

エキスパート・システム—研究動向と技術的課題—

Expert System : Research Trend and Technical Problems

上野 晴樹 東京電機大学理工学部経営工学科*
Haruki UENO

*Dept. of Systems Engineering, College of Science and Engineering, Tokyo Denki University.

1986年5月15日 受理

Keywords : expert system, artificial intelligence, knowledge-based system, knowledge.

1. は じ め に

エキスパート・システム (expert systems) は、問題領域の専門知識を利用して推論を行い、専門的に高度な問題の解決に関して、エキスパート (専門家) と同等の能力を持つ知的問題解決システムをいう。典型的なエキスパート・システムは、医学診断支援システムにみられるように、特定の問題領域の専門家から獲得された知識 (domain knowledge) を利用して、その分野についての非専門家の意思決定を支援するためのシステムであり、人工知能における最も期待の大きな応用分野である。

この種のエキスパート・システムの応用分野は、医学診断、機器やシステムの故障診断、法律相談、政策決定、性能判定、能力評価などのような分析型問題と、CAD、システム設計、プログラム設計、VLSI設計、薬品設計などのような設計型問題とに大別される。前者は、与えられたデータの分析に基づいて想定される仮説の中から最も可能性の高いものを選択するという問題であり、後者は、与えられた要求を最もよく満たすようなシステムを部品の合成によって生成するという問題である。両者は対照的問題であるが、共通な点は、いずれもその問題領域のエキスパートの知的能力を必要とすることである。ただし、これらの問題は知識表現や問題解決のアプローチにおいて、相互に関係が深く、特にいわゆる深い知識 (deep knowledge) に基づくシステムにおいては区別することの意

味が少なくなるか、もしくは困難になると考えてよいであろう^[28]。

エキスパート・システムは、1965年にスタンフォード大学のFeigenbaumらによって開始された化学構造式同定システムDENDRAL^[3]に端を発することは知られているが、主として米国の大学や研究機関において1970年代に盛んに研究開発が行われ、80年代になってそれらの成果が産業界へ波及し始めた。今日では、世界的に先端技術の代表のように認められ、企業化も急速に行われていることは、周知の事実である。しかしながら、実用システムの開発が盛んな一方で、現場で歓迎されると期待されていた医学エキスパート・システムがいずれも研究室レベルでの実験システムの域を抜け出られずにいるなど^[25]、エキスパート・システムの実用化が当初予想されていたより容易ではないことが明らかとなり、見直しが行われた結果、どんな問題が容易でどんな問題が困難かということや、学問的研究として取り組むべき課題などが認識されるようになった。ただし最近では、エキスパート・システムの意味や範囲が多少不明瞭になっている。エキスパート・システム開発ツールが比較的容易に入手できるようになり、これを用いた簡単な知識ベース・システム (knowledge-based systems) をエキスパート・システムと呼んだり、知識ベースによる画像理解システムなどで、専門知識を利用しているか否かを問わずエキスパート・システムと呼ばれることがあることなどが一つの原因であろう。これらは、プログラム構造からみるとほとんど同一である。

ここでは、エキスパート・システムの本来の概念、

歴史的考察、および技術的課題などについて議論し、今後の研究への指針となるようにしたい。

2. エキスパート・システムとは

エキスパート・システムの定義をまず行って、次に基本的概念や構造などについて論ずる。「エキスパート・システムとは、問題領域のエキスパートから獲得された専門知識を用いて推論を行い、専門的に高度な問題を、エキスパートと同等のレベルで解決できる知的システムをいう」と定義できよう。キーポイントを箇条書きにすると、

- (1) 問題領域の専門知識を用いて推論を行うこと
- (2) エキスパートから獲得された知識であること
- (3) 専門的に高度な問題を対象とすること
- (4) 能力がエキスパートと同等であること

などである。つまり、エキスパートでなければうまく解決できないような問題を、エキスパートと同等のレベルで解決できる能力 (performance) をもつことがポイントである。

このような問題領域は、一般に複雑で曖昧性を含んだ不完全な開いた世界 (ill-structured open world) であり、科学的あるいは理論的に取り扱うことが困難である要素を多く含んでいるという特徴がある。エキスパートと呼ばれるほどの専門家は、その問題領域に関する高度に専門的な知識に精通しているばかりでなく、彼ら自身の永年の体験を通して得たいろいろな経験則 (heuristics) を利用して“うまく”推論していると考えられる。一般の人にとっては途方にくれるような難問をいとも簡単に解決してくれる場面に接したと

