

TR-167

論理型言語によるハイブリッドな
問題解決支援環境 KORE の設計
— 知的意志決定支援システムへの接近 —

新谷虎松, 片山佳則, 平石邦彦, 戸田光彦
(富士通)

April, 1986

©1986, ICOT

ICOT

Mita Kokusai Bldg. 21F
4-28 Mita 1-Chome
Minato-ku Tokyo 108 Japan

(03) 456-3191~5
Telex ICOT J32964

Institute for New Generation Computer Technology

論理型言語によるハイブリッドな 問題解決支援環境KOREの設計

—知的意思決定支援システムへの接近—

新谷虎松・片山佳則・平石邦彦・戸田光彦
富士通国際情報社会科学研究所

1. はじめに

本研究の目的は人間の意思決定を支援するための知的な問題解決支援ツールを構築することである。人間の意思決定を支援するツールとしては、(1)デシジョンサポートシステム(データベースやモデルベースに基づく支援) [Ginzberg 82]、(2)エキスパート・システム(領域依存の知識ベースに基づく支援) [Hayes-Roth 84]、(3)意思決定分析法(ユーザのバックグラウンドな知識に基づく支援) [市川 83]、[ケブナー 85]等が良く知られている。

(1)で代表される既存の意思決定支援システム(DSS)の多くは、主にデータベースに基づく機能を提供する。これら多くの既存DSSの主な利用法としては、データの出力、操作、管理、簡単な分析処理等が挙げられる。このような既存DSSを用いて高度な支援機能を発揮させるためには、DSSの専門家を仲介とするか、ユーザ自身がシステムに習熟する必要がある。DSS専門家は、意思決定者が要求する概念をDSSが受け入れることのできる特定の質問へ変換し、DSSを直接に運用する。もし、質問に対するデータ、ルーチンそしてモデルが欠けていたなら、それらを補う。これには多くの時間が必要とされる[広内 83]。

一般に、DSSを運用するにはこのような様々な不都合が生じる。このような不都合は主にDSSにおけるモデルの利用の困難さとDSSそれ自身の利用法の困難さに起因している。ここで言うモデルとは「データや判断を処理するための一連の処理手続き」を示す。

知的な意思決定支援機能を計算機上で効果的に実現するためには、以上で述べた既存DSSの問題点の解決を考慮したシステムを設計する必要がある。このようなシステムは、DSS専門家(もしくは問題領域の専門家)の知識や決定問題にたいするノウハウを知識ベースに実現したものであり、(2)のエキスパート・システムの機能や(3)の意思決定分析法による支援機能を導入したものとなる。我々はこのようなシステムを知的DSSと呼ぶ。

本論文では、このような知的DSSに必要な機能を明らかにし、そこから得られたニーズを実現するために構築した問題解決支援環境KORE(Knowledge Oriented Reasoning Environment)について論じる。

2. 知的DSSの支援機能

我々は知的DSSにおける必要な支援機能として以下の機能を考慮する。(1)データやモデルの蓄積と利用を支援する機能、(2)意思決定問題に関連した知識としてユーザが気付いた知識の断片を積極的にシステムの中に取り込み利用する機能、(3)専門家の知識を知識ベース化することによるリアルタイムな意思決定支援機能

2.1. モデル知識ベースに基づく支援

(1)のデータやモデルに基づく支援機能は、既存DSSのデータ指向・モデル指向的機能[Alter 84]を知的に支援するためのものである。この支援においては、ユーザは望むデータを要求することにより、その求めるデータを得ることができる。つまり、DSSはその要求に基づくモデルをDSS自身が構築することにより、その要求に答えることになる。我々はこのような支援をモデル知識ベースに基づく支援と呼ぶ。

Bonczek等[Bonczek 81]は、DSSにおけるモデル指向的支援を達成する手段として一階述語論理の有効性を論じている。彼等が提案する方法は、モデル間結合のメカニズム処理を中心とするものである。具体的には、応用領域特有のモデル化知識を述語論理で表現し、得られた公理の集合を知識ベースに蓄積する。モデルを使った処理が必要なユーザの要求に対して、知識ベースを用いてモジュールを自動的に組み合わせ、必要となるデータをデータベースから抽出し、導出原理を使って実行する。この方法は、知的DSSにおけるモデル管理機構を構築するうえで有効であり、論理型言語(Prolog等)を用いることにより素直に実現できる。

そこで、我々は以上のようなモデル知識ベースに基づく支援機能構築のニーズとして、論理型言語に基づくモデルベースやデータベースの構築・利用機能を挙げ、これを実現するために論理型言語を用いたデータベースの設計とこのデータベースに推論エンジンを結びつける機構をKOREに実現する。

