

## エキスパートシェル MIND-1

4B-6

— 概要 —

丹羽 雅司 谷口 久衛 山本 雅基 藤井 敬久

日本電装株式会社

## 1 はじめに

FA分野で容易に導入でき、種々の対象分野に適用可能なエキスパートシステム構築用ツール MIND-1(Machine Intelligence at NipponDenso)を開発したので報告する。本稿では、エキスパートシステム構築用ツール MIND-1の開発背景、開発思想、特長、機能、性能評価について報告する[1]。

## 2 MIND-1の開発背景

現在、多数のエキスパートシステムが開発されているが、FA分野でエキスパートシステムの使用が充分であるとは言い難い。これは、FA分野における実状を考慮したエキスパートシェルが提供されていないからである。

FA分野の実状を、エキスパートシステムに関して考察すると、次のような点が重要である。

領域専門家自身で開発・保守が可能

製品技術と生産技術での幅広い範囲の固有技術の知識と、生産現場で生産設備を効率良くかつ安定に動かすノウハウが存在する。生産現場のノウハウは、現場の作業者がQC活動等の職場活動と連動し、設備の調整、改善等の継続的な試行錯誤の積み重ねにより増大し、また製品・生産設備の変更により変遷する。この幅広く時間的に変化する知識を少数の知識工学技術者で継続的にカバーすることは、人的にも時間的にも甚だ困難であり、製品技術者、生産技術者と生産現場の作業者の領域専門家自身で、知識ベースの開発と保守が可能でなければならない。

日本語で知識の記述が可能

領域専門家は、固有技術には熟知していても、コンピュータに不慣れなために、知識が単に日本語表記されるということだけでなく、日本語で知識を自由に記述できることが必須の要件である。

実用的な機能と高速な推論

FAにおける診断、制御、計画、設計の幅広い適用分野をカバーするのに必要な機能を、確立された技術で、秒単位の制御が可能で高速な推論処理を実現することが重要である。

良好な移植性と既存システムとの統合性

導入及び運用の点では、パソコン・マイコンからメインフレームまでの既存のハードウェア環境と、既に多数のシステムが存在しているソフトウェア環境とに適合し、種々のマシン上で利用可能で、既存システムとの統合性が良いことが要望されている。

同一のユーザインタフェースと容易な操作性

同一の技術分野が大規模にも小規模にも使用されて、同様なエキスパートシステムが適用箇所によってはメインフレーム上でも、パソコン・マイコン上でも動作しなければならない。現状のエキスパートシェルは動作するマシンの

レベル毎に異なり、領域専門家も知識工学技術者も動作するマシン毎にエキスパートシェルを熟知する必要があるというボトルネックがある。このために、メインフレームからパソコン・マイコンまでをカバーし、同一のユーザインタフェースで、操作性が良く、既開発の同種のエキスパートシステムの知識あるいは知識の枠組みが利用可能なエキスパートシェルが要望されている。

## 3 MIND-1の開発思想

本エキスパートシェル MIND-1は、FA分野の実状を考慮し、次のような点を開発のポイントとした。

知識表現と開発作業の分担

領域専門家自身で知識ベースの開発・保守ができるためには、領域専門家が日本語で文章を書くような手軽さで、知識を記述できることが重要である。日本語文で知識の記述に馴染みのあるルールを採用し、ルールとルールに使用される語句に着目した。即ち、知識はルールと語句という2段階の知識から構成され、知識工学技術者が用意した語句を使用し、領域専門家がルールの作成・チューニングを行えば、領域専門家自身で知識ベースの開発・保守が可能であり、知識工学技術者と領域専門家との開発作業の分担が可能である。

また、語句にオブジェクト指向を導入し、ルールの知識表現能力を補完し、階層的知識の表現を可能とし、語句をドメイン毎に集め、再利用を可能にする。これは、ドメインシェルへのひとつのアプローチであると考えている。

推論機能と推論速度

実用的な機能として、前向き、後向き推論以外に、制御で有用なファジィ制御を実現するために、ファジィ推論を提供する。ファジィ推論の導入では、ユーザがファジィを意識することなく、自由に知識を記述できる推論方式を実現する。これらの推論を秒単位の制御が可能にように、Rete アルゴリズムを用いて高速化を行う。

エキスパートシェル開発言語と知識表現言語

メインフレームからパソコンまでサポートされているC言語によるエキスパートシェルの開発と、知識ベースのC言語へのプログラム変換により、良好な移植性と既存システムとの統合性を実現する。

使い易い専用知識エディタの開発

領域専門家が、操作に煩わされることなく、知識を抽出し、知識のチューニングに集中できる支援環境として、操作が容易で判り易い知識エディタを提供する。

開発環境と運用環境の分離

同一のユーザインタフェースと操作を備えた開発環境と、メインフレームからパソコンまでの種々のマシン上での運用環境とを分離可能にし、エキスパートシステムの開発効率の向上と運用コストの低減を計る。

## 4 MIND-1の概要

開発したエキスパートシェル MIND-1の構成、機能概要、特長について述べる。

本エキスパートシェルは、オブジェクト指向を導入し、日本語による知識記述を可能にし、良好なユーザインタ

Expert Shell MIND-1:Outline

Masashi NIWA, Hisamori TANIGUCHI, Masaki YAMAMOTO,  
and Norihisa FUJII

NIPPONDENSO Co.,Ltd.