き、“さすがにエキスパートは違う”とうなされる。また、身近にエキスパートがいてほしいと思うことが少なくない。エキスパート・システムの目的の一つは、人間エキスパートに代わって非エキスパートの問題解決を支援することである。コンピュータ化されたコンサルタントと考えてもよい。意思決定支援エキスパート・システムのことをコンサルテーション・システムと呼ぶことがあるのはこのためである。コンサルタントが相談者に代わって意思決定することがないように、コンサルテーション・システムも支援する立場を守り、最終的意思決定は相談者が自ら行う。コンサルテーション・システムは、相談者に質問を発し、入力された情報を分析し、推論によって得た結論を示し、対策に関する助言を行い、結論の根拠を説明するなど、相談者の意思決定の支援となるいろいろな機能を持つ必要がある。

エキスパート・システムのユーザが専門家である場合もある。この場合には、問題解決に関する作業能率や信頼度の向上などの利点が多い。現代のように専門領域の細分化が進んでくると、一人の専門家だけで関連するすべての領域に関する十分な知識に精通することは困難となっている。このような状況では、複数の専門家の知識を知識ベースに統合して彼の問題解決を支援することが有効となろう。コンサルテーション・システムに限らず、各種設計支援システムなどこの類の応用は少なくない。

では、このようなシステムを実現するにはどんなソフトウェア構造をとることが必要かを考えてみよう。典型的なエキスパート・システムは、Fig. 1 に示されるような構造をもつ。これは、次のような構成要素から成り立っている。すなわち、

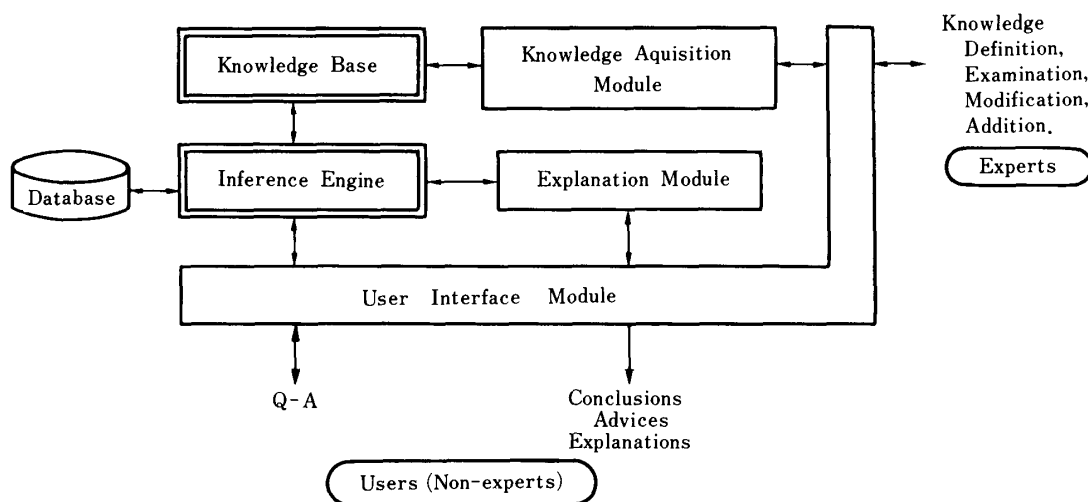


Fig. 1 An overview of an expert system.

- (1) 専門知識を表現しそれを統合的に管理する機構である知識ベース (Knowledge Base)
- (2) 知識ベース内の知識を利用して、推論を実行するための推論機構 (Inference Engine)
- (3) ユーザとの応答をスムーズに行うためのユーザ・インタフェース
- (4) エキスパートから専門知識を獲得し知識ベースを構築する作業を支援するための知識獲得支援機構
- (5) ユーザの要求に応じて、推論で導いた結論の根拠を説明するための推論課程説明機構

などである。エキスパート・システム開発ツールは、これらの機能モジュールを持つソフトウェア・システムである。

3. 歴 史 的 考 察

人工知能の研究者たちが推論中心から知識中心の研究へと方向転換を始めるきっかけになったのが、1965年にスタンフォード大学で研究が開始された前述のDENDRALであることが知られている。これはエキスパート・システムと呼ばれる最初のものであり、構造が未知の科学物質の分子式と質量スペクトル・データを入力して、最も可能性の高い科学構造式を推定するシステムである。科学物質の熱分解に関する専門知識や質量スペクトルと構造とを結びつけ、重要性を評価するための経験則を利用することによって、専門家とほぼ同程度の同定能力を実現できたという。DENDRALプロジェクトは現在でも続いており、NMR (核磁気共鳴) にも応用されている。このDENDRALはプロダクション・システムとして知られているが、最初からそのような形態をとっていたのではなく、試行錯誤の結果としてたどり着いたようである。つまりシステムの問題解決能力を高めることに焦点をしばってプログラムの改良などの努力の過程で、知識表現をプロダクション・ルールで統一することにより、理解しやすく、変更や拡張が容易となり、推論制御メカニズムも簡単な構造となるなど、柔軟性に富んだシステムが実現できるという理由によるものであった^[4]。

