

# 動的プリファレンスを扱う議論エージェントシステムの構築

関口 知之<sup>1</sup>若木 利子<sup>2</sup>

芝浦工業大学 システム工学部\*

## 1 はじめに

近年、コンピュータ上のエージェントに議論を行わせることを目的として、議論の不動点意味論や対話的証明論などが提案されている [3, 4]。ところで法廷における論争等では、使用する法令間のコンフリクトに起因して議題の判定ができない場合、新法優先等の法令間のプリファレンス知識、或いはそれらの間のプリファレンス知識 (メタプリファレンス) を用いて判定されることが多い。このような種々のメタレベルのプリファレンスを用いた議論の形式化について、Brewka[1] や Prakken[3] らは、オブジェクトレベルの知識表現を拡張して同一知識ベースにメタ知識のプリファレンスを表現していた。最近、S. Modgil[2] はオブジェクトレベルと種々のメタレベルのプリファレンス等を表現できる階層的知識ベースを用意し、対応するレベルのプリファレンス知識は対応するレベルの知識ベースに格納して議論計算をする階層的議論の方法を提案した。Modgilの方法ではルールの強さを定量的に与え、それを用いて論証間のプリファレンス (優先関係) を算出している。本研究では Modgilの方法と類似の階層的議論ではあるが Modgilのようにルールに強さを付与せず、単にルール集合上の優先関係から論証間優先関係を導いて議論計算を行う方法を提案する。さらに提案手法に基づく議論システムを実装し、種々の例で検証を行った。

## 2 議論フレームワーク

本研究の議論フレームワークの定義と意味論を示す。  
**定義 1 (拡張論理プログラム)**[4] 拡張論理プログラム (ELP) は以下の形式のルールの集合である。

$$L_0 \leftarrow L_1, \dots, L_m, \text{not } L_{m+1}, \dots, \text{not } L_{m+n}$$

但し  $m, n \geq 0$ ,  $L_i$  はリテラル (原子式  $A$ , またはその論理否定  $\neg A$ ),  $\text{not}$  は失敗による否定 (NAF) である。  $\text{not } L_i$  を NAF リテラルと称する。  $\leftarrow$  に対して左辺を頭部, 右辺を本体と呼び、それぞれ  $\text{head}(r)$ ,  $\text{body}(r)$  で表す。特に  $\{L_1, \dots, L_m\}$  を  $\text{body}(r)^+$  で、  $\{\text{not } L_{m+1}, \dots, \text{not } L_{m+n}\}$  を  $\text{body}(r)^-$  で表す。

**定義 2 (論証と攻撃関係)**[4]  $P$  のルール  $r_i \in P$  の有限系列  $Ag = [r_1; \dots; r_n]$  について、  $\forall L \in \text{body}(r_i)$  ( $1 \leq i \leq n$ ) に対して  $\text{head}(r_k) = L$  なる  $k > i$  が存在する  $Ag$  を ELP  $P$  の論証という。  $Ag$  の各ルール  $r$  について、  $\text{head}(r)$  を  $Ag$  の結論、  $\text{not } L \in \text{body}(r)^-$  を  $Ag$  の仮定と称する。  $Ag$  の全ての結論と全ての仮定を  $\text{conc}(Ag)$ ,  $\text{assm}(Ag)$  で、  $P$  の全ての論証の集合を  $\text{Arg}_{SP}$  で表す。

攻撃関係  $R$  は  $\text{Arg}_{SP}$  上の関係 ( $R \subseteq \text{Arg}_{SP}^2$ ) である。  $\text{rebut}$ ,  $\text{undercut}$ ,  $\text{attack}$ ,  $\text{defeat}$  (略して  $r, u, a, d$ ) 等がある。  $r, u, a$  は以下で定義される (詳細は [4] 参照)。

(i)  $(Ag_1, Ag_2) \in r$  if  $\exists L$  s. t.  $L \in \text{conc}(Ag_1) \wedge \neg L \in \text{conc}(Ag_2)$ . (ii)  $(Ag_1, Ag_2) \in u$  if  $\exists L$  s. t.  $L \in \text{conc}(Ag_1) \wedge \text{not } L \in \text{assm}(Ag_2)$ . (iii)  $a = r \cup u$ .

