kokubo, and T. Suzuki, “Knowledge representation of new helic II,” Workshop on Legal Application of Logic Programming, ICLP ’94, pp. 90–104, 1994.

[8] A. Oberschelp, “Untersuchungen zur mehrsortigen quantorelogik,” Mathematische Annalen 145, pp.297–333, 1962.

[9] M. Schmidt-Schauss, Computational Aspects of an Order-Sorted Logic with Term Declarations, Springer-Verlag, 1989.
[10] C. Walther, A Many-Sorted Calculus Based on Resolution and Paramodulation, Pitman and Kaufman Publishers, 1987.
[11] C. Walther, "Many-sorted unification," J. Association for Computing Machinery, vol.35, no.1, pp.1–17, 1988.

# 付　　録

## 1. 補題 3.2 の証明

(1)　解釈 $\mathcal{I}=(M,\alpha)$ が与えられて，$\mathcal{I}\models p(a_1\Rightarrow t_1,\dots,a_n\Rightarrow t_n)$ とする．すなわち，$\{(a_1,[\![t_1]\!]_\alpha),\dots,(a_n,[\![t_n]\!]_\alpha)\}\in I(p)$ である．一方，定義 3.5 により以下が成り立つ．

$$\sigma(q(\bar{\mu}))=q(b_1\Rightarrow t_{b1},\dots,b_m\Rightarrow t_{bm},\\ e_1\Rightarrow c_{e_1}\!:\!SCP(e_1),\dots,e_k\Rightarrow c_{e_k}\!:\!SCP(e_k))$$

ただし，$\{b_1,\dots,b_m\}=ARG(p)\cap ARG(q)$，$1\leqq i\leqq m$ に対して，$b_i=a_j$ のとき，$t_{bi}=t_j$ であり，$\{e_1,\dots,e_k\}=ARG(q)-ARG(p)$ である．一方，$\iota_q(I(p))$ を以下のように定義する．

$$\iota_q(I(p))=\{(\varphi\cap\langle ARG(q),U\rangle)\cup\\ \{(e_1,d_1),\dots,(e_k,d_k)\}|\varphi\in I(p)\}$$

ただし，$1\leqq i\leqq k$ に対して，$d_i=[\![c_{e_i}\!:\!SCP(e_i)]\!]_\alpha$ である．

定義 2.7 によって，$\{(b_1,[\![t_{b1}]\!]_\alpha),\dots,(b_m,[\![t_{bm}]\!]_\alpha),(e_1,d_1),\dots,(e_k,d_k)\}\in\iota_q(I(p))\subseteqq I(q)$．したがって，$\mathcal{I}\models\sigma(q(\bar{\mu}))$ となる．

(2)　解釈 $\mathcal{I}=(M,\alpha)$ が与えられたとき，$1\leqq i\leqq n$ に対して $\mathcal{I}\models p_i(\bar{\mu}_{p_i})$ とする．つまり，$\{(a_j,[\![t_j]\!]_\alpha)|a_j\in ARG(p_i)\}\in I(p_i)$ である．更に，$\Pi=\{p_1,\dots,p_n\}$ かつ $\{a_1,\dots,a_m\}=\bigcup_{p\in\Pi}ARG(p)$ とする．定義 3.5 により，

$$\sigma(q(\bar{\mu}_q))=q(b_1\Rightarrow t_{b1},\dots,b_u\Rightarrow t_{bu},\\ e_1\Rightarrow c_{e_1}\!:\!SCP(e_1),\dots,e_k\Rightarrow c_{e_k}\!:\!SCP(e_k))$$

ただし，$\{b_1,\dots,b_u\}=\{a_1,\dots,a_m\}\cap ARG(q)$ であり，$1\leqq l\leqq u$ に対して，$b_l=a_j$ のとき $t_{bl}=t_j$ であり，$\{e_1,\dots,e_k\}=ARG(q)-\{a_1,\dots,a_m\}$ である．定義 2.8 から，$\sqcap_{p_i\in\Pi}I(p_i)$ は，$\{(a_1,[\![t_1]\!]_\alpha),\dots,(a_m,[\![t_m]\!]_\alpha)\}$ を含む．

続いて，$\iota_q(\sqcap_{p_i\in\Pi}I(p_i))$ を以下のように定義する．

$$\iota_q(\sqcap_{p_i\in\Pi}I(p_i))=\{(\varphi\cap\langle ARG(q),U\rangle)\cup\\ \{(e_1,d_1),\dots,(e_k,d_k)\}|\varphi\in\sqcap_{p_i\in\Pi}I(p_i)\}$$

