

TR-0780

定性推論における階層的知識の
自動生成

新庄 広、大木 優、大平 栄二
阿部 正博 (日立)

June, 1992

© 1992, ICOT

ICOT

Mita Kokusai Bldg. 21F
4-28 Mita 1-Chome
Minato-ku Tokyo 108 Japan

(03)3456-3191 ~ 5
Telex ICOT J32964

Institute for New Generation Computer Technology

定性推論における階層的知識の自動作成

Acquisition of Hierarchical Knowledge in Qualitative Reasoning

新庄 広 大木 優 大平 栄二 阿部 正博
Hiroshi Shinjo Masaru Ohki Eiji Oohira Masahiro Abe

日立製作所 中央研究所
Central Research Laboratory, Hitachi, Ltd.

Abstract

In using qualitative reasoning, hierarchical knowledge representation is indispensable for reasoning of massive target systems. So this method acquires shallow knowledge from deep one. The acquisition process consists of two parts. One is qualitative simulation to analyze the behavior of the module which is composed of deep knowledge. The other is refinement of the simulation result to acquire the generalized knowledge. The representation of acquired knowledge is based on input and output of the module. And we extend the framework of the knowledge representation in order to treat with all behavior of the module. The acquired knowledge has the information of behavioral transition as well as conditions and relations. The characteristics of the acquired knowledge is as follows; (1) High speed reasoning is accomplished, because modularized knowledge is utilized by only input and output without analyzing inside of the module. (2) The acquired knowledge can be utilized with knowledge of other hierarchical level (shallow and deep).

1. 緒言

従来のエキスパートシステムの多くは浅い知識と呼ばれる専門家の経験的な知識を用いて問題解決を行なっている。しかし、(1)知識の獲得が困難、(2)あらかじめ予想された問題しか解決できない、(3)推論対象の変更に対し知識ベース内の知識を矛盾なく追加削減することが困難である、という問題点が指摘されてきた。これに対し、個々の問題に依存しない原理的な知識を利用することによって、上記の問題を解決しようとする研究がある。このような知識を深い知識といい、物理法則などがこれに相当する。

定性推論は深い知識を用いた推論方式の一つである[西田 88]。深い知識を用いることにより、個々の問題に依存しない推論システムの実現が期待できる。その反面、常に深い知識から問題を解決しては不都合な面もある。大規模な問題に対しては、推論の探索空間が爆発的に増加するため、推論時間が長くなり、推論の曖昧さも増大するからである。このため、定性推論は規模の大きな問題に適用された例は少なかった。

深い知識を用いた定性推論を実問題に適用するには、問題に応じて用いる知識の深さのレベルを使い分ける必要がある[Falkenhainer 88, 吉田 89]。詳細に解析したい場合には、より深い知識を利用すべきである。そうでなければ、より浅いレベルの知識を利用する方が無駄な処理を省くことができる。

本研究では、深い知識から、より浅い階層の知識を自動的に作成する方式を提案する。ここで、深い知識と、作成する浅い階層の知識を、電子回路の例で説明する。抵抗やトランジスタなどの特性とキルヒホッフの法則等、回路一般に扱えるも

のを深い知識とする。それに対し、アンプやシュミット・トリガ回路などの入出力特性等、深い知識を用いて求めることのできるものを、浅い知識とする。そして、深い知識を用いて浅い知識を作成することを知識のモジュール化と定義する。モジュール化した知識を作成する作業は、従来はユーザー自身が手作業で行なっていた[岩佐 91]。しかし、一般に知識のモジュール化をユーザーが行なうことは困難であるといえる。なぜならば、そのモジュールの挙動をもれなく調べた上で、挙動を矛盾なく表現できるように、知識として記述しなければならないからである。本方式は、この問題を解決するものである。

本方式では、(1)定性推論によるシミュレーションを用いてモジュール化する部分を取り得る全ての状態遷移関係を求め、(2)その推論結果に基づいて浅い階層の知識を作成する。全ての状態遷移を求めていることにより、あらゆる利用条件に対応できる。

さらに、モジュール化した浅い階層の知識を利用する際の特長は以下の通りである。

- (1)モジュールの内部状態を解析することなく扱うことができるため、処理が高速になる。
- (2)深い知識として定義された他の知識と併用できる。
- (3)一般に定性推論は連続的な変化を解析するものであるが、本方式では、不連続変化を持つモジュールも扱うことができる。