70年代に入ると、DENDRALの成功に刺激されて、知識型エキスパート・システム (knowledge-based expert systems) の研究が米国の大学で盛んに行われるようになった。MYCIN^[17]、PIP、CASNET^[21]、INTERNIST^[16]、HEARSAY-IIなどのシステムがこのころ開始されたものである。MYCINは感染症の診

断および投薬決定支援システム、PIPは腎臓病の診断支援システム、CASNETは緑内障の診断治療支援システム、INTERNISTは内科全般にわたる診断支援システム、HEARSAY-IIは (エキスパート・システムではないが) 話言葉理解システムである。

この時期の研究開発の重要な特徴は、応用問題指向型の研究であるということであろう。このころは、専門知識の利用によってシステムの能力をいかに高めるかということに重点が置かれていた。そのために、専門知識をどんなモデルで表わすか、これと組み合わせでどんな推論制御機構を適用すべきかということや、コンサルテーション・システムとしていかなる機能を持たせるべきかということなどに、研究の焦点があてられたといえる。また、Fig. 1に示されるように、知識ベースと推論機構を分離するというソフトウェア構造が採用されるようになったことが、大きな特徴であった。

たとえば、MYCINは最初からすべての知識をIF-THEN型ルールで表現するプロダクション・システムとして設計されたが、ここではさらに、コンサルテーションに必要なHOWやWHYなどの推論課程に関する説明機能や助言機能が組み込まれた。これらは、エキスパート・システムはユーザの意思決定を支援すべきものであり、代行すべきものではないというフィロソフィの現れである。このMYCINは実際の医学データに基づくテストではかなり高い診断能力を示したが、現場のユーザには受け入れられるまでには至らなかった。また、このシステムが初めて採用したあいまいさを取り扱うためのCF (確信度係数) とその評価法やプロダクション・システムそのものの長所・短所などについて、さまざまな議論がまき起こった。

要約すると、MYCINのようなプロダクション・システムの構造では、知識表現、能力、推論速度などにおいて、複雑な問題への応用が困難であり、あいまいさの取扱い方にも不自然さがある^[2] というものであった。これらの議論はその後のシステムの設計へいろいろな形で取り入れられ、歴史上の重要なエキスパート・システムとして、批判派を含めて高く評価されている。また、MYCINは代表的な後ろ向き推論型プロダクション・システムであるが、このタイプの推論は原理的に診断や評価に適しているので、現在でも応用を限定してより汎用な推論制御の一部として使われている。

さて、これらのシステムで採用された知識表現モデルは、MYCINがルール・モデル、PIPとINTERNISTがネットワーク・モデル、CASNETが因果ネットワー

ク・モデル、HEARSAY-IIが黒板モデルなどである。これらはいずれも特定の応用分野のための専用システムとして設計された。したがって、知識表現形式や推論制御の方法に、それが対象とする問題そのものからくる特徴や、それを開発した研究者がその問題をどう理解したかということ、あるいは好みなどが滲み出ている。人工知能システムが人間の知能のモデル化という側面を強く持っていることから、これは当然のことであるといえる。これらの研究成果はそれぞれ学問的に新しい知見をもたらした、その後のこの分野の研究に多大の影響を与えた。しかしながら、いずれも実験システム以上のものではなかったといつてよい。

70年代後半になるとこれら個別的応用指向のエキスパート・システム開発の経験をもとにして、汎用化された知識表現言語が開発され始めた。つまり、上述のシステムはいずれも一種の応用プログラムとして設計されたものであったが、個々の問題解決のために工夫された知識表現形式および推論制御技法は、それらが対象とした問題に類似の別の問題にも応用できるわけであり、かつ性格的に類似の問題が少なくないことが、これらのケース・スタディを通して明らかとなってきたからである。別な表現を使えば、それぞれの問題の選択の時点で、このようなことがすでに念頭におかれていたといえる。AI研究は一種の実験科学であると指摘されるのは、これが主な理由である^[9]。つまり、適切な問題を選択し、その解決を通して一般性の高い原理を究明していくのが、未だ完成されていない研究領域における適切な研究のあり方であり、これまでの人工知能の歴史である^[14]。

さて、これらの研究を通して、ルールやネットワーク、あるいはブラックボードによる知識表現と推論制御の普遍性と有用性がある程度確認されたが、汎用ツールに大きな影響を与えたモデルあるいは理論としては、このほかにMinskyのフレーム理論^[13]、一階述語論理、意味ネットワークなどがある。