2.2. ユーザ知識に基づく支援

(2)のユーザの知識に基づく支援機能は、構造化されてい

ない意思決定を意思決定分析法 [Simon 79] , [Toda 82] , [近藤 81] , [Gaglio 85] 等に基づき知的に支援するためのものである。このモードにおけるDSSは自分自身の情報は提供しないが、ユーザ自身の知識を探索したり、構造化することにおいてユーザを支援し、問題解決のための他に必要な行為についてアドバイスを提供する。我々は、このような支援をユーザ知識に基づく支援と呼ぶ。

ユーザ知識に基づく支援はdomain-independentである。この種の支援はユーザが自分のバックグラウンドな知識と専門知識の大部分を用いることを想定しており、意思決定問題に関連した知識としてユーザが気付いた知識の断片を積極的にDSSのなかへ取り込む。このユーザ知識に基づく支援においては、DSSは知的なノートとして機能する。つまり、このモードにおけるDSSは自分自身の情報は提供しないが、ユーザ自身の知識を探索したり、構造化することにおいてユーザを支援し、問題解決のための行為の他に必要な方向についてのアドバイスを提供する。

このようなユーザ知識に基づく支援を行うシステムはユーザと会話することにより情報を交換する。この会話はまず最初に目標を述べることから始まり、これら目標を実現するさらに詳細な手段を説明して行く。このシステムはユーザの注意を問題に対して決定的である論点へ集中するよう誘導する。つまり、このようなユーザ知識に基づく支援は図1のようなユーザ支援サイクルを実行することにより、ユーザから意思決定支援のための情報を獲得し、構造化することによって、ユーザの意思決定を支援する。

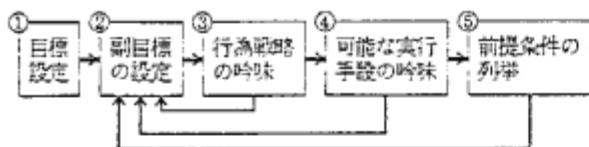


図1. ユーザ知識に基づく意思決定支援サイクル

図1において①～⑤は以下のことを示している。

- ①目標設定: 意思決定者の主目標の設定
- ②副目標の設定: 全体的な目標を達成するために構成する詳細な項目、目標の属性(attributes)もしくは目標の次元(dimensions)の設定
- ③行為戦略の吟味: 特定の副目標を達成するために意思決定者が採用できる主な行為の戦略(strategies)の吟味
- ④実行可能な手段の吟味: 各々の行為を遂行する際に実行可能な手段の吟味
- ⑤前提条件の列挙: 特定の手法を効果的に進めるために必要となる環境もしくは自然の状態の列挙

図1は意思決定ネットワークとして考慮されるべきであ

る。従って、目標はいくつかの副目標へ分割され、各々の副目標はその副目標を達成するであろういくつかの可能な行為(action)を有しており、各々の行為はその行為を遂行するいくつかの方法を有しており、また、各々の方法は達成されなければならないいくつかの前提条件を有している。一度、そのような前提条件が明確化されたなら、それら前提条件はさらに新たな副目標を導くことになる。もし、前提条件の達成がユーザの直接的なコントロールの範囲外であったり、前提条件の達成が外部的な偶発性に依存しているれば、そのような前提条件は不確かなイベントノードとして扱われる。

ユーザは、システムとの会話の間、上でのべた①～⑤間の様々な関係の強さを指定することを要請される。これら、指定された量は意思決定ネットワークを通じて伝播される。このことにより、ユーザは主目標を実現させるために最も決定的である論点へと会話が集中させられる。この意思決定支援のためのサイクルは再帰的に繰り返され、ユーザが主目標に対して満足する行為や戦略を見出すことにより、ユーザにより終結させられる。その時ユーザはそこまで行われた意思決定ネットワークの提示とそれより得られる結論とをシステムからレポートの形で得ることができる。

そこで、我々は以上のようなユーザ知識に基づく支援機能構築のニーズとして、知識を構造化し利用する機構や知識の一貫性を保持管理するための機能を挙げ、これを実現するためにオブジェクト指向概念に基づく知識表現機構とネットワーク構造を管理するための機構をKOREに実現する。このようなネットワーク管理機構は暗黙推論 [Reiter 80] やtruth maintenance system [Doyle 78] を実現する上で重要なdependency-directed backtracking [Stallman 77] のための情報 [McDermott 83] を提供する。

2.3. 知識ベースに基づく支援

(3)の知識ベースに基づく支援機能は、問題領域特有なヒューリスティクスを用いることにより知的に支援するためのものである。このような支援は一般に知識ベースに基づく支援と呼ばれ、エキスパートシステムの機能として良く知られている [Hayes-Roth 83]。知識ベースに基づく支援は、前もって必要とされる知識を知識ベースに蓄積・利用することにより、ユーザの意思決定のための知的な支援を可能にする。

