

## 関係テーブルに基づく推論エンジンKORE/IE

7M-1

新谷虎松・平石邦彦・片山佳則・戸田光彦

(富士通(株)国際情報社会科学研究所)

### 1. はじめに

我々は、知的な意思決定支援システムを構築する環境として問題解決支援環境KOREを試作した[新谷 86]。KORE/IEはこのKOREにおける推論エンジンの機能を提供する。

KORE/IEはKOREにおけるルール指向的問題解決・知識表現機能を担う一方、単独ではOPS5[Forgy 81]等に代表される強力な前向き推論型プロダクションシステムとして機能する。本講演では、KOREにおけるKORE/IEの推論エンジン機能と非単調推論[Reiter 80]の実現方式について論じる。

### 2. 問題解決支援環境 KORE の概要

問題解決支援環境KORE(Knowledge Oriented Reasoning Environment)は問題解決を支援するための知識の制御機構と知識の構造化機能を提供する。知識の制御機構としてはKOREのサブシステムとして(1)KORE/IE(Inference Engine subsystem)以外に、(2)データ指向的制御を提供するKORE/DB(Data Base subsystem)、(3)オブジェクト指向的制御を提供するKORE/KR(Knowledge Representation subsystem)、(4)ネットワーク指向的制御を提供するKORE/EDEN(Extended Dependency Network subsystem)がある。これらサブシステムは、単独でも独立的に機能し、全体としてKOREに様々な問題解決支援機能を提供する。

KOREでは、KOREサブシステムで用いられる知識の内部表現を統一的に管理するための構造として関係テーブルを用いる。これらサブシステムは、内部表現において共通の関係テーブル表現をもつことにより、KOREとして統合化される(図1参照)。このような内部表現の統一性は問題解決過程において、KOREサブシステムを協調させるための手段を提供する。つまり、KOREの各サブシステムはこのような関係テーブルをコミュニケーションの場として用いることにより、他のサブシステムの問題解決過程の結果を利用したり、問題解決そのものを支援することが可能になる。

関係テーブルは、Hayes-Roth[1978]等が提案した黒板モデルにおける黒板と見ることができる。しかしながら、KOREにおける関係テーブルは異なった知識制御を担うKOREサブシステム間における情報の交換場所であり、KORE/DB

により管理される関係データベースであるという点に特徴がある。

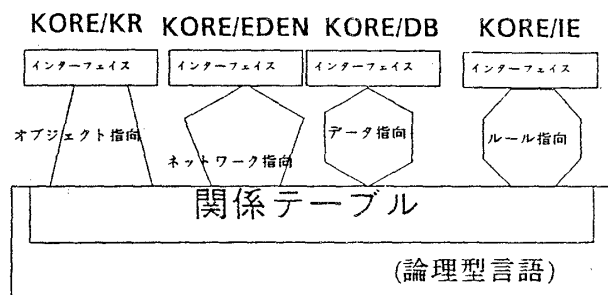


図1. KOREの構成

### 3. KORE/IE の概要

KORE/IEの基本的機能はOPS5をベースにしており、OPS5的なルール指向プログラミング環境を提供する。KORE/IEにおけるルールは次のシンタックスに従って記述される。

```

RULE_NAME : if Condition1 & Condition2 ...
             then Action1 & Action2 ...
    
```

ここで、RULE\_NAMEはルール名である。Condition, Actionは、それぞれLHS(もしくはIF部)におけるルールの条件要素(パターン)、RHS(もしくはTHEN部)におけるルールのアクションを示す。例えば、ルールは次のように記述される。

```

temp:
if weather_data(place=PLACE,temperature=TEMP) &
work(name=measuring_temperature,place=PLACE)
then
modify(1,[temperature=compute(TEMP+10)]).
    
```

例で示されるように、KORE/IEが扱うパターンは(1)クラス名と(2)いくつかの[属性=値]対により構成される。我々は、このように記述される情報を宣言的情報と呼ぶ。KORE/IEのワーキングメモリーはこのような宣言的情報の集まりであり、これはKORE/DBが管理する関係データベースに相当する。

PSはその構成要素として①PSI(PS Interpreter)、②WM(Working Memory)、③PM(Production Memory)がある。KORE

Inference Engine KORE/IE based on Relational Tables

Toramatsu SHINTANI, Kunihiko HIRAISHI, Yoshinori KATAYAMA, Mitsuhiro TODA  
International Institute for Advanced Study of Social Information Science, FUJITSU, Ltd.