70年代後半になって開発された知識表現言語の主なものは、MYCINの汎用化版であるEMYCIN^[23]、CASNETでの経験をもとにルール・モデル化されたEXPERT、HEARSAY-IIを汎用化したAGE^[15]、HEARSAY-III、およびフレーム理論をもとに設計されたFRL、KRL、UNITS^[19]などがある。このほかにルール・モデル（プロダクション・システム）として、OPS、ROSIE、RITAなどが知られている。

わが国で開発された初期の知識表現言語の例としては、MECS-AI（プロダクション・システム）、COMEX^[20]（簡易フレーム・モデル）、FMS^[26]（フレ

ーム・モデル）などがあるが、これらは70年代後半から80年代前半に開発されたものである。ヨーロッパで開発された知識表現言語にPROLOGがある。これは一階述語論理に基づいて設計された言語であるが、知識表現言語としても利用されている一方、LISPと並ぶ人工知能用の汎用プログラミング言語として普及される傾向にある。なお、これらの言語の多くはエキスパート・システム開発言語として設計されたが、フレーム型言語のように、汎用性の高いことから、画像理解や自然言語理解あるいはそのほかに応用されているものもある。

知識表現言語が提供され始めたことによって、大きな変化が起こり始めた。これらの言語を利用して比較的簡単にさまざまな応用エキスパート・システムが試作できるようになったことである。その中でも特にプロダクション・システムは比較的修得が容易であり、使いやすく、簡単な応用問題には十分な能力を持っていることがわかるにつれて、医学診断、機器、システムの故障診断や営業活動支援などに、それまで人工知能はリスクが大きすぎるとして手を出さなかった産業界が、次々と応用し始めた。

このようにエキスパート・システム技術が発展し普及するようになった原動力として見落としてならないのが、米国におけるARPANET、TYMNETなどの全国的コンピュータ・ネットワーク・システムおよびNIH（米国国立健康研究所）の支援によるSUMEX-AIM（スタンフォード大学に本部があるAIの医学応用のための共同利用コンピュータ施設）である。これによって、地理的に遠く離れたツール開発グループと応用システム開発グループとがお互いに協力して研究開発プロジェクトを推進できた。たとえば、EXPERTの開発を行っているラッガース大学とこれを利用してリユーマチ診断システムを開発しているミズーリ大学とは、2000kmも離れている。

エキスパート・システムの研究開発においては、コンピュータ・サイエンティストとその問題領域のエキスパートとの協力が不可欠であるとともに、いろいろな研究グループが常に交流して活発な意見やアイデアの交換をすることが重要であるが、上記の環境がきわめて有効に機能したことは明らかである。70年代のエキスパート・システムの研究成果が医学に片寄っているのは、これが主な理由であると考えてよい。なお、NIHと並ぶ資金援助はARPAであったようであるが、これに関連した研究成果は、その性格からして公表されていないものも少なくないので余計に医学応用が目立ったという事実もある。

80年代になると、この傾向に加速度がつき、それま

で大学の研究室で行われていた研究成果を米国の産業界が先を争うように導入するようになった。これに対応して人工知能応用に的を絞ったベンチャー・ビジネスが出現し始めた。これらは大学のAI系教授によって設立されたものである。一方、わが国では82年に発足したいわゆる第5世代コンピュータ開発計画と推進機関としてのICOTの設立が契機となり、大学や産業界におけるエキスパート・システム研究開発が盛んとなり、今日の隆盛に至っていることはよく知られるとおりである。ただし、研究の歴史、研究実績、研究者、専門技術者および教育研究環境などに関して、日米間の格差は未だかなり大きいのが実情である。

これらの変化に伴って、エキスパート・システム研究開発環境のほうも格段に改善されてきた。すなわち、LISPマシンに代表される、いわゆるAIワークステーションの出現と、マイコン用から大型汎用コンピュータ用に至る各種のエキスパート・システム開発ツールの提供は、それまでの高価な共同利用型AIコンピュータより優れた環境を各研究者レベルにもたらし、どこでも手軽にエキスパート・システムの開発が可能となった。

以上、簡単にながめてきたように、エキスパート・システムの分野は10年前とは大変な違いがある。実用性のあるエキスパート・システムもR1 (XCON)^[12]など、少しずつ現れており、うまく問題を選べば実際に役立つシステムの実現も可能であることが実証されつつある。しかし、一方では基本的かつ重大な壁に直面していることも事実である。プロダクション・システムを使って解決できる問題が多い反面、当初、最も応用事例の多かった医学診断治療支援の分野において、未だに現場で実際に利用されていないことに見られるように、当初の見込みに反して次々に困難な問題が現れ、解決の糸目をさがすために悪戦苦闘しているケースが少なくないのが現状である。これらは、簡単なデモ・システムと実用システムとの間には、天と地ほどの差があることを示している。この問題については、後で検討する。