フェースと容易な知識の入力手段を備え、移植性、他システムとの統合性に優れたエキスパートシェルである。

MIND-1の構成を図1に示す。MIND-1は、開発環境と運用環境とに分かれており、二種類の知識エディタ、知識コンパイラ、推論エンジンから構成されている。

これらの機能の概要と特長を、以下に示す。

1) 知識表現

知識表現は、ルールと語句から構成されている。

・ルール表現

ルールの前件部・後件部は論理積で結ばれた複数の日本語文で構成され、各日本語文は先頭の語句を修飾する語句が順に続く単文である。

・語句表現

語句をC言語にオブジェクト指向を導入した専用の記述言語で記述し、語句の持っている概念、属性等の階層的な知識の表現が可能である。

2) 推論

・ファジィ推論

ファジィ制御に対応可能な推論の実現が重要であると考え、従来の推論(非ファジィ推論と呼ぶ)とファジィ推論とを前向き推論の枠組みの中で融合した。同一ルール中に、ファジィ表現と非ファジィ表現とを同一のレベルでの記述を可能とし、前件部のファジィ表現の成立度合いを後件部のファジィ表現に与えるという推論方式を実現した。ルールの実行に関しては、ファジィ制御の特長を活かすために、デフォルトでは図2に示すような方式にした。後件部にファジィ表現があるルールに関しては、ファジィ制御の方式に従い成立したルールは全て発火されるのに対して、後件部が非ファジィ表現からのみなるルールは、通常の前向き推論と同様に競合解消されたルールのみ発火されることにした。

・前向き推論と後向き推論

前向き推論における非同期入力の受付と各種の競合解消戦略を有し、後向き推論機能の呼出が可能である。

後向き推論では、説明機能、ゴールに近い質問の優先、不必要な質問を抑制する機能を備えている。

3) 知識エディタ

ルールと語句の2レベルの知識表現に応じて、2種類の知識エディタがあり、X-WindowとC言語で開発した。

・語句エディタ

語句の階層関係の管理、バージョン管理等のほぼSmalltalk-80のブラウザ相当の機能を有する。

・ルールエディタ

ほぼ単語に対応した適切な語句のみを表示し、マウスによる選択でルールの作成が可能である。

4) 知識コンパイラ

Reteアルゴリズムに従い、全ルールをC言語のプログラムに変換し、ルール全体と語句との整合性をチェックする。

5 性能評価

本エキスパートシェルとOPS83を同一の適用事例に適用し、性能評価を行った結果を図3に示す。本エキスパートシェルは、オブジェクト指向のメソッドサーチのオーバーヘッドに係わらず、OPS83よりも高速であるという結果が得られた。これは、OPS83のワーキングメモリ上のエレメントのタイムタグの管理等とReteネットワークのノードの単位が細かいことによるオーバーヘッドと、MS-CコンパイラとOPS83のコンパイラのコード生成の差等が要因であると思われる。

6 おわりに

FA分野で有効に利用されることを目指し、特長あるエキスパートシェルMIND-1を開発した。

今後は、本エキスパートシェルを、実際に応用し、機能、ユーザインタフェースの洗練化とともに、更に知識獲得、エキスパートシステムのユーザインタフェース作成支援を計る予定である。

参考文献

- [1] 山本、谷口、藤井、丹羽: エキスパートシェル MIND-1 — 知識表現とユーザインタフェース — 第39回情報処理学会全国大会論文集, 1989

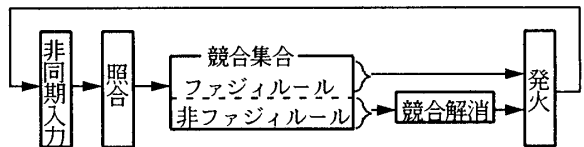


図2 ファジィ推論の融合

	推論時間 (s)	ルール当りの推論時間(ms)	推論速度比
MIND-1	28	7.8	2.1
OPS83	58	16.1	1.0

測定条件: PC9801VX(10MHz)  
MS-DOS(Ver.3.1)  
100回のデータ入力による推論時間  
36ルール  
MS-Cコンパイラ

図3 MIND-1の性能評価

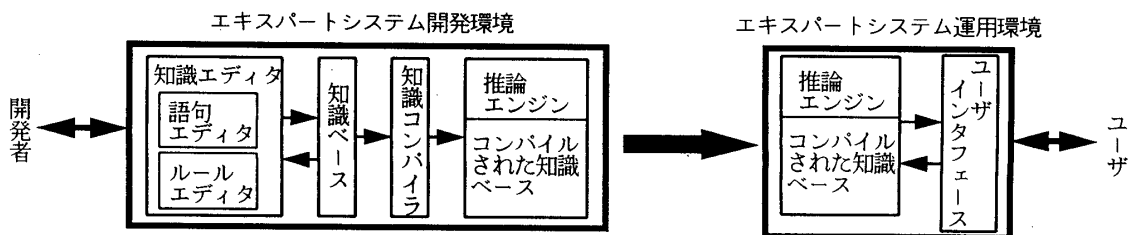


図1 MIND-1の構成