

故障診断用エキスパートシステム

柳吉洙 志村正道
東京工業大学 工学部

船舶主機関用故障診断システムにおける初期の知識ベースを構築する方法について述べる。本システムはデータベースや専門家が入力したデータから知識抽出用ルールを用いて診断用知識を獲得する機能を有している。知識抽出用ルールは概念間の因果的つながりを記述した一般的なルールであるので、システムが冗長な知識および専門知識に矛盾する知識を抽出する可能性がある。このため、本システムでは専門家との対話によって知識抽出過程を学習する方法を用いている。また、ルールベースには領域に依存する知識と領域に依存しない知識が分離され、制御知識がメタルールによって作成されている。

Expert System for Trouble-shooting

Keel-Soo RHYU and Masamichi SHIMURA
Faculty of Engineering, Tokyo Institute of Technology

This paper describes a knowledge acquisition method in conjunction with a trouble-shooting system for marine engines. Diagnostic knowledge is usually given by a human expert, but it is difficult to obtain a large amount of specialized knowledge from experts or knowledge engineers. In our system, this knowledge is acquired from a data base or input data by employing heuristic knowledge-extraction rules, which find the causal relations between every part of a marine engine. The extracted knowledge is not always appropriate and correct because of errors induced by a generalization process. Therefore, the knowledge is refined with the aid of human experts.

1. まえがき

エキスパートシステムが様々な専門分野で実用化されると同時に、領域における知識の表現および獲得が主な研究対象になっている。本論文においては、船舶主機関用故障診断システムを対象とし、主にその初期の知識ベースを自動的に構築する知識獲得の方法について述べる。知識獲得の方法としては、知識エンジニアが専門家と対話することによって知識を獲得する方法、専門家が直接エキスパートシステムと対話しながら知識ベースを構築する方法^[3]などがある。また初期の知識ベースの構築を容易にするための手法として、ケースデータを統計的に解析することによって専門知識を獲得する方法などが研究されている^{[1], [4], [5]}。しかしながら、いまだに知識獲得はエキスパートシステムの構築において解決すべき問題が多々ある。

ここで述べるシステムは後向き推論および前向き推論をもち、船舶主機関の故障の原因や結果を導くことによってその故障診断を行なうエキスパートシステムである。本システムにおける診断用知識の獲得方法としては、知識抽出用ルールを用いてデータベースや専門家の入力データから診断用知識を抽出するものである。一般に知識抽出用ルールとして特殊な事実間の因果関係の記述を用いるとルール数が膨大になり、システムは実用的でなくなる。逆に一般化されたルールを用いると誤った知識を抽出する可能性はあるが、多くの領域に適用することができる。それゆえ、一般化された知識抽出用ルールを用いて知識抽出を行い、専門家を介してシステムの誤った動作を正す方法について述べる。

2. 知識獲得の方法

本システムが用いる診断用知識は図1のようなプロダクションルールの形式を持ち、以後診断用ルールと呼ぶ。この知識の中で、各フレームの状態の程度すなわち異常度を表す値には normal、abnormal を用い、abnormal の中で特に危険な状態には critical を用いる。

```
<rule>          ::= (if <condition> then <conclusion>)
<condition>      ::= <element> | (<connector1><conditions>)
<conditions>     ::= <condition> | <condition><conditions>
<conclusion>     ::= <element> | (<connector2><conclusions>)
<conclusions>    ::= <element> | <element><conclusions>
<element>       ::= (<frame-ID><slot-ID><value>)
<connector1>    ::= AND | OR
<connector2>    ::= AND
```

図1 抽出する診断用知識の構文則

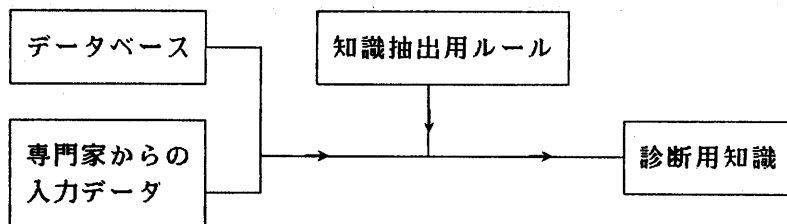


図2 知識獲得の形態

図2は本システムの診断用ルールの獲得方法を示している。一般化された知識を用い診断用知識を獲得することは多くの領域に適用することが可能であるので、知識抽出用ルールとして上位概念すなわち一般化された事実間から診断用知識を獲得するための規則を用いる。例えば船舶主機関において、上位概念は機械類、ポンプ類、タンク類、バルブ類、液体類、部品類に分けられるが、この概念は他の機械領域でも部分的に適用できる。