このことは、知識ベースに基づく意思決定支援機能を実現するために、対象領域に依存した知識が整理・記述されることの必要性を示している。つまり、対象領域ごとに記述し易い知識表現形式を採用する必要がある。我々は、このような要求に対してKOREにハイブリッドな知識表現形式を導入することにより対処する。しかしながら、様々なブ

プログラミングパラダイム（オブジェクト指向・ルール指向・アクセス指向等）[Bobrow 84] を一つの知識表現形式に押し込むことは、プログラマ以外のエキスパートが直接にその表現技法を用いることを困難にし、また、その技法によるアプリケーションの構築も困難にする。そこで、我々は、論理型言語上に関係テーブル概念を用いて様々なプログラミングパラダイムの新たな統合化法をKOREで実現する。

3. 問題解決支援環境KOREの設計

前節で述べた知的DSSの機能を具体的に計算機上に実現するために、我々はPrologをベースにした問題解決支援環境KORE(Knowledge Oriented Reasoning Environment)

[新谷 86a] を構築した。知的DSSはこのKOREを用いることにより構築される。我々は、KOREを用いて構築することの知的DSSをKORE/DSSと呼ぶ。KORE/DSSはKOREのサブシステムを効果的に用いることにより、知的DSSとして考慮されるべき支援機能を実現する。KOREはKORE/DSSに必要なこれらの機能を実現するための次のようないくつかのサブシステムにより構成される。(1)KORE/DB(Data Base subsystem)[平石 86]、(2)KORE/IE(Inference Engine subsystem)、(3)KORE/KR(Knowledge Representation subsystem)[片山 86]、(4)KORE/EDEN(Extended Dependency Network subsystem)[新谷 86b]。

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。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2参照)。この関係テーブル表現はPrologにおける関係表現として実現される[Kowalski 77]。

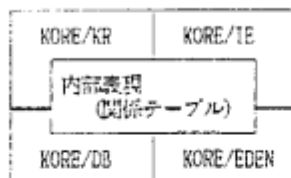


図2. KOREサブシステムの統合

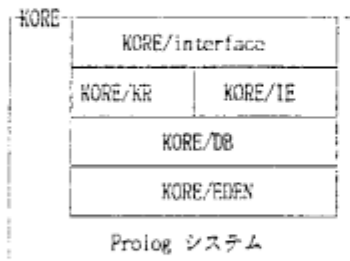


図3. KOREの構成

また、各サブシステムはそのシステム記述言語である

Prolog（もしくはESP[Chikayama 84]）とのインターフェイスをもつことにより、システム記述言語が持っている様々な機能（例えば、Prologにおける論理型プログラミングの能力等）を利用することを可能にする(図3参照)。

KOREにおけるサブシステムは図4で示すように関係テーブルを用いてその内部表現を統一化する。このような内部表現におけるKOREサブシステムの統一化は、システムが提供するプログラミングパラダイム（例えば、オブジェクト指向、ルール指向、アクセス指向、論理指向などの概念）を素直にKOREに取り込み、KOREを用いたアプリケーションの開発を容易にする。(図5参照)

図4における「統一表現の関係」とは、各サブシステムにおいて同じ関係テーブル構造(関係スキーマ)を用いているものであり、そのような関係にあるものが図では実線により結ばれている。図で示すように統一表現には、①の関係情報、②の宣言的信息、③の手続き的信息に関する3種類が存在する。

サブシステム	内部表現の種類	統一表現の関係
KORE/KR	(1) オブジェクト間の関係情報 (2) スロット情報 (3) 各メソッドの情報	①関係情報 ②宣言的信息
KORE/IE	(4) ワーキングメモリー要素 (5) ルールの情報 (6) ルール間の関係情報	②宣言的信息
KORE/DB	(7) テーブルの関係スキーマ (8) タプル(テーブルの行) (9) トリガー情報	③手続き的信息
KORE/EDEN	00 知識間の関係情報	③手続き的信息

図4. KOREサブシステムにおける内部表現

KOREサブシステム名	プログラミングパラダイム	KOREにおける機能
KORE/KR	オブジェクト指向	知識ベース
KORE/IE	ルール指向	推論エンジン
KORE/DB	データ指向	データベース
KORE/EDEN	ネットワーク指向	知識管理機構

図5. KOREにおけるサブシステムの機能

一般的に、多くのプログラミングパラダイムを同じ言語内に取り込んだシステム（例えば、[Bobrow 83]）は、非常に複雑な言語シンタックスになりがちであり、このことはその言語を用いたアプリケーションの開発を困難にすることがある。KOREは有用なプログラミングパラダイムを固有なサブシステムに分散実現することにより、様々なプ