ただし，$1\leqq i\leqq k$ に対して，$d_i=[\![c_{e_i}\!:\!SCP(e_i)]\!]_\alpha$ である．

定義 2.7 により，$\{(b_1,[\![t_{b1}]\!]_\alpha,\dots,(b_u,[\![t_{bu}]\!]_\alpha),(e_1,d_1),\dots,(e_k,d_k)\}\in\iota_q(\sqcap_{p_i\in\Pi}I(p_i))\subseteqq I(q)$ である．したがって，$\mathcal{I}\models\sigma(q(\bar{\mu}_q))$ が証明できる．

## 2. 補題 3.4 の証明

$\mathcal{I}_\mathcal{P}$ が $\mathcal{P}$ のモデルであることを証明するためには，$\mathcal{P}=(\Sigma,\mathcal{C})$ に含まれるすべての節 $L\leftarrow G$ に対して，$\mathcal{I}_\mathcal{P}\models L\leftarrow G$ を示せばよい．$\mathcal{I}_\mathcal{P}$ は，エルブラン 解釈なので，補題 3.3 より，$\mathcal{I}_\mathcal{P}\models L\leftarrow G\Leftrightarrow\mathcal{I}_\mathcal{P}\models ground(L\leftarrow G)\Leftrightarrow L\leftarrow G$ のすべてのソート基礎代入 $\theta$ に対して $\mathcal{I}_\mathcal{P}\models(L\leftarrow G)\theta$ である．

$(L\leftarrow G)\theta$ を節 $L\leftarrow G(G=L_1,\dots,L_n)\in\mathcal{C}$ の例とする．$\mathcal{I}_\mathcal{P}\models L_1\theta,\dots,\mathcal{I}_\mathcal{P}\models L_n\theta$ を仮定すると，$\mathcal{I}_\mathcal{P}$ の定義より，$\mathcal{P}\vdash L_i\theta$ である．よって，(代入規則) により，$\mathcal{P}\vdash L_1\theta\wedge\dots\wedge L_n\theta\rightarrow L\theta$ であり，(カット規則) より $\mathcal{P}\vdash L\theta$ がいえる．したがって，$\mathcal{I}_\mathcal{P}$ の定義から，$\mathcal{I}_\mathcal{P}\models L\theta$ となる．

次に，$\mathcal{I}_\mathcal{P}$ が $H$-解釈であることを示す．$p\sqsubset q\in D_P$ のとき，以下を仮定する．

$$(\varphi\cap\langle ARG(q),U\rangle)\cup\{(e_1,d_1),\dots,(e_n,d_n)\}\\ \in\iota_q(I(p))$$

ただし，$1\leqq i\leqq k$ に対して，$d_i=[\![c_{e_i}\!:\!SCP(e_i)]\!]_\alpha$ であり，$\varphi=\{(a_1,[\![t_1]\!]_\alpha),\dots,(a_n,[\![t_n]\!]_\alpha)\}$ $(\in I(p))$ である．定義 2.11 より，$\mathcal{I}_\mathcal{P}\models p(\bar{\mu})$ (ただし，$\bar{\mu}=a_1\Rightarrow t_1,\dots,a_n\Rightarrow t_n$) である．$\mathcal{I}_\mathcal{P}$ の定義から，$\mathcal{P}\vdash p(\bar{\mu})$ を導けるので，(抽象化規則) を使って $\mathcal{P}\vdash\sigma(q(\bar{\mu}))$ となる．ゆえに，$\mathcal{I}_\mathcal{P}$ の定義から，$\mathcal{P}\models\sigma(q(\bar{\mu}))$ が成り立つ．一方，定義 3.5 から以下が成り立つ．

$$\sigma(q(\bar{\mu}))=q(b_1\Rightarrow t_{b1},\dots,b_m\Rightarrow t_{bm},\\ e_1\Rightarrow c_{e_1}\!:\!SCP(e_1),\dots,e_k\Rightarrow c_{e_k}\!:\!SCP(e_k))$$

ただし，$1\leqq i\leqq m$ に対して，$\{b_1,\dots,b_m\}=ARG(p)\cap ARG(q)$ であり，$b_i=a_j$ のとき，$t_{bi}=t_j$ であり，$\{e_1,\dots,e_k\}=ARG(q)-ARG(p)$ である．ゆえに，$(I(p)\cap\langle ARG(q),U\rangle)\cup\{(e_1,[\![c_{e_1}\!:\!SCP(e_1)]\!]_\alpha),\dots,(e_n,[\![c_{e_n}\!:\!SCP(e_n)]\!]_\alpha)\}\in I(q)$ である．