専門家が入力可能なデータには“～なら～である”のようなルール形式のデータと“～は～の～である”のように事実を表すフレーム形式のデータがある。ルール形式のデータは図3のような機械の動作を表すデータおよび状態を表すデータから構成され、また状態を表すデータは異常状態を表すデータ、正常状態を表すデータに分けることができる。

種 類	例	内 部 表 現
動作	valve を open pumpを start	(valve open) (pump start)
正常状態	冷却水の圧力は 2.0	(cooling-water pressure 2.0)
異常状態	主機関は異常	(main-engine condition ?ac)

図3 入力データの分類

図3で、正常状態や異常状態を表すデータの内部表現はフレーム名、スロット名、値で表されるので、フレームの参照やルールの抽出にそのまま用いることができる。しかし、動作を表すデータの内部表現はスロット名を持たないので、動作を表すデータは状態を表すデータに変換する必要がある。例えば、“starting-air-valveを open する”はバルブを open することによってバルブが open 状態になるため、“starting-air-valve state open”に変換する。また、異常状態を表すデータはそれ自身が故障診断用の知識を表しているので、知識を抽出するときに直接このデータを利用することができる。したがって、入力データから診断用知識を抽出するときは、正常状態を表すデータからの知識の抽出についてのみを考えればよい。

3. システムの構成

3.1 データベース

データベースでは機械に関する事実やその階層関係をフレーム形式で表し、16個のスロットを用いている。スロット値には数値と非数値がある。数値のとき最小値を“MIN 数値”、最大値を“MAX 数値”のように表す。なお、値が非数値のときは複数個の値を表すために、連結子としてANDおよびORの使用が許される。

次の例は、船舶の主機関の部品である piston に関するフレームを示している。purpose-of スロットおよび need-of スロット値の中で数値はそのスロット値の有効の度合を表し、度合を以下有効度と呼ぶことにする。システムは抽出する診断用ルールの異常度を決定するときにこの有効度を用いる。

```

(piston (part-of main-engine)
  (purpose-of (((run main-engine) 1.0)))
  (lubricating cylinder-oil)
  (cooling jacket-cooling-water)
  (need-of ((cylinder-oil 0.5)
    (jacket-cooling-water 1.0))))

```

3.2 知識抽出用ルール

データベースに貯蔵されているフレームから診断用知識を抽出するときは各フレーム間の因果関係を見つけ出す知識を用いる。この知識はデータベースからの知識抽出用ルールであり、以後 RULE-F と呼ぶ。例えば、次のようなルールは機械の部品という概念から機械概念や液体概念を結ぶ知識を表している。

```

IF      フレーム名が機械の部品であり
        かつ、スロット名に{part-of},{lubricating},
        {cooling},{state} を持つ
THEN    (F-MAKE (IF (AND (part-ofの値 state run)
                          (OR {(フレーム名 state stop)}
                              {(lubricatingスロットの値 condition ?AC)}
                              {(coolingスロットの値 condition ?AC)} ))
                THEN (フレーム名 condition ?AC)))

```

このルールは、『注目しているフレーム名が機械の部品でありかつそのフレームに part-of、lubricating、cooling、state スロットが存在する場合は、「part-of スロットの値である機械が運転中であり、その部品が停止しているかまたは部品を潤滑および冷却させるための液体が異常状態であれば、部品は異常状態になる」という診断用ルールを抽出する』意味を持っている。なお、?AC はフレーム名の異常度を表す。

専門家が入力するデータは正常状態の記述が多く、このようなデータから診断用知識を抽出する場合には異常状態に対応できる形に変換しなければならない。このため、通常は正常状態の対偶が取られる。例えば、(A run) -> (B run) のようなデータに対する対偶は (B stop) -> (A stop) となる。しかし、このようなルールは診断用知識としては意味があいまいとなり不十分である。したがって、このようなデータから A および B が正常状態 (normal) または異常状態 (abnormal、critical) になるための条件などに関するヒューリスティック知識が必要となる。この対遇のための知識やヒューリスティック知識が知識抽出用ルールであり、以後 RULE-I と呼ぶことにする。入力データから診断用知識を抽出するためのルールの一例を次に示す。

```

IF      条件節や結論節が run 状態を含む
THEN    (F-MAKE (IF (AND (結論節が run 状態) (条件節の逆状態 ))
                    THEN (結論節のフレーム名 condition ?AC)))