Argument-based agent systems which reason dynamic preferences

<sup>1</sup>Tomoyuki Sekiguchi, <sup>2</sup>Toshiko Wakaki

\*Shibaura Institute of Technology

**定義 3 (ルール間優先関係)** ELP  $P$  について、(静的) ルール間優先関係  $\leq$  は  $P$  上の関係 (i.e.  $\leq \subseteq P^2$ ) である。  $r \leq r'$  (i.e.  $(r, r') \in \leq$ ) は “ルール  $r$  よりルール  $r'$  を優先する” ことを意味する。  $\leq$  は pre-order である。他方、  $P$  上の関係  $<$  は、  $r < r' \stackrel{\text{def}}{=} r \leq r' \wedge r' \not\leq r$  で定義される。

**定義 4 (論証間優先関係)** ELP  $P$  と  $P$  上の関係  $\leq$  が与えられた時、論証間優先関係  $\sqsubseteq$  は  $\text{Arg}_{SP}$  上の関係 (i.e.  $\sqsubseteq \subseteq \text{Arg}_{SP}^2$ ) として以下で定義される。

$Ag_1 \sqsubseteq Ag_2$  iff 「 $r_1 \leq r_2$  なる  $r_1 \in Ag_1, r_2 \in Ag_2$  が存在し、  $r_2 < r_3$  なる  $r_3 \in Ag_1$  が存在しない。」

この時、  $Ag_1 \sqsubseteq Ag_2$  を “論証  $Ag_1$  より論証  $Ag_2$  を優先する” という。  $\sqsubseteq$  は pre-order である。

**定義 5 (議論フレームワーク)** ELP  $P$  と  $P$  上のルール間優先関係  $\leq$  から構成される議論フレームワーク  $AF$  を、  $AF \stackrel{\text{def}}{=} (\text{Arg}_{SP}, R, \sqsubseteq)$  で定義する。

以下の議論フレームワーク  $AF$  の意味論は Prakkenらの不動点意味論 [3, 4] を拡張したものである。

**定義 6 (プリファレンス付き攻撃関係と defend)**

議論フレームワーク  $AF = (\text{Arg}_{SP}, R, \sqsubseteq)$  において、  $D \stackrel{\text{def}}{=} R - \sqsubseteq$  と定義する。この時、  $(B, A) \in R$  か  $(B, A) \notin \sqsubseteq$  の時、そしてその時に限り  $(B, A) \in D$  であり、 “ $B$  は  $A$  をプリファレンス付き攻撃をする” という。

論証集合  $S \subseteq \text{Arg}_{SP}$  と論証  $A \in \text{Arg}_{SP}$  について、 「 $(B, A) \in D$  なる任意の論証  $B \in \text{Arg}_{SP}$  に対して、  $(C, B) \in D$  なる論証  $C \in S$  が存在する」 ならば、 “ $S$  は  $A$  を defend する” と称する。

**定義 7 (AF の不動点意味論)** 所与の  $AF$  より、単調関数  $F_P : \mathcal{P}(\text{Arg}_{SP}) \rightarrow \mathcal{P}(\text{Arg}_{SP})$  を以下で定義する。

$$F_P(S) = \{A \mid S \text{ は } A \text{ を defend する}\}$$

$F_P$  の最小不動点を  $J_P$  で表す。  $A \in J_P$  ならば、論証  $A$  は justified (正当化) されたという。正当化とは議論での勝利を意味する。

## 3 動的プリファレンスと階層的議論

本研究の提案手法である動的プリファレンスを扱う階層的議論の方法を以下に述べる。

**定義 8 (階層的知識ベースと知識表現)** 階層  $i$  ( $0 \leq i \leq k$ ) の知識ベースを ELP  $P^i$  で表し、階層的知識ベースを  $HKB \stackrel{\text{def}}{=} \langle P^1, \dots, P^k \rangle$  で表す。ルール  $r \in P^i$  のルール名を  $n_r$ ,  $P^i$  のルールの名前の集合を  $N^i$  とする。  $P^i$  上のルール間優先関係を  $\leq_i$ ,  $N^i$  上のルール名間優先関係を  $\leq_i$  とすると、  $r \leq_i r'$  iff  $n_r \leq_i n_{r'}$  である。

本研究では、  $\leq_i$  を述語記号  $\preceq$  で表す。  $n_r \preceq n_{r'}$  をプリファレンスアトムと称し、  $r, r' \in P^i$  の時、  $n_r \preceq n_{r'}$  は  $P^{i+1}$  のルールにリテラルとして出現可能とする。

**定義 9 (rebut の拡張)**  $\text{Arg}_{SP^i}$  は ELP  $P^i$  から構成される論証集合とする。  $\text{rebut}$  ( $r$  と略記) を以下のように拡張する。  $Ag_1, Ag_2 \in \text{Arg}_{SP^i}$  について、  $(Ag_1, Ag_2) \in r$  if  $n_r \preceq_i n_{r'} \in \text{conc}(Ag_1) \wedge n_{r'} \preceq_i n_r \in \text{conc}(Ag_2)$ .