また、学習という観点から本方式を見ると、

- (1)学習に必要な事例をシステム自身が生成し、
- (2)その事例を一般化する機能を持つ

学習システムとみなすことができる。しかも、(1)

では全ての事例を生成できるため、一般化された知識は健全性を持つ知識を学習していることになる。

2. 定性推論

2.1 定性推論の概要と問題点

定性推論は連続的に変化する動的システムの挙動を推論し、その結果を人間にわかりやすく説明することを目的としている。推論対象を解析する際には、変数の値の変化ではなく推論対象の特徴的な挙動の変化に注目して挙動を求める。例えば、1段アンプの入力電圧が上昇する場合、詳細な電圧の値には注目せず、OFFからONの状態変化を求める。これは状態遷移を求めていることに相当する。

定性推論には、大きく2つの種類がある。一方は推論対象の挙動を表すモデルを、あらかじめユーザーが数式などを用いて宣言的に記述しておくものである[Kuipers 84]。しかし、この方式の問題は、矛盾なく全体のモデルを記述することが難しいことである。もう一方は推論対象の構成要素のモデルと、それらを組み立てる物理法則などを深い知識として用意しておき、問題に応じて推論対象全体のモデルを自動的に構築するものである[Forbus 84]。本研究は後者の推論方式を採用している。

また、定性推論の特徴の一つにエンビジョニングの機能がある。これは、推論に必要な情報が不足していても、与えられた初期条件のもとで、起こり得る全ての状態とその状態遷移を求めることができるというものである。

最も素朴な定性推論では、量を[+]、[-]、0の3通りに量子化して増加状態、減少状態などをはじめとする性質の変化をとらえている。しかし、電子回路を扱う場合に、抵抗の値は全て[+]であるため区別できないなど、正確な量の情報が欠落するため推論が曖昧になるという問題がある。

定性推論では、局所的に得られた制約を、推論対象全体に伝播することによって、全体としての挙動を求めている。しかし、局所情報を伝播することは、必ずしも一意に決まるものではない。この曖昧性のために、大規模な推論対象に適用しようとするとき、致命的な組合せの爆発を招く。

2.2 定性推論システム Qupras

定性推論システム Qupras[大木 88, Ohki 92a]は、推論対象の挙動を定性的に解析するものの、量を定性的に扱うことによる曖昧さを減らすため、定性情報のみでなく定量情報も扱うことができる。すなわち、値が既知の変数は量子化せず、定量的な値をそのまま用いて推論

を行なうことができる。Quprasの特徴は以下の2点である。

- (1)深い知識から推論対象のモデルを連立方程式(不等式)の形で作成することができる。
- (2)定量情報を扱うことができる。

図1にQuprasのシステム構成を示す。入力データは初期条件と推論対象の構造情報である。これらと知識ベース内の深い知識を用いて推論を行なう。初期条件や深い知識は、方程式もしくは不等式で記述されている。電子回路の場合、深い知識は素子の特性やキルヒホッフの法則などの物理法則であり、構造情報は素子の結合関係である。モデル構築推論部は、回路を構成する素子一つずつ評価し、素子の特性に関する方程式もしくは不等式をワーキングメモリに格納する。全ての素子や端子の評価が終ると、ワーキングメモリに記録された連立方程式(不等式)は、推論対象を表現するモデルになる。制約ソルバーが数式処理によってこの連立方程式を解くことにより、個々のパラメータを求め、推論対象の状態を決定する。さらに、一つの状態を求めた後、限界解析部が変数の変化状態を解析する。ここで求めた変数の変化に基づいて次状態の解析を行なう。このような過程を繰り返して得られた推論結果は、推論対象の挙動の状態遷移を表している。

図2(a)に示すDTLについて、入力電圧を増加させたときの挙動を推論した結果を表1に示す。この時、初期条件には入力電圧(V_{in})が0、増加を表現するものとして入力電圧の微分値(dV_{in}/dt)が[+]と与えられる。推論は瞬間と時区間の状態の解析の繰り返しである。瞬間の解析とは、回路の内部状態の変化が起こる可能性のある点(ランドマーク)を見つけ、その瞬間での状態を求めることである。次のランドマークまでの時区間では状態は変化しないので、まとめて解析する。表1より、瞬間と時区間の解析を10回行なっている。その間、内部状

