

TM-0490

エキスパートシステムの構築事例
(SHOOTX)

和田慎一，古関義幸

April, 1988

©1988, 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

1. まえがき

電子交換機などの最近の電子装置は、ますます大規模かつ複雑になりつつある。このため、本来の機能が優れているだけではなく、高い保守運用性が必要である。従来からも電子装置の保守運用性を高めるために様々の努力が成されてきているが、最近、これをより良くするため、エキスパートシステム技術の応用が注目されている。本節では、このような背景から開発された、電子交換機故障診断エキスパートシステムSHOOTXについて述べる。

最初に、電子交換機の保守・運用の現状、故障診断における問題点について述べた後、電子交換機の保守の専門家の診断のもとになっている知識について述べる。次に、これらの考察にもとづいて試作した故障診断システムSHOOTXについて、その目的、機能、知識獲得方式、診断方式、知識表現ツール、知識表現方式、及びユーザインタフェースについて説明し、最後に実用化への問題点と今後の研究課題について述べる。

2. 電子交換機の故障診断

2. 1 SHOOTX開発の背景

電子交換機に代表される電子装置は、その社会的重要度から、障害時には速やかに復旧することが必要である。このため大部分が二重化されており、障害時にもサービスの続行が可能である。そして、故障箇所の発見のために自己診断プログラムが実装されている。

しかし、現実には自己診断プログラムの診断能力には限界があり、必ずしもすべての障害の診断は可能ではない。特に、テストパターンから外れた部分の障害、インターフェース部分の障害などは自己診断プログラムによる診断が困難である。このような障害の診断のためには、システム全体を見通して判断できるようなスキルの高い保守要員が必要である。電子交換機は、他の電子装置に比べて寿命が長く、このような要員を長期に渡って確保することは困難である。特に雇用者の定着率の低い国では要員の安定確保は困難である。このため、短いトレーニング期間で、熟練度の低い保守者にも複雑なシステムの保守が可能に出来るような運用保守システムが必要となっている。

2. 2 専門家による故障診断

内蔵診断によって診断不可能な障害に対しても、保守の専門家は何らかの方法で診断を行ない、障害を復旧する。このために保守者との間のインタビューにより得られた故障診断の方法、知識について述べる。

2. 2. 1 専門家の故障診断方式

人間の専門家は、図1にも示すように、基本的に次のような手順で診断を行なっている。

まず、障害が起き、症状が現れた場合、その症状データを分析して故障の原因と考えられる部分を被疑部分として思い浮かべる。次に、この被疑部分を狭くするための手段（ここではこれをテストと呼ぶ）を選択し、実行する。これを繰り返しながら被疑部分を狭めていき、最終的に修理（部品の交換、コマンド投入）をして、症状が無くなったことを確認する。

2. 2. 2 専門家の知識

電子交換機の故障診断を行なうために専門家の持っている知識を整理すると以下のようなものがある。

- (1) 装置の論理的構造の知識（機能、動作、接続関係）
- (2) 装置の診断方法の知識（症状の分析、テスト方法）
- (3) 装置の物理的構造の知識（部品の種類、実装位置、交換方法）

装置の論理的構造の知識とは、装置の各構成部分の機能、動作と論理的接続関係に関す

る知識である。今日の電子交換機のようなシステムの場合、部品交換単位であるプリント基板が、部品のLSI化により複数の機能ブロックを含んでいる。このため保守者は、この知識を通常個別部品（論理ゲート、LSI）のレベルではなく、機能ブロックのレベルで理解し、診断に利用している。

装置の診断方法の知識のうち、症状に関する知識とは、その装置特有の症状の分析方法に関するものである。例えば、内蔵障害検出機能が障害を発見している場合には、交換機から打ち出されるエラーメッセージの詳細情報を基に被疑部分を推定できる。一方、サービスの異常により障害が発見された場合には加入者からの聴取情報などが症状情報であり、その解析方法は異なる。