4. 分析型と設計型

エキスパート・システムが対象とする問題は多種多様であるが、おおむね分析型問題 (analysis-based problems) と設計型問題 (synthesis-based problems) に分類できるようである。ただし、両者の境界は必ずしも明瞭に分割できるわけではない。ここでは

一応この二つに分類できるものとして、両者の間の相違点について議論しよう。この違いは非常に重要であり、知識表現モデルおよび推論制御技法の選択に大きな影響を与えるだけでなく、実現の困難さが大きく異なる。

分析型意思決定問題は、あらかじめ設定された仮説の集合の中から、与えられたデータを最もよく説明する仮説を、データの分析に基づいて選択して結論とする類の問題である。たとえば、医学診断支援システムでは、あらかじめ設定された仮説 (病名) の集合の中の一つ (あるいは複数個) を、患者データの分析に基づいて選択 (診断) するという手段がとられているので、典型的な分析型意思決定の問題に入る。これと類似の問題としては法律相談、故障診断、政策決定、能力評価や設計支援などがあげられる。ここでいう設計支援とは、実際の設計は人間が行うが、手段の選択、パラメータの決定、設計もれの検出、不完全性の検出および助言などを指している。

設計型意思決定問題は、一定の条件 (拘束) のもとで与えられた要求を満足する最適 (または妥当) なシステム構成を生成する類の問題である。一般に設計では複数個の要素 (部品) を組み合わせて、目的とする一つのシステムを構築するので、あらかじめ仮説として解の候補のすべてを設定しておくことはできない。選択型問題はアナリシスが中心であり、設計型問題はシンセシスが中心である。たとえば、プログラム作成は典型的な設計型問題である。ここでは、与えられた要求事項に基づいて設計仕様を作成し、これに基づいてプログラム構造 (データ構造およびアルゴリズム) を設計し、これをプログラム言語で記述するというプロセスがとられる。複雑で大きな問題になると、この各段階で設計者のエキスパタイズが利用される。ただし、作られたプログラムは想定されたコンピュータ環境下で与えられた要求どおりに働く完全なものでなければならない。つまり、設計の各段階では設計者の経験則や趣向に基づいた意思決定 (これは論理的に不完全なものである) が入るが、問題解決の結果としての製品は完全なものでなければならない。

設計型意思決定の問題には、スケジューリング、LSI設計、コンピュータ・アーキテクチャ設計、電子回路設計、プラント設計、実験計画、薬品設計、構造物設計やいわゆるCADが含まれる。医学における治療計画やCAIにも設計的色彩がある。

分析型意思決定問題の特徴は、選択すべき仮説の集合に合わせて問題空間 (problem space) が決定できるので、モデルの作成が容易なことである。典型的な方

法は、所見と仮説とを結びつける評価規則を知識ベースとして準備しておき、与えられたデータに基づいて各仮説に対する評価値を求め、値の最も大きな仮説を結論として決定するというやり方である。AND/OR木でモデル化できる場合が多い。

これに対して、設計型意思決定問題では、極端に言えば無限に近い仮説が存在し得るので、問題空間が非常に大きくなり、モデルの設定が困難である。ただし、設計は部分的に見ると選択で置き換えられるので、拘束 (constraints) の強い問題ならば、分析型問題の延長として取り扱うことができよう。DENDRAL, HEARSAY-II, R1などはこの類のシステムである。また、分析型問題では各仮説は記号で現すことができるので比較的簡単であるが、設計型問題では、設計対象の構造的モデル表現が必要となる。複雑なシステムのモデル表現およびその合成などの操作はむずかしい問題であり、これまでに開発されたエキスパート・システムの大部分が分析型問題を対象としたシステムであって、設計型問題は未だ簡単な例が実現されているにすぎない。

設計型問題へのアプローチは、さらに2種類に分類できるようである。第1のアプローチは、一定の拘束条件の下で、与えられた要求を満たすシステムを合成するというやり方である。合成に基づく設計といえよう。要求がどんな形で与えられるかによって異なるが、与えられた要求を満たすようなシステムを、利用可能な部品をうまく組み合わせながら合成していくというやり方がその典型である。設計仕様の作成は、この種の代表的問題であると思われる。また、要求として明確な形で与えられる形態とは異なるが、前述の、DENDRAL, HEARSAY-II, R1などもこの範ちゅうに含まれるエキスパート・システムである。すなわち、DENDRALは質量スペクトルのデータの中に、HEARSAY-IIでは音声信号データの中に暗黙の要求が含まれており、R1では注文表が要求を現していると考えることができる。いろいろな実現法が考えられようが、このような表現の要求から明確な要求を引き出す仕事 (推論) を問題解決における最初のステップとすれば、スッキリとした推論制御の構造となろう。ここでは、専門家のエキスパートが利用される。次のステップが合成の処理となる。このように考えると、画像理解、音声理解、自然言語理解なども、これと類似の設計型問題であるといえる。