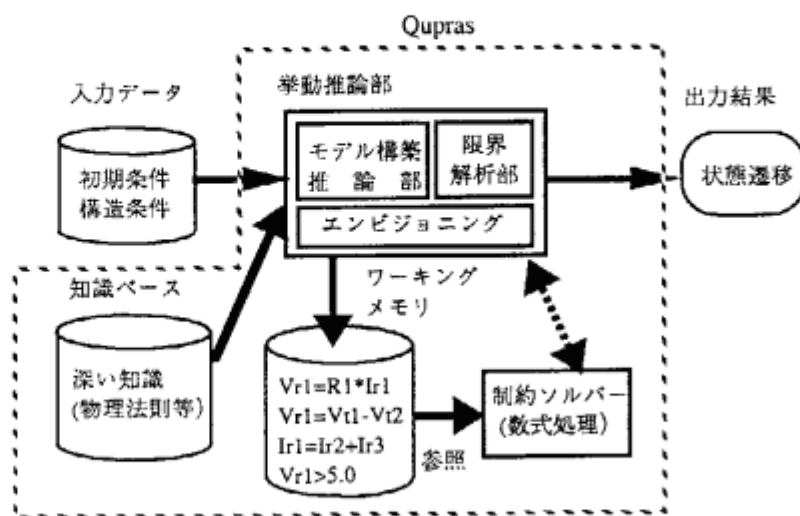
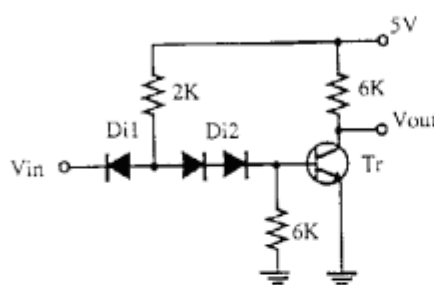
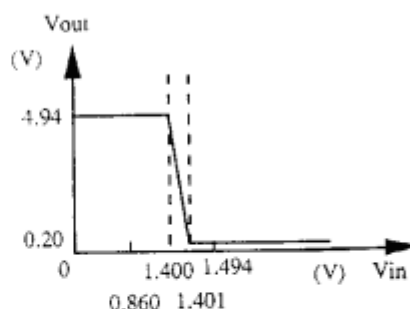


図1 定性推論システム Qupras のシステム構成



(a) 回路図



(b) 入出力特性

図2 DTL

態は5状態の遷移を行なっていることがわかる。なお、表中において、内部状態がOFF-ON-SATとは、入力側のダイオード (Di1) がOFF、トランジスタ側の2個のダイオード (Di2) がON、トランジスタが飽和状態であることを示す。ここでは、トランジスタのベースに直列に接続している2個のダイオードは、電流が同じであり、状態の変化も同時に起こるため、ひとまとめに扱っている。この表1に示した推論結果に基づいて、図2(b)の入出力関係も求めることができる。

Quprasの深い知識は存在条件とその時に成立する関係式によって記述されている。図3にダイオード (Di) を深い知識として記述した例を示す。supersはそれが継承している上位の概念である。ダイオードは、2つの端子を持っている (paris_of) thingを継承しているので、2端子の素子である。attributesはその知識の内部変数である。ここでは、ダイオードの内部抵抗と電圧、電流が定義されている。stateは状態に関する定義である。ここでは、電流が流れる状態をon、ほは電流が流れない状態をoffと定義している。各状態はその存在条件 (conditions) と成立する関係式 (relations) で表現されている。

Quprasは、エンヴィジョニングの機能を実現するため、不足した情報は推論システム自身が仮説を立てて補う。前述のDTLにおいて、ある抵抗値がわからないために、ダイオードの端子間の電位差がわからない場合、ダイオードはONとOFFの2状態を持つので、両方の状態が成立することを想定して2通りの仮説を立てる。ONとOFFの状態は、端子間の電位が0.7V以上かそれ未満かで決まるので、それぞれの仮説は $V \geq 0.7$ と $V < 0.7$ という不等式になる。この仮説に基づいて2通りのモデルを構築する。その後の解析の結果、一方の仮説に基づくモデル

表1 Qupras によるDTLの推論結果 (状態遷移)