```

このルールが持つ意味は、『入力データの条件節や結論節のフレーム名が両方とも run 状態である場合、「結論節のフレーム名が run 状態でありかつ条件節のフレーム名が逆状態であれば、結論節のフレーム名は異常状態になる」という診断用ルール

を抽出する』ことを表す。

3.3 ルールベース

システムは図4のようにルールベースの階層構造を持ち、領域依存知識、領域に依存しない知識、および制御知識に大別して構成されている。知識抽出用ルールは領域依存知識であり、このルールを制御するルールは領域依存メタ知識である。領域に依存しない知識は異常度の決定、検証、有効度の変更などに用いるルールである。また、システムを制御する知識はメタルールである。このようにルールを階層的に表現することは、ルールの探索空間の減少、診断用知識の効率的な抽出を実現するに有効である。

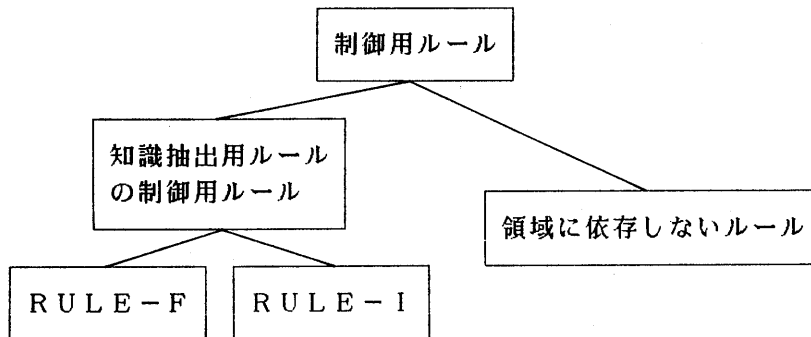


図4 ルールベースの階層構造

4. システムの制御

診断用知識を抽出するためにシステムは、まず専門家との対話の中で RULE-F を起動するか RULE-I を起動するかを決定する。データベースから知識を抽出する場合は、RULE-F を起動し診断用ルールを抽出するが、この際 RULE-F から適用すべきルールを選ぶための競合解消は次のような基準に従うことにする。

- ① 条件節の要素の数が最も多いルールを優先する。
- ② RULE-Fで用いる述語に予め与えられた有効度が最も高いルールを優先する。
- ③ RULE-Fで用いるフレームのスロット名に予め与えられた有効度が最も高いルールを優先する。

専門家が入力したデータから知識を抽出する場合には、まず入力データが事実を表すフレーム形式のデータかまたは動作や状態を表すルール形式のデータかを識別する。データがフレーム形式のデータである場合には、その事実がデータベースに存在するかどうかを調べ、存在しない場合には新しいデータとしてデータベースに追加する。同時に、RULE-F を起動し変更したフレームから抽出すべき診断用知識があるかどうかを確認する。また、データが動作や状態を表すときは、RULE-I を起動する。

次にシステムは診断用ルールを抽出し、抽出したルールの各部の異常度を決定する。また、領域の基本知識あるいは知識ベース内のルールとの矛盾または冗長性について検証を行い、この診断用ルールに対する専門家の判断に従い知識ベースに追加する。専門家が誤った知識であると判断した場合には、以後知識抽出過程で誤った知識を抽出しないように有効度を変更される。

4.1 異常度の決定

異常度が決定されていない診断用ルールは次のような三つのケースがある。

- ① 未決定値を含む要素が結論節に存在するルール

② 未決定値を含む要素が条件節に存在するルール

③ 未決定値を含む要素が両方に存在するルール

①の場合、その要素にあるフレームの need-of スロット値の中に、条件節にあるフレーム名が存在するかを調べ、その有効度によって異常度を決定する。有効度が0.3未満であれば normal、0.7以上である場合は critical、それ以外は abnormal にする。存在しないときは、need-of スロット値に適するかを専門家に質問する。

②の場合、その要素のフレームの purpose-of スロット値を調べ、その有効度によって異常度を決定する。

③の場合、その要素にあるフレームの need-of スロット値や purpose-of スロット値の有効度を参照することによって異常度を決定する。

4.2 抽出した診断用ルールの検証

本システムで用いている知識抽出用ルールは上位概念間の記述であるので、誤った診断用ルールが抽出される可能性がある。したがって、抽出した診断用ルールの検証が必要となる。検証は2段階に分け行なわれる。