第2のアプローチは、マクロな表現を詳細な表現へ展開 (もしくは分解) することによってシステムの設計を行うというやり方である。展開に基づく設計といえ

よう。厳密な表現形式で記述された仕様書が与えられて、一定の拘束条件の下で、具体的システム設計に展開していくという設計のやり方がこの典型である。たとえば、知的VLSI設計システムとして、与えられた仕様記述から、演算モジュール、論理回路、さらには電子回路へと、次々に自動的展開を行って詳細設計をするというアプローチは、この例である。自動展開処理において、専門家のエキスパートを利用した推論を適用して、無駄な探索をできるだけ避けるような工夫が重要となる。

第1のアプローチは人間主導型であり、第2のアプローチは機械主導型であるともみることができる。複雑な問題に対する本格的な設計支援システムにおいては、両方のアプローチが組み合わされる必要があると思う。

なお、設計型問題と分析型問題を明確に区別することは必ずしもできない場合がある。それは、合成に基づく分析 (analysis by synthesis) によって診断を行うという、高度な診断法がこの例である。この場合にも、与えられたデータに基づいて推論を行い対象のモデルを合成するという方法が採られる。診断は、このモデルを評価することによって行われる。したがって、高度な問題解決は設計型問題として取り扱われる必要があるはずである。

5. 技術的問題

エキスパート・システムに関する課題はいろいろあるが^{[27], [33]}、ここではその中で技術に関する問題、特にシステムの能力に関連する問題に絞って議論してみる。これは、エキスパート・システムの最も重要な問題が、システムの問題解決能力をいかにして十分に高めるかということであるからである。また、最も基本的な知識表現モデルであるルール・モデルを中心に議論する。

さて、問題領域のエキスパートから獲得された専門知識を利用することによって、高度な能力が初めて達成されることは前述のとおりである。このためには、エキスパートはどんな知識を持っていて、それを問題解決のどの時点でどのように使っているかを詳細に知らなければならない。知識表現および推論制御のモデルは、これらが十分にわかって初めて決定できるはずである。しかしながら、現在の人工知能は未だこのような問いに明確にこたえることができない。むしろ、基本的なことの多くが未だわかっていないといつてよいであろう。したがって、現在のところでは、困難な問

題を対象として知能の原理を究明するための学問研究と、すでに開発された技術を用いて、見通しのある問題の解決を目指す応用研究とを分けて取り扱っている。

後者においては、システムの能力を高める工夫はおのずと限定される。たとえば、ルール・モデル（プロダクション・システム）を知識表現モデルとして選択すれば、検討すべき課題はいかにしてこのモデルの枠組みの中でシステムの能力を高めるかという問題となる。次の問題は、すでに存在するツールを使うか新しく言語およびツールを開発するかによって異なる。ここでは、より自由度の大きい後者の場合を中心に考えてみよう。

プロダクション・システムは、事実の集合を管理するためのデータベース、ルールの集合を管理するためのルールベース、および推論制御を行うためのインタプリタから構成されている。まず、プロダクション・システムの性格は採用される推論制御技法によって基本的に決まる。プロダクション・システムにおける推論制御技法は、前向き推論、逆向き推論およびこれを組み合わせた双方向推論の三つである。

前向き推論型はモデル合成に向き、データベースに対するルールの条件部のパターンマッチと選択されたルールの行為部のデータベースへの更新の繰り返しが推論であり、その結果としてデータベース内に対象モデルが合成される。OPSが代表的な前向き推論型プロダクション・システム言語であり、R1が典型的な応用エキスパート・システムである。

これに対して、逆向き推論型は仮説の評価に適し、仮説から出発してルールの逆向き連鎖をAND/OR木の形に構築しつつ評価する処理が推論である。EMYCINが代表的な逆向き推論型言語であり、医学診断や故障診断などの分析型問題に應用されている。

双方向推論型では、まず与えられたデータから出発して仮説を設定し、次にその仮説の評価を逆向きに行うという推論法が採られる。この過程で、さらに別のデータが必要となり、それからさらに別の仮説の生成とその評価を行う処理が繰り返される。このようにしてデータベース内に対象モデルを合成することができる。モデルの合成の過程において、部分的な診断が必要となるので、双方向推論型の採用が最も自然であり、比較的複雑な問題の解決にも応用できると考えられる。RITAが代表的な双方向推論型言語である。したがって、対象とする問題の性質に合わせて適切な推論制御方式を選択しなければならない。逆向き推論型を合成型問題に應用することは困難であり、前向き推論型で診断システムを開発しようとすれば無用かつ不自