プログラミングパラダイムを容易に用いることを可能にし、かつ、それらサブシステム間の通信（情報交換）は、共通の内部表現を保有することにより容易に達成する。

これらサブシステムの基本機能の概要は次のように記述することができる。(1)のKORE/DBは、KOREにおける関係テーブルを操作管理する機能を担う一方、単独では関係データベースを操作するためのパーソナルな関係データベースシステムとして機能し、KORE/DSSにおいてはデータベース指向的支援機能を提供する。(2)の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OPSS [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し（KORE/IEの基本的機能や言語シンタックスはOPSSをベースにしている）、KORE/DSSにおいてはモデル知識ベースに基づく支援を提供する。(3)のKORE/KRは、知的DSSにとり必要となるオブジェクト指向概念 [Goldberg 83] に基づく知識表現・利用機能を提供する。また、KORE/KRはKORE/DSSにおいてDSS専門家の知識を表現することにより、KORE/DSSに知的なマンマシンインターフェイスを提供する。図3において、KOREのインターフェイスであるKORE/interfaceはこのKORE/KRを用いて構築される。(4)の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、知識ベース管理機能の一機能を担う一方、単独ではネットワーク [Brachman 83] を操作するための強力なシステムとして機能する。これら基本サブシステム間の関係は図3のように図示できる。

4. 関係テーブルの表現とその操作

関係データベースにおけるすべてのデータは関係概念によってグループ化され、関係テーブルとして表現できる。このような関係テーブル表現はPrologにおける関係表現を用いて素直に表現ができ [Kowalski 77]、また、Prologはこの関係テーブルを操作するための基本的機能を提供する。例えば、次のテーブル1で示すような関係テーブル（もしくは、関係データベース）は

測定	名前	体重	身長
	太郎	80	172
	次郎	75	180
	花子	55	157

テーブル1. 関係テーブルの例

Prologのアサーションにより関係表現として以下のように記述できる。

測定 (太郎, 80, 172)

測定 (次郎, 75, 180)

測定 (花子, 55, 157)

ここで、“測定”はテーブル名であり、“名前”、“体重”、“身長”は属性を示す。

このようなn項関係表現における引数の位置は、テーブルの列（関係データベースにおける属性）に相当し、各々の関係はテーブルの行（関係データベースにおけるタプル）に相当する。KOREでは、このような関係テーブルをKORE/DBを用いて表現・蓄積・操作利用する。KORE/DBは次のような関係テーブル操作機能を提供する。

①関係テーブル（関係スキーマ）の定義機能

②データ操作機能

(a)関係の追加、(b)関係の更新、(c)関係の削除、

(d)トリガ機能、(e)ルール付加機能

③問い合わせ（検索）機能（ビュー機能）

(a)セレクション、(b)プロジェクション、(c)ジョイン、

(d)セット・オペレーション (union, intersection, difference)

ここで、②の(d)のトリガ機能はテーブルへの関係の追加、更新、削除を行ったときに自動的に起動する手続き（デモン）である。また、②の(e)は検索時に起動される手続きであり、Prologのルール形式を保存することにより達成する。テーブル1を例にすると、ルールは以下のように用いられる。例えば、“検索時に、体重70kg以下の人がいたらその名前を印字せよ”のルール定義は

```
attach_method(write(#名前),nl)
into 測定 where 体重 < 70.
```

で表現され、これを実行すると、花子に関するタプルが次のようになる（花子だけが体重が70以下の条件を満たしている）。

```
測定 (花子, 55, 157) :- write(花子),nl.
```

以上のように、Prologを用いた関係テーブルの実現は、これら関係テーブル（関係データベース）における、タプル、ビューそして特別なテーブル操作プログラム等が同じ言語で表現できるという利点を提供する。このような利点は、特に、データベースにおけるintegrity constraintを容易に記述することを可能にする [Kowalski 78]、[Yokota 85]。

KORE/DBは以上のような関係データベースシステムの基本的機能を実現しているが、一般的な関係データベースとしては、例えば、以下の点等で不十分である。

①実際に扱うデータベースのサイズが一般的関係データベースより小さい。

②同時に複数のユーザの使用を前提としていない。

③ハードウェア・エラーによるシステムの回復処理を考慮していない。

これら問題点はPrologを用いたプログラミング技法により対処できる [Dahl 82]。しかし、KORE/DB はパーソナルな関係データベースシステムであり、かつ、KOREにおける基本的関係テーブル操作機構として機能するので、以上のような一般的な関係データベースに必要となる機能は、現在、提供していない。つまり、KORE/DB はKOREのサブシステムに共通に必要となる内部状態（関係テーブル）を記述・操作するために用いられ、かつ、小規模でパーソナルなデータベース機能を提供する。