一次的検証では領域の基本知識に矛盾しないかを調べる。例えば、専門家が入力したデータから次のような診断用ルールが抽出されたとする。

```
I F (fuel-oil pressure (>$ 2.0 ?v))  
T H E N (fuel-oil-booster-pump condition ?AC)
```

これは『fuel-oil-booster-pump を起動し、fuel-oilの圧力が 2.0であるとfuel-oil-booster-pump は正常である』という入力データから抽出した診断用ルールである。このルールは、fuel-oil-booster-pump が異常になるためにはfuel-oilの圧力の制約とfuel-oil-booster-pump の状態が runである条件が必要であるので、誤った知識として検出される。

二次的検証では、抽出した診断用ルールと知識ベース内に存在するルールが矛盾しないか、一方が冗長ではないかなどを調べる。二つのルールを比較するとき、冗長なルールは次のように定義する。

- ・条件節が同じで、結論節が一般化されている場合
- ・結論節が同じで、条件節が特殊化されている場合

連結子を含む節ではANDを含むものが特殊、ORを含むものが一般的である。また二つのルールで、ある対応した要素の値だけが異なり、他の部分が全て同じ場合には両者は矛盾すると定義する。

5. 有効度の変更による学習

5.1 RULE-1 起動中の有効度の変更

RULE-1 から抽出した診断用ルールに対する専門家の判断は、“GOOD”、“PARTLY-GOOD”、“BAD”の3種類が与えられる。

抽出した診断用ルールが誤った知識であれば専門家の判断は “BAD”である。例えば、次のような例からルールAが知識ベース内にありルールBが抽出されたとすると、以後類似した診断用ルールを抽出する場合は abnormal が得られるようにしなければならない。したがって、診断用ルールの異常度を決定するときに用いるフレーム F₂ の need-of スロットの値 F₁の有効度を0.7未満に変更する。

ルールA : (IF (AND (F₁ S₁ V₁) (F₂ S₂ V₂)) THEN (F₂ condition abnormal))

ルールB : (IF (AND (F₁ S₁ V₁) (F₂ S₂ V₂)) THEN (F₂ condition critical))

ルールC : (IF (F₃ condition critical) THEN (F₄ S₄ V₄))

ルールD : (IF (F₃ condition abnormal) THEN (F₄ S₄ V₄))

ルールCが知識ベースに存在しルールDを抽出した場合、以後 critical を含む診断用ルールを抽出するようにフレーム F₃ の purpose-of スロット値 F₄ の有効度を増加させる。

抽出した診断用ルールが正しいときは専門家の判断は "GOOD" である。したがって、知識ベース内に冗長なルールが存在するかどうかを確認する。上記の例のルールBはルールAの特殊な場合であるので、知識ベース内のルールAを削除し、抽出したルールBを知識ベースに挿入する。

診断用ルールに一部誤りがある場合は専門家の判断が "PARTLY-GOOD" となる。したがって、システムは専門家に同ルールを部分修正するように要求する。この修正されたルールを新しいルールと呼び、このルールがどのように修正されたかを調べることによって学習を行う。上の例で、ルールA、Cを新しいルール、ルールB、Dを抽出したと仮定すると、フレーム F₂ の need-of スロット値 F₁ の有効度を減少させ、フレーム F₃ の purpose-of スロット値 F₄ の有効度を増加させる必要がある。

一般的に知識獲得過程を改良するため、抽出した診断用ルールの結論節を修正する必要がある場合は need-of スロットの有効度を変更し、条件節の場合であれば purpose-of スロットの有効度を変更する。

5.2 RULE-F 起動中の有効度の変更

RULE-F の実行によって抽出された診断用ルールに対する専門家の判断は "GOOD" あるいは "BAD" である。一個のフレームから抽出した診断用ルール中で、専門家の判断が "BAD" であるルールが存在することは適切でない知識抽出用ルールの適用により生じる。したがって、競合解消過程で二番目の選択だったルールを実行し、新たに抽出した診断用ルールに対して専門家の判断を聞く。二つの知識抽出用ルールの実行結果を比較して二番目のルールの実行結果が良い場合は、以後知識を抽出するときは二番目のルールを先に実行するように、競合解消で用いる述語やスロットの有効度を変更する。