状態	内部状態 (Di1 - Di2 - Tr)	入力電圧 Vin (V)	出力電圧 Vout (V)
1	ON-OFF-OFF	Vin = 0.000	Vout = 4.940
2	ON-OFF-OFF	0.000 < Vin < 0.860	Vout = 4.940
3	ON-ON-OFF	Vin = 0.860	Vout = 4.940
4	ON-ON-OFF	0.860 ≤ Vin < 1.400	Vout = 4.940
5	ON-ON-OFF	Vin = 1.400	Vout = 4.940
6	ON-ON-ON	1.400 < Vin < 1.401	0.200 < Vout < 4.940
7	ON-ON-ON	Vin = 1.401	Vout = 0.200
8	ON-ON-SAT	1.401 < Vin < 1.494	Vout = 0.200
9	ON-ON-SAT	Vin = 1.494	Vout = 0.200
10	OFF-ON-SAT	1.494 < Vin	Vout = 0.200

に矛盾が生じた場合には、その仮説と構築したモデルを破棄する。このように、仮説を立てながら、起こり得る全ての状態を求めることができる。さらに、Quprasの数式処理機能を用いてモデルの連立不等式を解くことによって、変数の値の範囲を求めることができる。これは、その状態を成立させるための、変数の条件に相当する。

3. 深い知識に基づく階層的知識表現

3.1 定義

本研究では、深い知識を用いて挙動推論を行なうことができる部分を、モジュール化して浅い知識を作成する。その概念を電子回路を用いて、図4に示す。図4(a)の回路を解析する際に、点線内のみを詳細に解析したい場合には、それ以外の部分は必ずしも深い知識から推論する必要はない。そこで、図4 (b)に示すように、深い知識をまとめて、モジュールの入出力のみに注目した知識を作成するものである。モジュールの入力と出力は、深い知識における成立条件とその時の成立関係に相当す

```

object tanshi:Tanshi
  attributes
    vr;
    ir;
end.

object thing:Thing
  parts_of
    tanshi1-tanshi;
    tanshi2-tanshi;
end.

object di:Di
  supers
    thing;
  attributes
    registance;
    vr;
    ir;
  state on
    conditions
      vr@Di >= 0.7;
    relations
      registance@Di=1.0;
      vr@Di=vr@tanshi1!Di-vr@tanshi2!Di;
      vr@Di=registance@Di*ir@Di+0.7;
  state off
    conditions
      vr@Di < 0.7;
    relations
      registance@Di=100000000.0;
      vr@Di=vr@tanshi1!Di-vr@tanshi2!Di;
      vr@Di=registance@Di*ir@Di;
end.

```

図3 Quprasにおける知識表現の例

る。したがって、入力が決まれば、内部状態を解析することなく、直ちに出力を得ることができる。ただし、モジュール化する範囲の指定はユーザーが行なう。電子回路では、モジュール内部の素子の結合関係の定義が範囲の指定にあたる。

作成した知識を用いて推論を行なう際には、以下のような条件が要求される。

- (1)他の深い知識と同様に扱うことができる。
- (2)モジュール内部の変数を考慮せず、入出力のみに着目して扱うことができる。

(1)の条件により、モジュール化した知識を用いて、さらに大きなモジュールの知識を作成することができる。この作業を繰り返すと、階層的な知識ベースを構築することができる。(2)の条件は、推論の高速化につながる。

本方式は、より浅い階層の知識を作成する際に、定性推論を用いて挙動推論を行なった結果を用いる。定性推論に基づくシミュレーションは推論対象の状態遷移を求めていることに相当する。状態遷移という観点から考えた場合、深い知識に基づくシミュレーション結果から、より浅い知識を作成するには、図5に示すように、状態遷移の違いによって作成すべき浅い知識を次の3つに分類することができる。