次に階層的知識ベースの下での議論計算手続きを示す。ここでは  $J_{P^i}$  を  $J_i$  で略記するものとする。

手続き  $Harg(P^1, \dots, P^k; J_1)$

入力:  $HKB = \langle P^1, \dots, P^k \rangle$ , 返値: 最小不動点  $J_1$

初期設定:  $J_{k+1} := \emptyset, i := k$ .

step1: ELP  $P^i$  から  $Args_{P^i}, R_i$  を計算.

step2: 階層  $i+1$  の最小不動点  $J_{i+1}$  より, 階層  $i$  のルール間優先関係  $\leq_i$  を次式に従って求める.

$$\leq_i = \{(r, r') \mid n_r \leq n_{r'} \in conc(A) \text{ for } A \in J_{i+1}\}$$

step3: 定義 4 に従って,  $Args_{P^i}$  と  $\leq_i$  から階層  $i$  の論証間優先関係  $\sqsubseteq_i$  を求める.

step4: 議論フレームワーク  $AF_i = (Args_{P^i}, R_i, \sqsubseteq_i)$  より,  $D_i = R_i - \sqsubseteq_i$  を用いて,  $AF_i$  の最小不動点  $J_i$  を定義 7 に基づいて求める.

step5:  $i = 1$  ならば return  $J_1$ .

そうでなければ  $i := i - 1$ . go to step1.

例 1 “船の担保権に関する問題”[1, 3] を以下に示す.

Assume a person wants to find out if her security interest in a certain ship is perfected. She currently has possession of the ship. According to the Uniform Commercial Cod (UCC) which is a state law, a security interest in goods may be perfected by taking possession of the collateral. However, there is a federal law called the Ship Mortgage Act (SMA) according to which a security interest in a ship may only be perfected by filing a financing statement. Such a statement has not been filed. Now the question is whether the UCC or the SMA takes precedence in this case. The principle of *Lex Posterior (LP)* gives precedence newer laws. In our case, UCC is newer than the SMA. On the other hand, the principle of *Lex Superior (LS)* gives precedence to laws supported by the higher authority. In our case the SMA has higher authority since it is federal law. In addition, we have information that LS has higher precedence than LP.

上記の最初の Roman 体の部分がオブジェクトレベル, *Italic* 体がメタレベル, *Italic* 体の次の Roman 体がメタメタレベルの知識であるので, 3階層の知識ベース  $HKB = \langle P^1, P^2, P^3 \rangle$  を構成する. 但し,

$$P^1: perf \leftarrow poss, not \neg perf. \quad (ucc)$$

$$\neg perf \leftarrow ship, \neg finstat, not perf. \quad (sma)$$

$$poss \leftarrow ship \leftarrow \neg finstat \leftarrow .$$

$$P^2: d_2 \leq d_1 \leftarrow moreRecent(d_1, d_2). \quad (lp(d_1, d_2))$$

$$d_2 \leq d_1 \leftarrow fedLaw(d_1), stLaw(d_2). \quad (ls(d_1, d_2))$$

$$moreRecent(ucc, sma) \leftarrow .$$

$$fedLaw(sma) \leftarrow . \quad stLaw(ucc) \leftarrow .$$

$$P^3: lp(sma, ucc) \leq ls(ucc, sma) \leftarrow . \quad (lps)$$

上記で, 右側の  $ucc, sma, lp(d_1, d_2), ls(d_1, d_2), lps$  はそれぞれ左に記載されたルールのルール名を表す. 攻撃関係  $R_i$  として,  $attack$ , 即ち  $a = r \cup u$  を考える.

$Harg(P^1, P^2, P^3)$  の計算を説明する.  $J_4 := \emptyset, i := 3$  の初期設定の後, step1 で  $Args_{P^3} = \{[lp(sma, ucc) \leq ls(ucc, sma) \leftarrow]\}$  と  $R_3 = \emptyset$  が得られ, step2 で  $\leq_3 = \emptyset$ , step3 で,  $\sqsubseteq_3 = \emptyset$ , step4 で  $D_3 = \emptyset$  を求めて最小不動点  $J_3 = \{[lp(sma, ucc) \leq ls(ucc, sma) \leftarrow]\}$  が算出される. step5 で  $i = 2$  に更新し, go to step 1.