テストの知識とは装置特有の被疑部分の切り分け方法についての知識である。電子交換機におけるテストの例として、以下のようなものがある。

- ・交換機内蔵のテストコマンドを実行する
- ・同じ症状の出ている装置の分布をみる
- ・装置についているエラーランプなどを見る
- ・二重化装置の系構成を変更し、症状の出力を見る

装置の物理的構造の知識とは、部品の種類、実装位置、交換方法などである。部品の種類によって壊れ易いものと壊れにくいものがある。また、実装位置も壊れ易さを判断する重要な要因である。また、これらの知識は、実際に部品を交換する際に必要なものである。

3. 故障診断エキスパートシステムSHOOTX

3. 1 目的および機能

SHOOTXは電子交換機の保守専門家の知識を内蔵することにより、熟練度の低い人にも保守作業を可能にする、また、熟練した保守者による日常の保守作業を軽減するなどのために開発されている。本システムは症状情報を分析し、障害となっている可能性のある部分を推論し、もし必要であれば被疑部分を狭くするためのテストを保守者に指示し、最後に部品交換などの修理方法を指示する。

3. 2 従来システムの問題点と本システムの方式

人工知能技術の診断への応用はMYCIN[Shortliffe76]を皮切りにして、これまで多くの試みがなされてきた。例えば、電子交換機に対する応用としても、最近[Bult87]を始めとしていくつか発表されるようになった。しかし、そのほとんどは診断の知識をif-then形式のルールで表わし、前向きあるいは後向きの推論を行なって診断を行なうものであった。この方式には以下のような問題点がある。

- (1) 症状の解析の方法の知識、テストの選択の知識、修理方法の選択の知識などの異質な知識をすべて同じルール形式で、混在して書いてしまうために、必ずしもわかり易い記述にはならず、むしろ誤りをおかしやすく、修正も容易とはいえない。また、ルールは定性的判断を記述するには適しているが、部品の交換方法などの手続き的知識を記述するには適していない。
- (2) 推論の途中で、明らかに成立しないことが分かっているようなルールに対しても、パターンマッチングを試みるために、処理速度が遅い。
- (3) 確信度などの値により、あいまいな推論を行なうことができるものがあるが、確信度の解釈の仕方などが不明確なために、その値の決め方、および結果の解釈の仕方が不明確である。

この他マンマシンインターフェイス等にも課題があるが、これらの問題に対処するために本システムは以下のような方式をとっている。

- (1) 構造、部品、症状、テストに関する知識を、ルール形式ではなく、すべて統一されたネットワーク形式で記述している。このため、知識の追加・修正が容易である。また、症状の解析による被疑部分の作成と、テスト結果の解析による被疑部分の切り分けに対して、信号経路に着目した方式を採用している。このため、症状及びテストの知識の簡潔な記述が可能である。
- (2) ネットワーク形式で知識を表現するために、Prolog言語ベースの知識プログラミングシステムPeaceを開発し、使用している。このため、オブジェクト指向などのプログラム方式を利用しながら、手続き的な記述と規則的な記述をPrologのルールとして統一して記述できるので、従来のようにルールベース、対象世界の記述、推論のための知識を別に記述するのではなく、統一したネットワークの形式で記述できる。
- (3) 診断のフェーズに合わせて、アクセスするオブジェクトを制御している。このため、無駄なパターンマッチングが回避される。
- (4) システムと保守者の協調した診断を可能とするため、次のアクション（実行するテスト、交換する部品など）をシステムが決定し利用者に強制するのではなく、常に、可能性のある選択肢を優先度順に表示し利用者に選ばせる方式をとっている。これにより、初心者には優先度の高いものから順に実行をすればよく、熟練者は有効なもののリストの中から自分の判断で実行を行なうことが出来る。
- (5) メニュー入力とグラフィック出力を利用したユーザインタフェースにより、初心者にも使いやすいシステムとなっている。
- (6) 診断におけるあいまいさを、被疑部分の優先度を表わす数値、テストの優先度を表わす数値、および修理の優先度を表わす数値に整理して記述し、それぞれの意味を明確にして取り扱っている。