/IE の特徴は、それらの内 WM(宣言的情報) と PM(手続き的情報) がKOREの統一的な内部表現形式である関係テーブルで表現され、PSI が制御機構であるKORE/IE として実現されることである。それゆえKORE/IE におけるWMの操作アクション(make, modify, remove 等) やトップレベルコマンドのliteralize (ルール中に使用されるクラス名, 属性名を宣言する) 等は表2で示すようにKORE/DB の関係テーブル操作コマンドを用いることにより実現される。

機能		KORE/DB	KORE/IE	機能	
テーブル定義		create	literalize	クラス定義	
タプル操作	挿入	ins	make	追加	WMの操作
	更新	update	modify	更新	
	削除	delete	remove	削除	

表2. KORE/IE とKORE/DB コマンドの対比 (同一性)

3. KOREにおける推論エンジン機能の実現

第2節で論じたように、KORE/IE と同様に、他のKOREサブシステムにおける宣言的情報も統一的な形式で関係テーブルとして表現されるので、KORE/IE は他のKOREサブシステムの推論エンジン部品として素直に組み込むことができる。

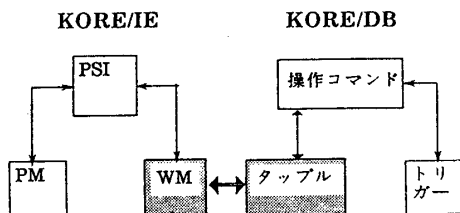


図2. KORE/IE とKORE/DB の協調

例えば、図2で示した例は、KORE/DB にKORE/IE を部品として組み込むことにより、関係データベースに推論エンジンを結合したことに相当する。つまり、KORE/DB によるタプルの追加はKORE/IE にとってはワーキングメモリー(WM)の変化に相当し、これによりルールが起動されるきっかけとなる。また、ルールの起動の結果、WMの変化はKORE/DB のトリガー起動のきっかけとなり関係テーブル(データベース)の変更をもたらす。このようにKORE/DB とKORE/IE は協調的に問題解決をはかることが可能になる。

図3で示した例は、KORE/KR とKORE/IE との協調問題解決の例を示している。つまり、KORE/KR のインスタンスの生成やそのスロット値の変更をKORE/IE におけるルールのトリガー条件として用いることができ、また、KORE/IE におけるルールの実行過程(WM)を構造化された知識としてKORE/KR を用いて管理することができる。これは、KORE/IE を推論エンジンとして、KORE/KR を知識ベースとして用

いたことに相当する。

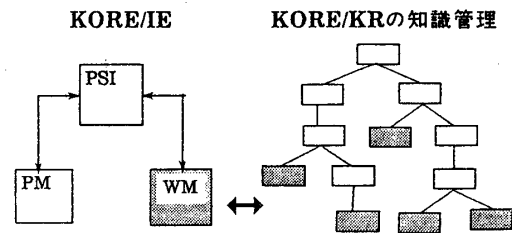


図3. KORE/IE と KORE/KRの協調

4. KORE/IE における非単調推論の実現方式

KORE/EDEN は、KOREにおいて知識を管理・利用するために必要となる知識間の関係情報を保存し、ネットワークを操作するための機能を提供する。このようなKORE/EDEN の関係情報の管理機能はKORE/IE に非単調推論のための基本的な情報等を提供する。

例えば、KORE/IE により新たに推論されたデータは、そのデータを支持するデータをjustification としてKORE/EDEN を用いて関係づけられることができる。KORE/EDEN の使用例は以下のように示せる。

```
rule1: if job(name=count_red_box) &
        box(color=red)
    then
        make(number,value=1)
    to eden.
```

この例では、クラスnumberのjustification としてLHS の各パターンにマッチするインスタンス(関係テーブル上のタプル) が記録される。このようなjustification は、truth maintenance のために必要であり、また、KORE/IE の説明機能を達成するための情報になる。つまり、このようなjustification を保存することにより、あるデータがWM(関係テーブル) から削除されたならそのデータに依存するデータをWMから自動的に消去する根拠を得ることができる。これにより、KORE/IE の非単調な推論を実現する。

5. おわりに

本論文では、KORE/IE の推論エンジン機能について論じた。このような推論エンジン機能は、関係テーブルに基づいたKOREサブシステムの統合形式により素直に実現した。また、KORE/IE では、OPS5で用いられているReteアルゴリズムに基づくルールコンパイル方式を用いることにより、その高速化を実現している。現在 KORE/IEはVAX11/780 上のC-Prolog (約800 ステップ) でインプリメントされている。なお、本研究は、第五世代コンピュータ・プロジェクトの一環として行われたものである。

[参考文献]

[1] C.L.Forgy : OPS5 User's Manual, CMU-CS-81-135, July, (1981).  
 [2] R.Reiter : A Logic for Default Reasoning, AI 13, pp.81-132, (1980).  
 [3] 新谷, 他: 論理型言語によるハイブリッドな問題解決支援環境 KORE, The Logic Programming Conference 86, 予稿集 pp.43-50, (1986).