

5 Q-2

対象モデルに基づく故障シミュレーションシステムの開発

林高太郎 上野晴樹

東京電機大学

1. はじめに

経験則をプロダクションルールの形で蓄積した知識ベースにのみ頼る従来のエキスパートシステムの多くは、問題解決能力の柔軟性や知識の獲得及び管理に関して限界が指摘されている。

我々は対象モデルを核とする知識ベースシステムがこの問題に対し有効であると考え[1]、このアプローチをシステムの開発支援に適用する試みを行っている[2][3]。これまでに我々が開発した故障診断システムに故障シミュレーションを組み込むことにより、故障仮説が引き起こす対象への影響を検証しながら診断を行うことによる診断能力の向上を目指している。また、設計情報から得られる対象モデルを使った故障シミュレーションが可能になることにより、診断業務の能率向上に対して有効な手段になると期待している。

本稿では対象モデルに基づく故障シミュレーションシステムのプロトタイプの開発について述べる。

2. 対象モデル

対象モデルは、いわゆる深い知識表現の1つの提案であり、専門家が問題解決にあたって思い描くメンタルモデルを指向する。その中には、構造や挙動といった問題対象そのものが持つ性質に関する知識が体系的に表現される。対象モデルの持つこれらの知識は、設計情報から得ることができると考え、現在研究を進めている。

我々の提案する対象モデルに基づくディジタルシステムのモデル表現は、階層性〔抽象／具体関係と全体／部分関係〕、構成モデル〔論理構成モデル（以下LCMと略す）と物理構成モデル（以下PCMと略す）〕、コンポーネントモデル〔ブラックボックスモデルとグレイボックスモデル〕、フレームによる表現、の4つの特徴を有する[1][3]。

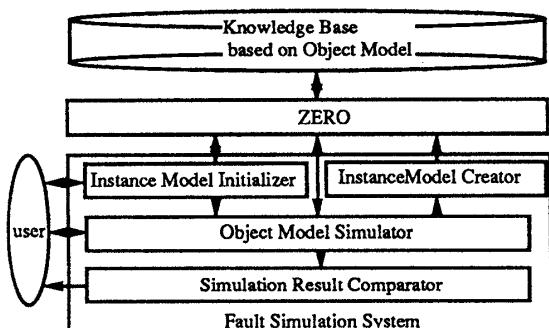


図1. 故障シミュレーションシステム構成図

Development on Fault Simulation System
based on Object Model
Kotaro HAYASHI, Haruki UENO
Tokyo Denki University

3. 故障シミュレーションシステム

3. 1. 故障シミュレーション

故障シミュレーションとは、正常なモデル(Normal Model, 以下NMと略す)と故障したモデル(Abnormal Model, 以下AMと略す)に対してテストパターン(故障を検出するための系列)を加えた場合のモデルの動作のシミュレーションを行い、どの故障を検出するかを調べるものである。

これにより、テストパターンとこの故障シミュレーションの結果から故障診断に必要な故障辞書が作成可能となる。

今回我々は、デジタルシステムの対象モデル上における故障シミュレーションのプロトタイプシステムを開発した。但し、今回作成したシステムは対象モデルのLCMに対してのみ行われる。

3. 2. 実現方法

本システムはSUN4/110上でSUN Common Lispとフレーム型知識表現環境ZERO[4]の拡張フレームシステムを用いて実現されている。システム構成を図1に示す。故障シミュレーションシステムの処理過程は、以下の3つのプロセスで実現されている。

1) 故障シミュレーションの準備：はじめに、対象の正しい振舞いや構造に関する知識が記述してあるレベルの対象表現をプロトタイプ／インスタンス階層の最下位にそのインスタンスモデルとしてNM, AMの2つの対象モデルを生成する。次に、故障仮説に従い図2に示すように特定の最小コンポーネントに故障挙動を埋め込んだAMを構築する。但し、故障を内部コンポーネントの特定の故障挙動に起因するものと仮定したとき、その内部コンポーネントと故障挙動の組を故障仮説と呼ぶ。また、故障は対象の最小構成要素の故障挙動に基づく单一縮退故障に限る。