3. 3 知識獲得

SHOOTXの開発において、知識獲得は実際の保守の経験者（エキスパート）に対してインタビューをすること、および保守者向けの診断マニュアルの検討により行なわれた。当初はインタビューする側（知識エンジニア）に交換機に対する知識が欠けていたために難航した。しかし、回を重ねて、知識エンジニアの側で交換機に関する一般的な知識と、交換機特有の語彙が理解できるようになるにつれ、インタビューも容易になってきた。つまり、インタビューにより、知識エンジニアがその問題領域を理解することが必要であった。

最初は、一般的な診断方式をインタビューにより聞き出す試みが行なわれたが、専門家自身もどのような方式で診断を行なっているのかは説明が出来ず、この方法はうまくいかなかった。そこで、典型的な例題を選んで、その上で保守者がどのように診断をしていくか調べる方法がとられた。この方法により、専門家も自分ならどの様にその問題に対処するかを説明できた。この方法で、一つの例題におけるすべての組合せを調べることににより、それを似たような問題について応用することが容易であった。

次に、すべての問題をカバーするために、症状の出方、障害原因などにより、電子交換機に起こる障害全体がいくつかの種類に分類された。そして、それぞれの分類の中で典型的な例題が選ばれ、それぞれについて保守者の知識が調査された。最後にそれぞれの分類について知識が広げられた。

つまり、知識獲得の方法は例題中心であったといえる。まず、例題について詳しく調べ、次にそれを広げていく方法である。この方式に基づいた知識獲得システムの研究(Explanation Based Generalization)が、最近盛んに行なわれている。

また、当初はルール形式による知識表現が試みられていたが、次第にルールによって自然に表現できないことが明らかになり、その結果、現在のような知識表現形式に落ち着いた。

3. 4 診断方式

本システムにおける診断の流れは図1に表わされる。まず、故障の症状のデータを分析し、故障の原因と考えられる部分（被疑部分）を求める。被疑部分は機能ブロックの集合として表わす。次に、被疑部分に応じて有効なテストを選択し、実行する。その結果を判断することによって被疑部分を狭める。これを繰り返していき、最終的には部品を交換して症状が消えるかどうかを確認する。

3. 5 知識表現ツール

SHOOTXにおける各種の診断知識の表現は知識プログラミングシステムPeace(Prolog based Expert Applications Environment)[Koscki87]を用いて行なわれている。PeaceはPrologをベースとして各種のプログラミングパラダイムを使用できるようにしたものであり、次の特徴を持つ。

- ・Prologの節の集合であるオブジェクトと、その間の関係の表現により意味ネットワーク形式の知識表現が可能。
- ・一般的関係を通したマルチプルインヘリタンスを利用してオブジェクト指向プログラミングが可能。
- ・Prologをベースにしているので、ルールと手続きが同じ形式で記述できる。また、Prologのパターンマッチ、バックトラッキング等の機能が利用可能。
- ・関係を中心とした「関係指向プログラミング」が可能。

3. 6 知識表現方式

本システムは、装置の構造・動作等の設計上の知識と、症状、テスト等の診断上の知識の両方を利用する診断を実現している。そして、これらの知識を統一した意味ネットワーク型の表現により表わしている。

症状、テスト、機能ブロック、部品などをネットワークの各ノードとして表現し、これらの間の関係に関する知識をノード間のアークとして表現している。このネットワークの例を図2に示す。例えば、症状Aに対応する機能ブロックは、信号経路Bに含まれるものであり、機能ブロックA、機能ブロックB、機能ブロックCであることが分かる。

推論の知識の記述は関係の記述あるいは、オブジェクト内のProlog節（メソッド）に記述している。

このようなネットワーク形式を用いた知識表現では、if-thenルールを用いて各症状・テスト結果と被疑とすべき部品の関係を直接的に結びつける表現とは異なり、症状・テストからネットワーク上の関係を通して機能ブロックが導き出される形になっている。このため、知識の記述がわかりやすい。また、各症状やテストに対して記述すべき知識の量が少なく済み、局所的な追加・修正が容易である。