然なテクニックの活用を要求される。

さて、プロダクション・システムの枠組みの中で、システムの問題解決能力を高めるためには、データベースを構造化して対象世界のモデル記述力を高めること、ルールの表現力を高めること、およびいわゆるメタ知識などの採用や高速実行技法などによって高度な推論制御を実現することなどが考えられる。

ルールの表現力を高めるには、条件部に複雑な述語の記述あるいは変数の使用を許し、結論部に“あいまいさ”を現す確信度などを用いる方法がよく採られる。複雑な条件指定ができれば、ルールの数が少なくてすむとともに読みやすくなる。変数を使えるようにすれば、複数のパターン（条件記述）を一つのルールで表現できるので、問題によっては大幅にルール数が減少するという利点がある。確信度はMYCIN以来さまざまなシステムで採用されているが、便利である反面ルール数が増えたとき知識全体の見通しを悪くし、結局システムの能力の向上の弊害になることが少なくないなどの欠点がある。確信度の影響範囲を局所的なものとするか、これを使わないですむような別の方法を採用する工夫が必要となる。たとえば、MYCINと同類の診断支援を対象としたONCOSINやCOMEXでは、確信度が使われていない。

データベースを構造化し、高度な概念構造を表現できるようにすれば、ルール数が少なくてすむようになるとともにシステムの能力の向上も達成できる。EMYCINのコンテキスト・トリー、HEARSAY-IIのブラックボードなどはこの種の工夫であり、最近ではさらにARTやESHELLなどに見られるようにフレームを用いるケースが増えている。逆に言えば、このような方法を採用しないときは、ルールでカバーしなければならないわけである。

推論制御に関しては、ルールのコンパイル、推論の繰り返しにおける冗長なパターンマッチを少なくするアジェンダと呼ばれる技法、さらには注視点制御の技法などの採用によって、実行速度の大幅な向上が可能となる。OPS5^[10]やARTなどにはこの種の技法が取り入れられている。実行速度はシステム的能力にとって重要な条件であることは、AI用並列処理コンピュータの開発が盛んに行われていることをみてもわかるように、実用性を左右する。

知識表現モデルをルールに限定しても以上のようにいろいろな選択が必要になるが、モデルそのものの設計から着手する場合には、設計者の自由度はさらに増大する。しかしその反面、負担もそれだけ大きくなる。エキスパートの知識構造や推論のパターンが十分

にわかっていればモデルはおのずと決まるはずであるが、これがわれわれに未だよくわかっていない。もし設計した知識表現モデルがエキスパートの知識構造と異なる場合には、知識獲得が困難となるばかりでなく、システムの能力を高めることもむずかしくなる。たとえば、前述のルール・モデルは表層知識表現であり、エキスパートの知識のうちの表面的な部分しか表現できないことがわかり、深層知識の表現や獲得に関する研究が始まっている [6], [29]。フレーム・モデルを用いると、より高度な知識表現と推論制御が実現できる [13], [25], [26], [30]、それでも深層知識に関するわれわれの理解ははなはだ不十分であるので、しばらく時間が必要である。

6. お わ り に

エキスパート・システムは、70年代に主として米国

の大学の研究室内で急速に進歩し、80年代に入って産業界へ波及し始めた。現在いろいろな問題分野でエキスパート・システムの開発が盛んに行われているので、80年代の終わりごろまでには応用例が増えて、実用化もかなり進むものと思われる。ただし、そのためには現在の技術レベルを考慮した適切な問題を選定することが前提である。エキスパート・システムは従来型のプログラムより知的意思決定の処理においてははるかに強力ではあるが、コンピュータ・プログラムであることには変わりがない。つまり、われわれにわからないことのプログラムの作成はいずれにしてもできないわけである。