二つのルールの実行結果を比較するために、本システムでは "GOOD" である診断用ルールの数と "BAD" であるものの数との差を比較し、多数の方を採用する。

RULE-F からの競合解消には図5のように三つのケースがあり、各ケースにより述語およびスロットの有効度を変更する方法は次のようである。

(A) 有効度の変更はない。

(B) ルール b₁ の実行結果よりルール b₃ の実行結果がよい場合は、ルール b₃ の中で有効度が最も高い述語の有効度をルール b₁ の中で有効度が最も高い述語の有効度より高くする。

(C) ルール c₆ の実行結果がよりよい場合は、ルール c₆ の中で有効度が最も高いスロットの有効度をルール c₂ の中で有効度が最も高いスロットの有効度より高くする。ルール c₆ の実行結果が悪かった場合は、ルール c₅ の実行結果とルール c₂ の実行結果とを比較することによって述語の有効度を変更する。

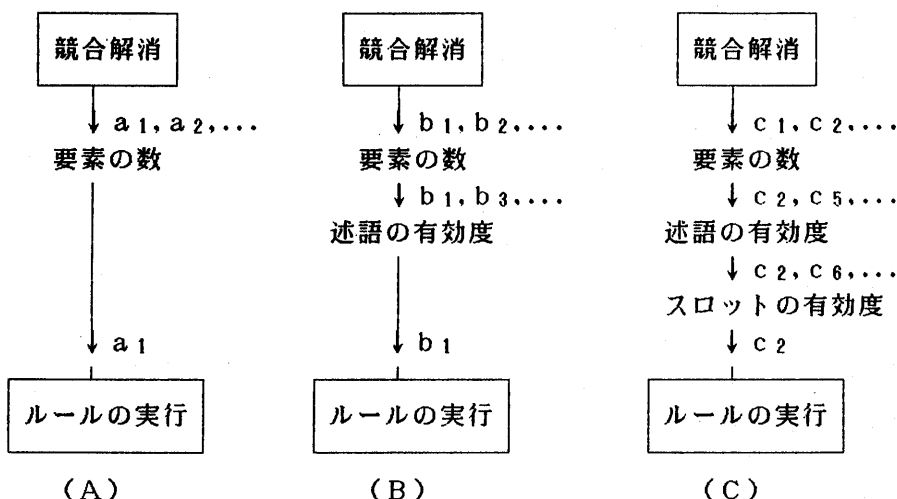


図5 競合解消の三つのケース

6. あとがき

故障診断用システムにおける初期の知識ベースを構築する方法について述べた。実際に構築したシステムは23個のデータベースからの知識抽出用ルール (RULE-F)、58個の入力データからの知識抽出用ルール (RULE-I)、船舶主機関に関する51個のフレーム、領域に依存しない知識として49個のルール、制御知識のための113個のメタルールを持っている。船舶主機関に対する実験結果では、RULE-F を用いて124個の診断用ルールが抽出され、8個のルールに対し専門家が誤った知識と判断した。この判断の中でシステムは2回有効度の変更を行なった。

本システムは一般化された知識抽出用ルールを用い、専門家の助言によって知識抽出過程を学習する方法を導入している。またルールベース中の専門知識と制御知識は分離されている。制御知識を専門知識から分離することは、専門知識を入替えることによって他の専門領域に対する知識獲得が容易に可能であり、一般化された知識抽出用ルールを用いることは、データベースのみを変更することによって機械の他の領域に関する知識獲得も可能になると考えられる。

参考文献

- [1] Blum, R. L., Representation of Empirically Derived Causal Relationships, Proceedings IJCAI-83, 268-271, 1983.
- [2] Clancey, W. J., The Advantages of Abstract Control Knowledge in Expert System Design, Proceedings AAAI-83, 74-78, 1983.
- [3] Davis, R., Interactive Transfer of Expertise: Acquisition of New Inference Rules, Artificial Intelligence 12, 121-157, 1979.
- [4] Ginsberg, A., Weiss, S., and Politakis, P., A Generalized Approach to Automatic Knowledge Base Refinement, Proceedings IJCAI-85, 367-374, 1985.
- [5] Politakis, P. and Weiss, S., Using Empirical Analysis to Refine Expert System Knowledge Bases, Artificial Intelligence 22, 23-84, 1984.
- [6] エキスパートシステム, AIUE0訳, 産業図書, 1985.