設計上の知識は、機能ブロックレベルの信号の流れとして表現する。図3がその例である。各ノードは機能ブロックを表わし、アークは機能ブロック間を流れる信号を表わす。

より詳細な知識表現方式および推論方式については[和田86]を参照されたい。

3. 7 ユーザインタフェース

本システムでは、次のアクション（実行するテスト、交換する部品）をシステムが決定し強制するのではなく、常に、実行可能な選択肢を優先度順に表示し、利用者には選ばせる方式をとっている。これにより、初心者は優先度の高いものを選択して実行すればよく、熟練者は有効なもののリストの中から判断して選択・実行することができる。

図4の実行画面からわかるように、ユーザインタフェースとしてはマルチウィンドウ、

メニュー、グラフィックスを用いることにより、装置の構成、部品の位置表示、状態表示等をわかりやすくし、操作性を高いものとしている。

4. 実用化への問題点と今後の研究課題

目的と機能をしぼれば、本システムの方式で十分に実用に耐えるシステムが作成可能であると考えられる。しかし、電子交換機のような大規模なシステムに対して、人間の専門家と同等に診断を行なう故障診断システムを実現し、本格的に実用化するためにはまだ、いくつかの問題点がある。これらについて、ここで述べる。

(1) 知識獲得

本システムの場合、他のエキスパートシステムと同様に、知識エンジニアがマニュアルおよび専門家のインタビューにより得た知識をコード化して知識ベースを作成している。この方法では、知識エンジニアがその問題領域の知識をかなり持たないと困難である。したがって、高度な知識エンジニアを必要とすることになり、知識ベース作成のコストが高くなる。知識ベース作成のコストがエキスパートシステムの実現によるメリットよりも小さくなることが実用化の条件であることは言うまでもない。このためには、より一層、知識エンジニアの知識獲得および知識ベースの作成をより容易にする研究が必要である。究極的には知識エンジニア無しで専門家が直接知識ベースの作成ができるような知識獲得方式が望まれる。

(2) 予期しなかった障害への対応

本システムを含めて、これまでの診断エキスパートシステムは知識ベースに記述されていない予期しなかった障害には全く対処できない。つまり、知識ベースにない障害が発見された場合、それに対処出来るようにするためには、常に新しい知識を追加していかなければならない。やはり、今後は[Genesereth84]のように装置の構造と動作から推論を行なう方式を融合していく必要があると考える。

(3) 設計データベースとの結合

故障診断において必要となる装置の構造、動作についての知識は装置の設計時にすでに得られているものである。そして、最近では、それらの情報は、装置の設計もCADプログラムを利用して行なわれるようになってきているためにデータベースとして得られる可能性がある。しかし、現在はそれらのデータベースの形式は診断で利用出来るような形式ではないため利用できない。今後は、これらの設計データベースと診断システムの結合をめざすと共に、装置の設計段階から運用保守までを一貫してサポートするCAE(Computer Aided Engineering)システムの研究を行なう必要がある。

(4) 経験による学習

人間は、経験を積むにしたがってより速く診断を行なうことが出来るようになる。故障診断の問題領域においては、故障部分の推定選択と有効テストの選択において経験による学習が行なわれると考えられる。そして、それらは、それまで知らなかった新しい障害原因やテスト方法を学習する場合と、既知のものの選択基準を学習する場合とに分類される。前者の学習は新しい概念の学習であり、すぐに実現は困難であるが、後者の学習は本システムの枠組みで実現可能と考えられる。

(5) 利用環境

本システムのようなシステムを実用化するためには、それを使用していく環境も同時に整えなければならない。例えば保守要員の配置方法、客先との保守責任の分担なども検討する必要があるだろう。また、付随的な情報のデータベース化も必要がある。例えば、システムが常に最新の実装情報に基づいて診断を行なうためには、実装情報が利用できる形態で用意されており、かつ常に更新されている必要がある。