5. KOREサブシステムの統合化

KORE/KR と KORE/IE の統合化は、オブジェクト指向概念とルール指向概念の統合化であり、KOREにおいてそれぞれが知識ベース、推論エンジンとして機能することに相当する。また、KORE/DB と KORE/IE の統合化は、2.1節で論じたモデル知識ベースに基づく支援機能を達成するために必要としたデータベースと推論エンジンの結びつきに相当する。KORE/KR と KORE/EDEN の統合化は、2.2節で論じたユーザ知識に基づく支援機能を達成するために必要とした知識の構造化とネットワーク管理の機能を実現するものである。また、全てのサブシステムの統合化は、2.3節で論じた知識ベースに基づく支援機能を達成するためのハイブリッドな知識表現技法を提供する。

5.1. 宣言的情報の統一化

KOREは宣言的情報を図6で示すような関係テーブルスキーマを用いて表現する。この関係テーブルにおいて、各々のタプル（テーブルの行）は、KORE/IE にとっては図7左に示すようなWM要素（ワーキングメモリー(WM)の要素 [Forgy 81]）であり、図7のようにクラスを示す名前といくつかのスロット名—スロット値ペアから構成される）であり、KORE/KR にとっては図7右に示すようなインスタンスのスロット（図7右のようにクラス定義文中のキーワード value _set から";" までの間にスロット名—スロット値ペアとして記述されている）を表現するために用いられる。

図6において<NAME>,<slot1> ... <slotN> は、それぞれ使用時に決まる具体的なクラス名、スロット名を表す。その他の属性は付加的に重要な役割を果たす。例えば、属性timeはタプルが生成された時刻を示すために用いられる。このような時刻はKORE/IE においてはワーキングメモリー要素のタイム・タグ（ルールの競合解消時に使われるキーとなる）として、またKORE/KR ではインスタンスの生成時期を表すために用いられる。属性usage はタプルの用途に関する情報を記述するために用いられ、KORE/IE で

は宣言的情報に関する信念を表現することにより、非単調な推論 [Reiter 80] のための制御情報として用いる。属性table は他に参照すべき関係テーブル名を記述するために用いられる。

db_<NAME>_slot						
<slot1>	...	<slotN>	time	usage	author	table
:	:	:	:	:	:	:

図6. KOREにおける宣言的情報の関係テーブルスキーマ

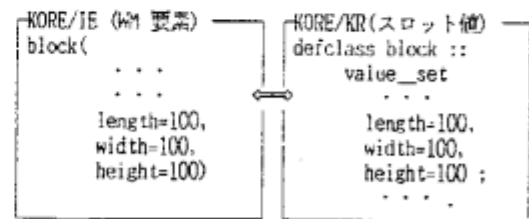


図7. KORE/IE やKORE/KR における宣言的情報

このようなKOREサブシステム間における宣言的情報の統一化は、KOREサブシステム間の協調した問題解決支援のための基本的な機能を提供する。例えば、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（ワーキングメモリー）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（図8参照）。これは、次節で述べる手続き的情報の統一化と共に、有用なプログラミングパラダイムを固有なサブシステムに分散実現し、これらシステムを協調して用いることを可能にする。

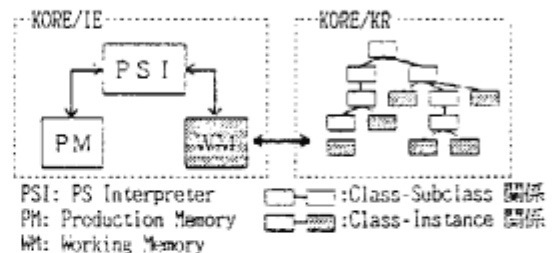


図8. KORE/IE とKORE/KR における宣言的情報の共用

5.2. 手続き的情報の統一化

KOREにおける手続き的情報は図9で示す関係テーブルスキーマにより表現される。

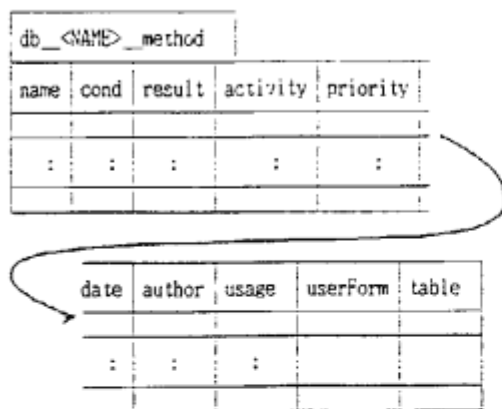


図9. KOREにおける手続き的情報の関係テーブルスキーマ

図9において<NAME>は使用時に決まる具体的な (KORE/KRにおける) クラス名もしくは (KORE/IEにおける) ルールベース名を表す。また、この関係テーブルにおける各属性は、例えば、図10で示すようなKORE/IE やKORE/KR の手続き的情報に関して図11のように用いられる。