続いて、モデルの内部状態、テストパターン、それを与える入力ポートを設定し、NM, AM両モデル上でシミュレータへ処理を移す。ただし、今回作成したシステムでは、複数のテストパターンには対応していない。

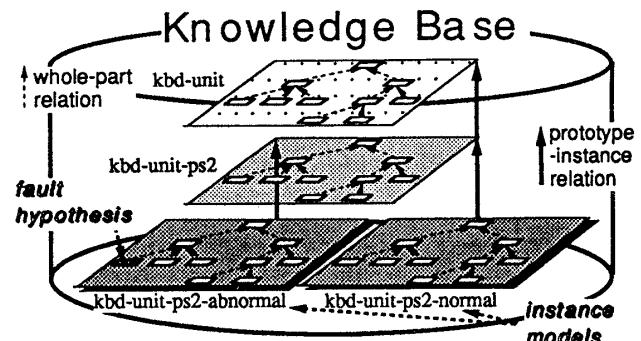


図2. 2つのインスタンスモデルと故障仮説

2) 対象モデルを使ったシミュレーション：NMに対するテストパターンの伝播過程において、システムは対象の全体／部分木の根に対応するするオブジェクトのBBMとしての振舞いだけに注目する。一方AMに対しては、特定の最小構成要素がもつ振舞いが異常であるので、GBMの機能経路に示される構成要素間の振舞いの相互作用として実現されるオブジェクトの振舞いに展開する。このGBMによる振舞いの展開は、注目しているコンポーネントの故障コンポーネントを含む副コンポーネントに関してのみ最小コンポーネントになるまで再帰的に繰り返し、それ以外のものはBBMとしての振舞いに注目するにとどめることにより、不必要にモデルの細部まで振舞いを展開しない。また、内部状態は各コンポーネントの振る舞いに対するテストパターンの伝播に伴いインスタンスモデルの内部状態は更新される。

3) テストパターンの評価：2)で得た2つの結果を比較し、設定した故障仮説に対するテストパターンの有効性を評価し、出力する。

3. 実行例

対象として、ワンチップのマイクロコンピュータを含むキーボードを扱った。実行例を図3に示す。“:>”は、ユーザの入力を促すプロンプトである。ユーザの入力はイタリック体で示してある。

故障シミュレーションシステムを起動すると、はじめにキーボードが唯一持つマイクロプロセッサの初期状態を設定する。次の故障仮説の設定は、LCMの最小構成要素のスイッチのひとつであるsw90が“圧力がかかったらパルスを送る”という機能を果たさなくなっている状態としている。続いて、テストパターンを与える対象のLCMにおける入力ポートを指定する。続いて、このポートに対するテストパターンの設定を行う。これで、システムの初期状態の設定を終了し、シミュレーションを実行する。

最後に、両モデルに対するシミュレーションの結果を比較し、設定した故障仮説に対するテストパターンの有効性を検証し、その結果を出力する。テストパターンが有効な場合には、NM、AMの各ポートにおける出力結果のリスト、対象モデル名、対象の内部状態の初期条件（初期状態を設定した部品名とその状態の並び）、故障仮説（故障コンポーネント名、振舞い名、振舞い記述の組）、テストパターンとそれを入力したポート名の組、を順に出力して終了する。有効でない場合には、そのテストパターンが設定した故障仮説に対して不適切であることを出力する。実行例は、テストパターンが有効な場合である。

4. おわりに

今回開発した故障シミュレーションシステムは、設計情報から作られる対象モデルに基づいたモデルベース推論により故障事例を生成するシステムである。故障シミュレーションの結果を学習し故障診断などにフィードバックすることにより、設計、開発、及び保守などの分野で有効に利用することが可能である[5]。