5. むすび

故障診断システムの事例として、電子交換機故障診断エキスパートシステムSH00TXについて述べた。

なお、本システムの研究の一部は通産省第五世代コンピュータプロジェクトの一環として行なわれたものである。

【参考文献】

- [Bult87] Bult, T., Peacocke, D., Rabie, S., and Snarr, V.: An interactive Expert System for Switching Maintenance, Proc. of International Switching Symposium, pp. 59-65(1987).
- [Genesereth84] Genesereth, M. R.: The Use of Design Descriptions in Automated Diagnosis, Artificial Intelligence 24, pp. 411-436(1984).
- [Koski87] Koseki, Y.: Amalgamating Multiple Programming Paradigms in Prolog, Proc. of International Joint Conference on Artificial Intelligence, pp. 76-82(1987).
- [Shortliffe76] Shortliffe, E. J.: Computer Based Medical Consultations: MYCIN, Elsevier, New York(1976).
- [和田86] 和田, 古関: 故障診断エキスパートシステムSH00TXの知識表現方式, 日本ソフトウェア科学会第3回大会論文集, pp. 209-212(1986).

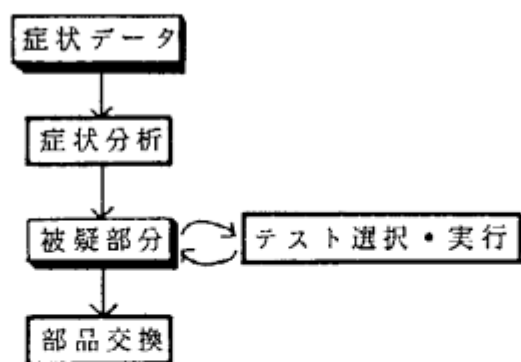


図1 診断の流れ

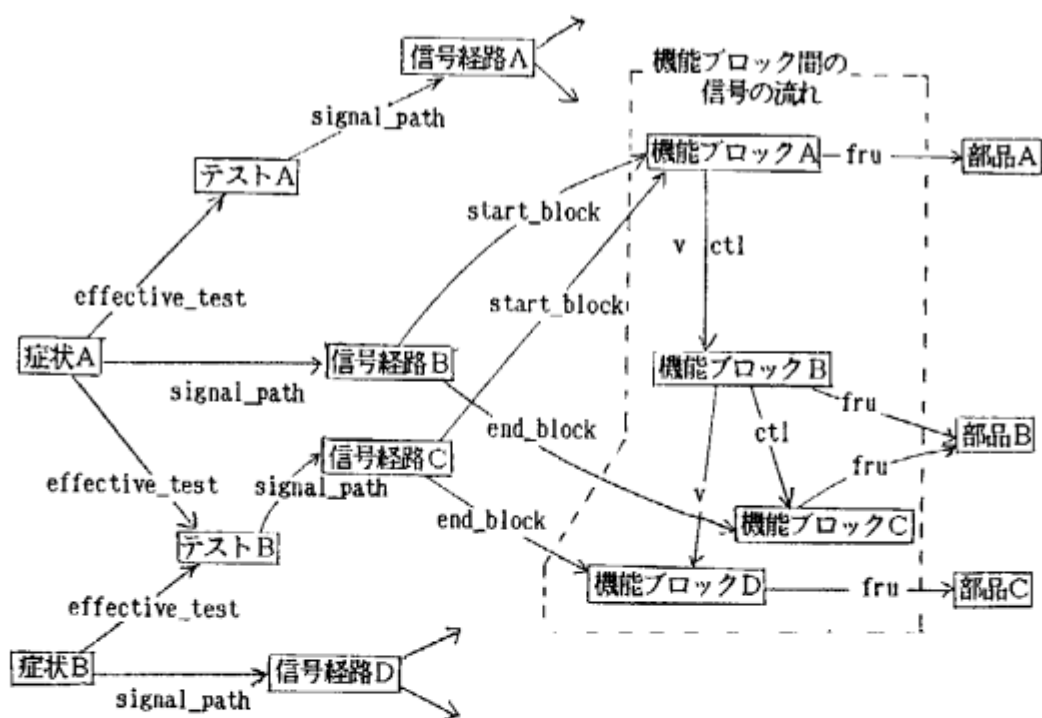


図2 ネットワーク形式の知識の表現

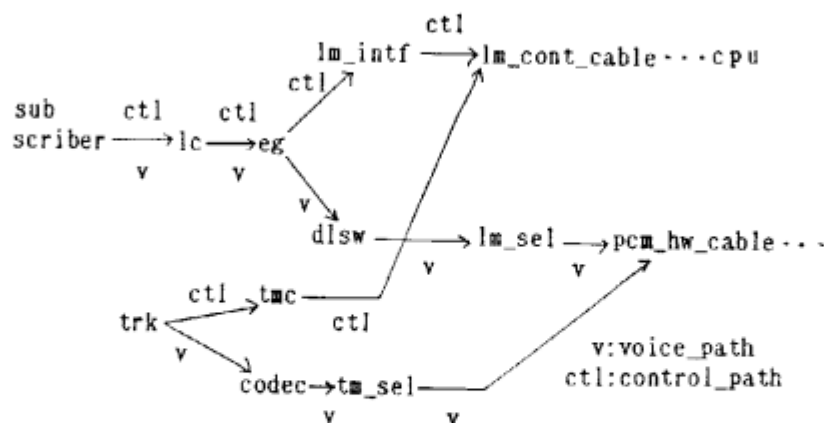


図3 機能ブロック間の信号の流れ

Date	Time	No.	Message	Synopsis
09/16	17:52	A 419	CP01-LOC102	ERROR 0400 8101 8100 (NNTQ)
09/16	18:20	A 419	CP02-LOC100	ERROR 0400 8101 8100 (NNTQ)
09/16	20:03	A 419	CP01-LOC102	ERROR 0400 8101 8100 (NNTQ)
09/16	21:16	A 419	CP01-LOC004	ERROR 0400 C001 8100 (NNTQ)