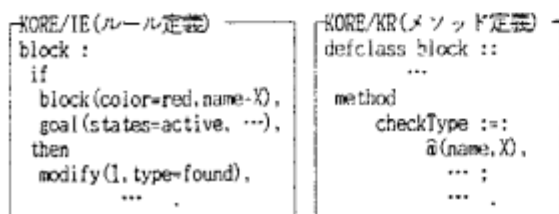


図10. KORE/IE とKORE/KR における手続き的情報

属性名	KORE/KR	KORE/IE
name	メソッドのセクター名	ルールに付加されているルール名
cond	メソッドのセクター	ルール起動条件部
result	メソッドの本体	ルール結論部
activity	手続き起動に際しての制御情報	
priority		
date	手続きの保守管理のためのデータ	
author		
usage		
userform		
table	手続き群を階層化するための情報	

図11. 図9における属性の利用形態

KORE/IE やKORE/KR の手続き的情報は、KORE/DB におけ

るコマンドとして翻訳され (翻訳される前の表現はテーブルの属性userFormに格納される)。図9に示すような関係テーブルに格納される。例えば、KORE/IE におけるルール定義において、属性condにはKORE/DB における検索に関するコマンド文 (fetch文) が、属性resultにはKORE/DB における検索・更新に関するコマンド文 (fetch文やupdate文) がその内部表現として記述される。ルールの実行に際しては、ルール条件部から結論部への変数の伝播のために、この2つの属性に記述されたKORE/DB のコマンドは cond-resultペアーとして、一つのゴール節に展開される。例えば、図10左におけるKORE/IE のルール定義は、以下のようなKORE/DB のコマンドへと翻訳される。

```
:- (fetch * from db_block_slot
    where color=red and name=X),
   (fetch * from db_goal_slot ...), !,
   (update db_block_slot set type=found)
```

KOREにおけるこのような手続き的情報の統一化は、関係テーブルを操作するための知識形式として様々なプログラミングパラダイムを用いることができることを意味している。このことは、KORE/DB を用いて様々な知識形態を統一的に管理できることを示し、また2.1節で論じたモデル知識ベースに基づく支援機能を達成するためのデータベースと推論エンジンの統合化が素直に実現することにもなる。

5.3. 関係情報の統一化

KOREでは、知識利用を効果的に達成するために、知識 (もしくはデータ) 間の関係情報を統一的に表現・管理する独立した機構としてKORE/EDEN を用いる。これら関係情報には、例えば、isa 関係 (知識の汎化階層を構造化する)、partof関係 (知識の部分全体関係の推移律を構造化する)、Data dependencies (知識導出における説明情報を構造化する) [McDermott 83] 等があり、一般に、図12で示すようなネットワーク構造を形成する。

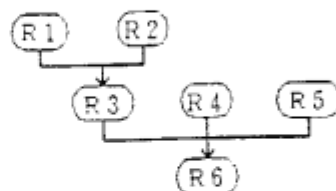


図12. 関係ネットワークの例

KORE/EDEN ではこのネットワーク構造を図13で示すような知識テーブル [Shintani 86] と呼ぶ特別な関係テーブルで表現し管理する。この知識テーブルは、ネットワークに

におけるノード間の隣接関係や到達可能性関係に関する情報を表現する行列（図13左）であり、その内部表現としては行要素を2進数に圧縮し正整数として保存する関係テーブル（図13右）で表現される（これにより知識テーブルの記憶容量の削減と操作の効率化を達成する）。

例えば、図13左において、ノードR3に関する行要素（斜線部分の“0101”であり、その他の行要素“00”は次の理由で省略できる）は関係テーブル表現において5（“0101”を2進数とみなすことより得られた正整数）として表現される。ここで、図13右における関係テーブルにおける“<NAME>”は使用時に指定される関係名（例えば、isaやpartof等）であり、node、element、positionはそれぞれ知識テーブルにおける行の名前（ノード名）、行要素を表す正整数、行の位置を表す属性である。また、テーブルの中の“—”は未定義なことを表す。