今後の課題として、以下の5つが挙げられる。

a. はじめに、対象モデルを構成する全てのフレームを生成するが、シミュレーションに不必要的フレームまで生成するため非効率的である。

b. 特定の故障仮説に対して存在しうる全ての入力パターンを適用することにより、正常に機能する部品と

機能不全の部品の振舞いの組み合せによる全体／部分関係における全体の故障の振舞いを生成することができると考えられる。

c. LCMとPCMのリンクを充実させることにより、対象モデル全体を通じた故障シミュレーションを実現し、故障診断に応用する。

```
;; Fault Simulation System start.
;; Focusing Object Model is KBD-UNIT-PS2.
--> Initialize internal states for PROCESSOR-PS2 of KBD-UNIT-PS2
initial state (on off) for /SCROLL-LOCK > off
initial state (on off) for /CAPS-LOCK > off
initial state (on off) for /NUM-LOCK > off
;; Both Models Initialization Finished.
--> Set a fault hypothesis.
Enter faulty object identifier> sw90
Behavior list: (IPRESS-PULSE-CONVERSION)
Choose a faulty behavior > IPRESS-PULSE-CONVERSION
Behavior template of IPRESS-PULSE-CONVERSION :
((.PRESSURE -> .PULSE)
((ACTION "down") (/KEY-NUMBER 90) -> (/ACTION "on") (/KEY-NUMBER 90))
((ACTION "up") (/KEY-NUMBER 90) -> (/ACTION "off") (/KEY-NUMBER 90)))
Describe the faulty behavior>
((.PRESSURE -> .PULSE)
((ACTION "down") (/KEY-NUMBER 90) -> (/ACTION "off") (/KEY-NUMBER 90))
((ACTION "up") (/KEY-NUMBER 90) -> (/ACTION "off") (/KEY-NUMBER 90)))
--> Set a port and a test pattern.
KBD-UNIT-PS2 has several ports as follows :(.KEYIN .INDICATION .SYSTEM)
Which port do you select to apply your test pattern > .keyin
The template of the test pattern for .KEYIN is :
((ACTION :VALUE ("push") "release"))(/KEY-NUMBER :VALUE (&RANGE 1 101))
Enter your test pattern > ((action "push") (key-number 90))
;; End of Initialization for Fault Simulation System.

simulating ...
;; The result of Fault Simulation followed.
;; The difference was found in the results between Normal and Abnormal Model.
;; The results are shown as follows.
The result of the Normal model:
((.SYSTEM ((OUTPUT "76")))
(.INDICATION
((/NUM-LOCK "bright") (/SCROLL-LOCK "dark") (/CAPS-LOCK "dark"))))
The result of the Abnormal model:
((.SYSTEM ((OUTPUT "F076")))
(.INDICATION
((/CAPS-LOCK "dark") (/NUM-LOCK "dark") (/SCROLL-LOCK "dark"))))
;; The initial condition for the results is shown as follows.
Object Model name: KBD-UNIT-PS2
Initial states for PROCESSOR-PS2:
((/SCROLL-LOCK "off") (/CAPS-LOCK "off") (/NUM-LOCK "off"))
Faulty behavior status for SW90:
IPRESS-PULSE-CONVERSION
((.PRESSURE -> .PULSE)
((ACTION "down") (/KEY-NUMBER 90) -> (/ACTION "off") (/KEY-NUMBER 90))
((ACTION "up") (/KEY-NUMBER 90) -> (/ACTION "off") (/KEY-NUMBER 90)))
Test Pattern for .KEYIN: ((ACTION "push") (/KEY-NUMBER 90))
;; End of Fault Simulation System.
```

図3 実行例

5. 参考文献

- [1]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－，信学技報，A186-4，1986
- [2]大森康正，上野晴樹：深い知識と浅い知識を組み合わせた故障診断システム－対象モデルの応用－，信学技報，A187-30，1987
- [3]山本雅仁 他：キーボードユニットの対象モデルとそれを利用した故障診断システムの設計，第4回人工知能学会全国大会論文集，16-9，pp507-510
- [4]今井，伊藤，吉村，上野：汎用フレーム・システム ZERO－その概要とユーザー・インターフェースについて－，信学技報，A187-22，1987
- [5]例えば、梅田 他：対象モデルに基づく定性物理を用いた故障シミュレータ，昭和63年精密工学会春季大会学術講演会論文集，C19，pp95-96