09/16	22:13	** 043	CP01-LOC004	DIAGNOSIS FAULT 02-402-007
09/16	22:37	A 419	CP02-LOC002	ERROR 0400 A001 8002 (NNTQ)
09/17	01:34	** 043	CP02-LOC002	DIAGNOSIS FAULT 02-202-004
09/17	05:10	A 419	CP01-LOC102	ERROR 0400 8101 8100 (NNTQ)
09/17	06:22	A 419	CP01-LOC004	ERROR 0400 C001 8100 (NNTQ)
09/17	08:48	** 043	CP01-LOC004	DIAGNOSIS FAULT 02-402-007
09/17	09:35	A 406	CP02-SPC00	ERROR 0C00 8111 8001 (NNTQ)
09/17	10:07	** 042	CP02-TSC00	DIAGNOSIS FAULT 02-001-002
09/17	11:13	A 419	CP02-LOC002	ERROR 0400 A001 8002 (NNTQ)
09/17	12:20	** 043	CP02-LOC002	DIAGNOSIS FAULT 02-202-004
09/17	15:02	A 419	CP01-LOC004	ERROR 0400 C001 8100 (NNTQ)
09/17	16:33	** 043	CP01-LOC004	DIAGNOSIS FAULT 02-402-007
09/17	19:35	A 419	CP02-LOC002	ERROR 0400 A001 8002 (NNTQ)
09/17	21:10	** 043	CP02-LOC002	DIAGNOSIS FAULT 02-202-004
09/17	21:40	A 406	CP02-SPC00	ERROR 0C00 8111 8001 (NNTQ)
09/17	23:45	** 042	CP02-TSC00	DIAGNOSIS FAULT 02-001-002
09/18	02:56	A 406	CP02-SPC00	ERROR 0400 8112 8802 (NNTQ)
09/18	04:29	A 419	CP02-LOC002	ERROR 0400 A001 8002 (NNTQ)
09/18	05:08	** 043	CP02-LOC002	DIAGNOSIS FAULT 02-202-004
09/18	06:49	A 419	CP01-LOC102	ERROR 0400 8101 8100 (NNTQ)
09/18	09:43	A 419	CP01-LOC004	ERROR 0400 C001 8100 (NNTQ)
09/18	11:53	** 043	CP01-LOC004	DIAGNOSIS FAULT 02-402-007

図4 ユーザインタフェース