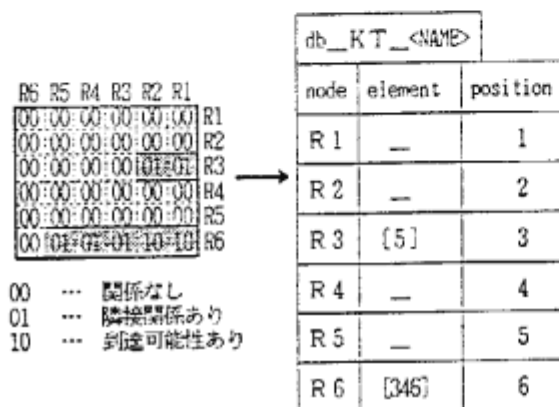


図13. 図12を表現した知識テーブルの例

知識テーブルへの操作は（関係の付加、削除、知識テーブルからの検索）これら正整数（2進数列）に対してビット論理演算操作（整数ビット論理積・和、ビットシフト等）を適用することにより効率的（行要素を整数として一度に扱うから）に達成する〔新谷 85c〕。

以上のようなKORE/EDENの関係情報の管理機能はKOREにtruth maintenance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。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。さらに、このようなjustificationを保存することにより、あるデータがデータベースから削除されたならそのデータに依存するデータを自動的に消去することも可能になる（例えば、図12においてノードR2が消去されたなら、R3とR6とを消去するための根拠を得ることができる）。

6. KOREの特徴

KOREは、知的意思決定支援システム構築のために必要とされるハイブリッドな知識表現技法を実現するために、既存のハイブリッドな知識プログラミングツール（例えば、KEE [Fikes 85]、LOOPS [Bobrow 83]、等）のアプローチとは異なり、固有な知識表現パラダイムを提供するサブシステムにより構成される。これらサブシステムは論理型言語上で関係テーブルを用いることにより統合される。このような統合化は、いたづらにシステムの言語シンタックスを複雑にすることを避けるために、そして、各知識表現パラダイム特有の強力な知識プログラミング技法をKOREに実現するために導入された。

KOREにおける基本機能は知識の制御と知識の蓄積・利用に大別される。つまり、知識の制御はKOREの各サブシステムに分散され、サブシステム固有の知識プログラミング技法を提供する。知識の蓄積・利用は関係テーブルとして統一化される。このことにより、各サブシステムは、KOREのアプリケーションを構築する際の部品（知識制御機構）となり、様々なアプリケーションを構築するために組み合わせて用いられる。

7. おわりに

本研究の目的は、人間を知的に支援するための知的意思決定支援システムを構築するためのツールの構成を明らかにし、それを構築することにある。そこで、本論文では知的意思決定支援システム構築のためのニーズを明らかにすることにより、本研究で試作した問題解決支援環境KOREの有効性について論じた。KOREではアプリケーション構築を容易にするための工夫として、様々な知識プログラミングパラダイムをパラダイム固有のサブシステムへ分散化し、それらサブシステムを関係テーブルを用いることにより統合化した。このことにより、KOREでは問題領域に依存した知識を、自然なシンタックスを使ってその問題領域に適した形態で記述することを可能にする。このような分散一統合化は、既存の意思決定支援ツール（エキスパートシステム、DSS等）を素直に統合化することに相当し、KOREにハイブリッドな問題解決支援機能を提供する。また、関係テーブルによるサブシステムの統合化の実現は、論理型言語の強力な関係記述能力をKORE/DBとして集約し、積極的に利用したことによる。KOREの能力の拡張（例えば、知識ベースの2次記憶の利用等）は、このように能力を集約したKORE/DBを拡張することにより達成できる。