$P^2$  のルール名  $lp(sma, ucc), ls(ucc, sma)$  のルールをそれぞれ  $r_{lp}, r_{ls}$  で表すと,  $r_{lp} \in B_1, r_{ls} \in B_2$  なる論証  $B_1, B_2 \in Args_{P^2}$  が存在し,  $r_{lp}, r_{ls}$  の形より,  $sma \leq ucc \in conc(B_1), ucc \leq sma \in conc(B_2)$  である. よって定義 9 よりルール  $r_{lp}, r_{ls}$  を含む  $B_1, B_2$  は互いに *rebut* 攻撃をするので,  $(B_1, B_2) \in R_2$  かつ

$(B_2, B_1) \in R_2$  なる  $R_2$  が step1 で得られる. step2 で  $J_3$  を用いて  $\leq_2 = \{(r_{lp}, r_{ls})\}$  が得られる. よって step3 で  $r_{lp} \leq r_{ls}$  だから, 定義 4 に基づき  $(B_1, B_2) \in \sqsubseteq_2$  が得られる. step4 では, step3 の  $\sqsubseteq_2$  と step1 で得られた  $R_2$  より,  $(B_2, B_1) \in D_2, (B_1, B_2) \notin D_2$  なる  $D_2$  を求めて最小不動点  $J_2$  を算出し,  $B_2 \in J_2, B_1 \notin J_2$  が得られる. step5 で  $i = 1$  に更新し, go to step 1.

$P^1$  のルール名  $ucc, sma$  のルールをそれぞれ  $r_{ucc}, r_{sma}$  で表すと, 再び, step1 で  $r_{ucc} \in A_1, r_{sma} \in A_2$  なる論証  $A_1, A_2 \in Args_{P^1}$  について,  $perf \in conc(A_1), \neg perf \in conc(A_2)$  であるので互いに *rebut*, また  $perf \in conc(A_1), not \neg perf \in assem(A_2)$  かつ  $\neg perf \in conc(A_2), not \neg perf \in assem(A_1)$  であるので互いに *undercut* している. よって  $(A_1, A_2) \in R_1$  かつ  $(A_2, A_1) \in R_1$  が得られる. step2 で  $B_2 \in J_2$  かつ  $ucc \leq sma \in conc(B_2)$  であるから,  $(r_{ucc}, r_{sma}) \in \leq_1$  が得られ,  $i = 2$  の場合と同様に step3 で  $\leq_1$  を用いて,  $(A_1, A_2) \in \sqsubseteq_1$  が得られる. step4 では, step1 で得られた  $R_1$  と  $\sqsubseteq_1$  を用いて  $(A_2, A_1) \in D_1, (A_1, A_2) \notin D_1$  を求め,  $A_2 \in J_1, A_1 \notin J_1$  なる  $J_1$  を算出して return される.  $\neg perf \in conc(A_2)$  なる  $A_2$  が正当化されたので議論の結論として  $\neg perf$  (担保権が無い)を得る.

## 4 システム実装と評価

手続き  $Harg$  に従って階層的な議論を行うシステムを JAVA によって実装した. 本システムでは, 階層的知識ベースが記述されたファイルを入力とし, 攻撃関係, 議題を設定した後, 議論を開始する. システムの入力画面を図 1 に示す. システムの出力として各階層ごとに論証集合, 攻撃関係, 反映される論証間優先関係, 正当化された論証集合を画面表示し, 最終的に議題の勝敗の判定結果を表示する. 図 2 にオブジェクトレベルの結果出力画面を示す. 種々の例題を用いて, 定義に基づく計算結果と一致していることを検証した.

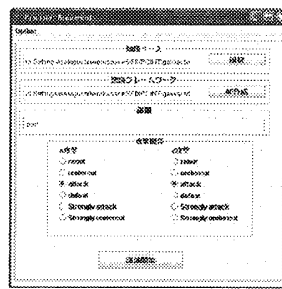


図 1: 入力画面

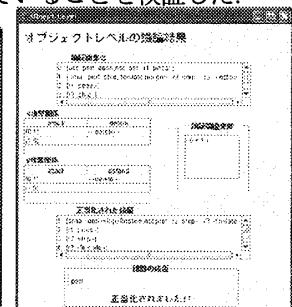


図 2: 結果出力画面

## 5 終わりに

本システムはネットワークを介さない単一エージェントシステムである. これを分散環境に対応するマルチエージェントシステムとして構築するのが今後の実装上の課題である.

## 参考文献

- [1] G. Brewka: Well-Founded semantics for Extended Logic Programs with Dynamic Preferences, Journal of Artificial Intelligence Research 4, pp. 19-36, 1996.
- [2] S. Modgil: Hierarchical Argumentation, Proc. of JELIA 2006, LNAI 4160, pp. 319-332, 2006.
- [3] H. Prakken and G. Sartor: Argument-based extended logic programming with defeasible priorities, Journal of Applied Non-classical Logics 7, pp. 25-75, 1997.
- [4] R. Schweimeier, M. Schroeder: A Parameterised Hierarchy of Argumentation Semantics for Extended Logic Programming and its Application to the Well-founded Semantics, Theory and Practice of Logic Programming, pp. 207-242, 2005.