研究の場合にもそうであるが、特に短期間での実用システムを目的とする場合には、問題をよく理解して、十分に見通しのあるテーマに取り組むことが肝要であると思う。

◇ 参 考 文 献 ◇

- [1] Bobrow, D. G. and Hayes, P. J., (ed.): Special Volume on Qualitative Reasoning about Physical Systems, *Artif. Intell.*, 24, pp. 1-3 (1984).
- [2] Brooks, R. and Heiser, J.: Some Experience with Transferring the MYCIN System to a New Domain, *IEEE TRANS PAM1-2*, No. 5, pp. 477-478 (1980).
- [3] Buchanan, B. and Feigenbaum, E.: DENDRAL and Meta-DENDRAL; Their Applications Dimension, *Artif. Intell.*, 11 (1978).
- [4] Buchanan, Bruce G.: Research on Expert Systems, HPP-81-1 (1981).
- [5] Buchanan, Bruce G. and Duda, Richard O.: Principles of Rule-Based Expert Systems, HPP-82-14 (1982).
- [6] Chandrasekaran, B. and Mitta, S.: Deep versus compiled knowledge approaches to diagnostic problem solving, *Developments in expert systems*, M. J. Combs ed., pp. 23-34, Academic Press (1984).
- [7] Chandrasekaran, B. and Milne, R. (ed.): Special Section on Reasoning about Structure, Behavior and Function, *SIGART*, 93 (1985).
- [8] Davis, R.: Expert Systems; Where Are We and Where Do We Go From Here, *The AI Magazine*, Vol. 3, No. 2 (1982).
- [9] Feigenbaum, E.: The Art of Artificial Intelligence, Themes and Case Studies of Knowledge Engineering, *IJCAI5* (1977).
- [10] Forgy, C. L.: RETE; A Fast Algorithm For the Many Pattern/Many Object Pattern Match Problem, Tech. Rep. CMU, Dept. of Computer Science (1980).
- [11] Hart, P. E.: Directions for AI in the Eighties, *SIGART*, 79, pp. 11-16 (1981).
- [12] McDermott, J.: R1; A Rule-Based Configurer of Computer Systems, *Artif. Intell.*, 19, pp. 39-88 (1982).
- [13] Minsky, M.: Representing Knowledge in Frames, in Winston, P. (ed.): *The Psychology of Computer vision*, McGraw-Hill (1975).
- [14] Newell, A.: The Knowledge Level, *AI Magazine Summer* (1981).
- [15] Nii, H. P. and Aiello, N.: AGE (Attempt to Generalize); A Knowledge-Based Program for Building Knowledge-Based Programs, *IJCAI6* (1979).
- [16] Pople, H.: The Formation of Composite Hypotheses in Diagnostic Problem Solving—an Exercise in Synthetic Reasoning, *IJCAI6* (1979).
- [17] Shortliffe, E. H.: Computer Based Medical Consultations: MYCIN, American Elsevier (1976).
- [18] Shortliffe, E. H., et al.: ONCOSIN; An Expert System for Oncology Protocol Management, *IJCAI7* (1981).
- [19] Stefik, M.: An Examination of a Frame-Structured Representation.
- [20] Ueno, H.: An End-User Oriented Language to Develop Knowledge-based Expert Systems, *Proc. IEEE COMPCON FALL*, pp. 523-529 (1983).
- [21] Weiss, S., et al.: A Model-Based Method for Computer Aided Medical Decision Making, *Artif. Intell.*, 11 (1979).
- [22] Yamada, N. and Motooka, H.: A Plant Diagnosis Method Based on the Knowledge of System Description, *J. of Information Processing*, 7, 3, pp. 143-148 (1984).
- [23] VanMelle, W., Scott, A. C., Bennett, J. S. and Pears, M.: The EMYCIN Manual, STAN-CS-81-885 (1981).
- [24] 石塚満: 曖昧な知識の表現と利用, *情報処理*, Vol. 26, No. 12, pp. 1481-1486 (1985).
- [25] 上野晴樹: メディカル・コンサルテーション・システムの可能性, 第2回医療情報学連合大会論文集, pp. 179-182

- (1982).
- [26] 上野晴樹：FMS：フレーム理論にもとづく汎用知識表現言語，電子通信学会AL82-64 (1982).
- [27] 上野晴樹：知識工学入門，オーム社 (1985).
- [28] 上野晴樹：知識工学の課題，情報処理，Vol. 26, No. 12, pp. 1554-1558 (1985).
- [29] 上野晴樹：対象モデルの概念に基づく知識表現について-深層知識システムへのアプローチ-，電子通信学会 AI86-4 (1986).
- [30] 小川均：フレーム理論に基づく知識表現言語，情報処理，Vol. 26, No. 12, pp. 1497-1503 (1985).
- [31] 大須賀節雄：知識ベース技術の展望，情報処理，Vol. 23, No. 10 (1982).
- [32] 小林重信：プロダクション・システム，情報処理，Vol. 26, No. 12, pp. 1487-1496 (1985).
- [33] 森健一：知識工学への期待，情報処理，Vol. 26, No. 12, pp. 1550-1553 (1985).

著 者 紹 介



上野 晴樹 (正会員)

昭和39年防衛大学校電気工学専攻卒，昭和46年東京電機大学大学院博士過程満期退学，青山学院大学経営工学科講師を経て，現在東京電機大学理工学部経営工学科教授，工博，情報処理学会，電子通信学会，IEEEなどの会員，電子通信学会人工知能と知識処理研究委員会委員長。

人工知能，知識工学，エキスパート・システムの研究を行っている著書：「知識工学入門」，「知識ベース入門」(オーム社)など。