尚、本研究は、第5世代コンピュータ・プロジェクトの一貫として行われたものである。

参考文献

- [Alter 80] S.L.Alter : Decision Support System: Current Practice and Continuing Challenges, (Addison-Wesley, Reading, Mass., 1980).
- [Bobrow 83] D.G.Bobrow and M.J.Stefik : The LOOPS Manual. KR-VLISI-81-13, 1983.
- [Bobrow 84] D.G.Bobrow : IF PROLOG IS ANSWER WHAT IS THE QUESTION, The International Conference on Fifth Generation Computer System, pp.133-145 (1984).
- [Bonczek 81] R.H.Bonczek, C.W.Hoisapple and A.B.Winston, A Generalized Decision Support System Using Predicate Calculus and Network Data Base Management, Operations Research, Vol.29, No.2, pp263-281 (1981).
- [Brachman 83] R.J.Brachman : What IS-A is and isn't : An Analysis of Taxonomic Links in Semantic Networks, COMPUTER (1983) October.
- [Chikayama 84] T.Chikayama : ESP Reference Manual, ICOT Technical Report, TR 044, (1984).
- [Dahl 82] V.Dahl : On Database Systems Development Through Logic, ACM Transactions on Database Systems, Vol. 7, No.1, pp.102-123, March (1982).
- [Doyle 78] J.Doyle : Truth Maintenance System for Problem Solving, MIT, AI-TR-419, (1978).
- [Fikes 85] R.Fikes and T.Kehler: The Role of Frame-Based Representation in Reasoning, Communication of the ACM, Vol.28, No.9, (1985).
- [Forgy 81] C.L.Forgy, OPS5 User's Manual, CMU-CS-81-135, July (1981).
- [Gaglio 85] Gaglio, S. : Multiperson Decision Aspects in the Construction of Expert Systems, IEEE Transactions on Systems, Man and Cybernetics, Vol.15, No.4, (1985).
- [Ginzberg 82] M.J.Ginzberg and E.A.Stohr, Decision Support Systems: Issues and Perspectives, in Decision Support Systems (M.J.Ginzberg, W.Reitman and E.A.Stohr, eds., North-Holland, Amsterdam, 1982), pp9-31.
- [Goldberg 83] Goldberg, I., Robson, D. : Snailtalk-80 : The Language and it's Implementation, Addison-Wesley, (1983).
- [Hayes-Roth 83] F.Hayes-Roth, D.A.Watterman, and D.B.Lenat (Ed.): Building Expert System, Addison-Wesley, (1983).
- [Hayes-Roth 84] F.Hayes-Roth : The Knowledge-Based Expert System: A Tutorial, COMPUTER, September, 1984, pp.11-28.
- [平石 86] 平石邦彦, 新谷虎松, 片山佳則: 問題解決支援環境 KORE (その4) - 関係データベース・サブシステム KORE/DB とその概要 -, 情報処理学会第32回全国大会 (1986).
- [広内 83] 広内, 小坂: 意思決定支援システム, 竹内書店新社 (1983).
- [市川 83] 市川博信編: 多目的決定の理論と方法, 計測自動制御学会 (1983).
- [片山 86] 片山佳則, 新谷虎松, 平石邦彦: 問題解決支援環境 KORE (その3) - 知識表現サブシステム KORE/KR とその概要 -, 情報処理学会第32回全国大会, 5L-10, 1986.
- [ケプナー 85] C.H.ケプナー, B.B. トリゴー (上野訳), 新・管理者の判断力, 産業能率大学出版部 (1985).
- [近藤 81] 近藤次郎: 意思決定の方法 P D P C のすすめ, NIK ブックス394 (1981).
- [Kowalski 77] R.Kowalski, Logic for Problem Solving Elsevier North Holland, (1977), pp31-44.
- [Kowalski 78] R.Kowalski, LOGIC FOR DATA DESCRIPTION, in H.Gallaire (Ed.), "LOGIC AND DATA BASES", Plenum Press, N.Y. (1978).
- [McDermott 83] D.McDermott : Contexts and Data Dependencies : A Synthesis, IEEE, Vol. PAMI-5, No.3, pp.237-246 (1983).
- [Reiter 80] R.Reiter: A Logic for Default Reasoning, Artificial Intelligence, 13, pp.81-132, (1980).
- [Simon 79] H.A.Simon (稲葉他訳): 意思決定の科学, 産業能率大学出版部 (1979).
- [新谷 86a] 新谷, 片山, 平石, 戸田, "問題解決支援環境 KORE (その1) - 概要 -" 情報処理学会第32回全国大会, 5L-8, pp1143-1144, (1986).
- [新谷 86b] 新谷, 片山, 平石: 問題解決環境 KORE (その2) - 知識記憶利用機構 KORE/EDEN とその応用 -, 情報処理学会第32回全国大会, 5L-9, (1986).
- [新谷 85c] 新谷, 片山: 知識テーブル (その実現) - 知識ベースにおける知識記憶管理方式とその応用 -, 情報処理学会第31回全国大会, 1P-8, (1985).
- [Shintani 86] T.Shintani : Knowledge Table: An Approach to Knowledge Utilization Mechanism for Knowledge Base, Fujitsu IIAS-SIS Research Report, 1986, 38p.
- [Stallman 77] R.M.Stallman, G.J.Sussman : Forward Reasoning and Dependency-Directed Backtracking in a System for Computer-Aided Circuit Analysis, Artificial Intelligence ,9, pp.135-196 (1977).
- [Toda 82] M.Toda, K.Sugiyama, S.Tagawa : A Method for Aggregating Ordinal Assessment by a Majority Decision Rule, Mathematical Social Science 3, pp227-242 (1982).
- [Yokota 85] H.Yokota, K.Sakai, H.Ito : Deductive Database System based on Unit Resolution, ICOT Technical Report, TR-123, (1985).