更に，$p_1, \ldots, p_n$ が $q \sqsubset_P^1 p_i$ であるようなすべての述語のとき，$q \sqsubset p_1, \ldots, q \sqsubset p_n \in D_P (n > 1)$ に対して，以下を仮定する．

$$(\varphi \cap \langle ARG(q), U\rangle) \cup \{(e_1, d_1), \ldots, (e_n, d_n)\} \\ \in \iota_q(\sqcap_{p_i \in \Pi} I(p_i))$$

ただし，$1 \leqq i \leqq k$ に対して，$d_i = [\![c_{e_i} \!:\! SCP(e_i)]\!]_\alpha$ であり，$\Pi = \{p_1, \ldots, p_n\}$ かつ $\{a_1, \ldots, a_m\} = \bigcup_{p \in \Pi} ARG(p)$ により，$\varphi = \{(a_1, [\![t_1]\!]_\alpha), \ldots, (a_m, [\![t_n]\!]_\alpha)\}$ $(\in \sqcap_{p_i \in \Pi} I(p_i))$（すなわち，$\{(a_j, [\![t_j]\!]_\alpha) | a_j \in ARG(p_i)\} \in I(p_i)$ である）．定義 2.11 により，$1 \leqq i \leqq n$ に対して，$\mathcal{I_P} \models p_i(\bar{\mu}_{p_i})$ である．ただし，$\bar{\mu}_{p_i} = \{(a_j, [\![t_j]\!]_\alpha) | a_j \in ARG(p_i)\}$．$\mathcal{I_P}$ の定義によって，$1 \leqq i \leqq n$ に対して $\mathcal{P} \vdash p_i(\bar{\mu}_{p_i})$ を導ける，よって，$\mathcal{P} \vdash \sigma(q(\bar{\mu}_q))$（ただし，$\bar{\mu}_q = \{a \Rightarrow t \in \bigcup_{1 \leqq i \leqq n} \bar{\mu}_{p_i} | a \in ARG(q)\}$）が,（抽象化規則）により導かれ，$\mathcal{P} \models \sigma(q(\bar{\mu}_q))$ が $\mathcal{I_P}$ の定義より成り立つ．また，定義 3.5 から以下が成り立つ．

$$\sigma(q(\bar{\mu}_q)) = q(b_1 \Rightarrow t_{b1}, \ldots, b_u \Rightarrow t_{bu}, \\ e_1 \Rightarrow c_{e_1} \!:\! SCP(e_1), \ldots, e_k \Rightarrow c_{e_k} \!:\! SCP(e_k))$$

ただし，$\{b_1, \ldots, b_u\} = \{a_1, \ldots, a_m\} \cap ARG(q)$ であり，$1 \leqq l \leqq u$ に対して，$b_l = a_j$ のとき，$t_{bl} = t_j$ であり，$\{e_1, \ldots, e_k\} = ARG(q) - \{a_1, \ldots, a_m\}$ である．

ゆえに，

$$(I(p) \cap \langle ARG(q), U\rangle) \cup \{(e_1, [\![c_{e_1} \!:\! SCP(e_1)]\!]_\alpha), \\ \ldots, (e_n, [\![c_{e_n} \!:\! SCP(e_n)]\!]_\alpha)\} \in I(q).$$

（平成 12 年 3 月 31 日受付，6 月 19 日再受付）

**兼岩　　憲**　（学生員）

1993〜1996 富士通（株）勤務．1998 北陸先端科学技術大学院大学情報科学研究科修士課程了．現在，同大学院情報科学研究科博士後期課程在学中．論理プログラミング，ソート論理に関する研究に従事．情報処理学会，人工知能学会，ソフトウェア科学会，ALP 各会員．

**東条　　敏**

1981 東大・工・計数卒．1983 同大大学院工学系研究科了．同年三菱総合研究所入社．1986〜1988 米国カーネギー・メロン大学機械翻訳センター客員研究員．1995 北陸先端科学技術大学院大学情報科学研究科助教授．2000 同教授．1997〜1998 ドイツ・シュトゥットガルト大学客員研究員．博士（工学）．自然言語の形式意味論，オーダーソート論理，法的推論，マルチエージェントの研究に従事，その他人工知能一般に興味をもつ．情報処理学会，人工知能学会，ソフトウェア科学会，言語処理学会，認知科学会，ACL, Folli 各会員．