- (a)一つの状態しか持たない
- (b)複数の状態を持つ
- (c)複数の状態を持ち、状態遷移に順序関係を持つ

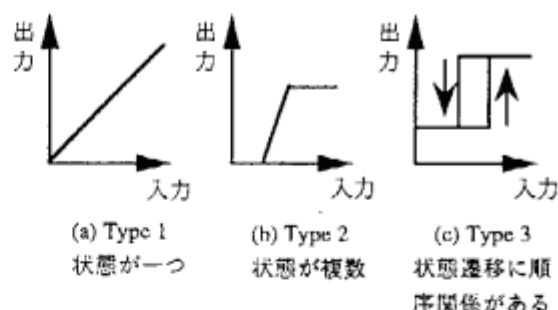


図5 浅い知識の分類

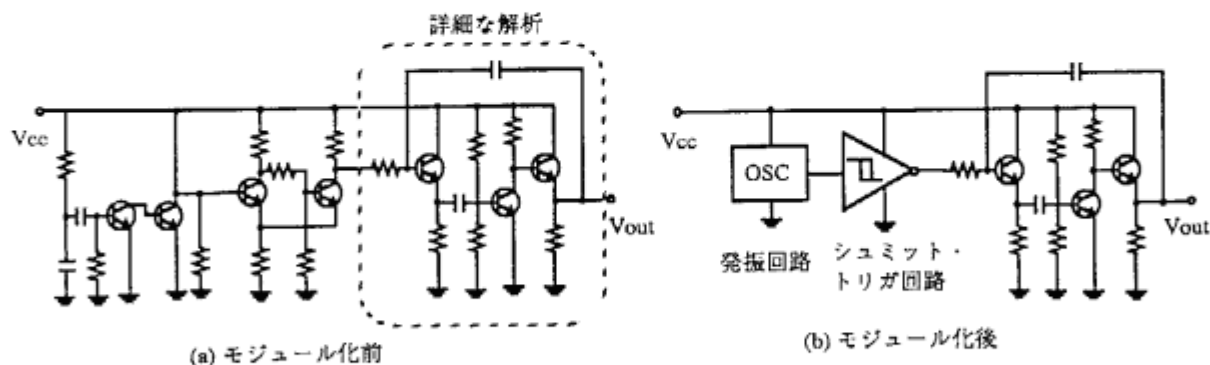


図4 深い知識のモジュール化

ただし、ここでいう状態とは、入出力が連続した一つの関数で表されているものを指す。

まず第1の種類は、図5(a)に示すように、モジュールの挙動が一つの状態しか持たないものである。これは、飽和特性のない仮想的なアンプなどに相当する。このような性質のモジュールの知識を作成するには、入出力関数を求めるだけでよい。第2の種類は、図5(b)に示すように、モジュール化した部分の挙動が、ある境界によって複数の状態に分けられるものである。この場合は、全ての状態における入出力関係を求めればよい。第3の種類は、図5(c)に示すように、モジュールの内部状態の影響によって、状態遷移に順序関係を持つものである。図5(c)の場合、入出力関係がヒステリシスを持つため、同じ入力に対して、2通りの出力をとり得る。正しい出力は、前の状態が分からなければ求めることができない。また、AND回路やOR回路は入力と出力が一对一の関係を持っているため第2の種類であるが、これらを組み合わせたフリップフロップなどは、同じ入力に対しても前の状態に依存して出力が異なるため、第3の種類に入る。したがって、このようなモジュールは、入出力関係だけから浅い知識を作成しても、正しい挙動を求めることはできない。状態遷移の順序関係の情報が必要である。

本方式はこの3種類のモジュールを統一的に扱える知識を作成することと目的としている。モジュールの知識を作成する前には、必ずしもどの種類になるかが分かるわけではないからである。そのために、モジュールが持つ全ての状態について、

- (1)入力条件（成立条件）
- (2)入出力関係（成立関係）
- (3)状態遷移の情報

の3つの情報を持つ知識を作成する。

3.2 不連続変化の表現

本方式で作成する浅い知識は、不連続な変化も適用可能な知識表現をとる。そこで、不連続な変化の表現方法について定義する必要がある。こ

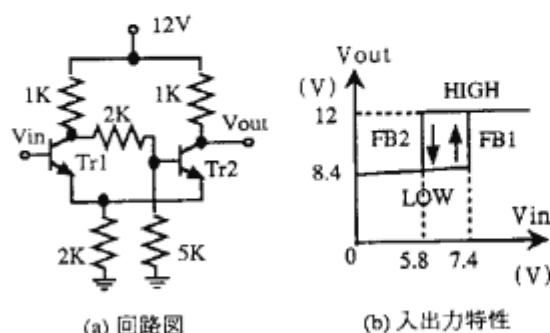


図6 シュミット・トリガ回路

では、代表的な不連続な変化として正帰還を例として、その取り扱いについて説明する。

図6に正帰還を持つ電子回路として、シュミット・トリガ回路とその入出力特性を示す。正帰還の状態は瞬時に起こるため、不連続にジャンプしているとみなすことができる。正帰還の状態の扱い方により、この回路をモジュール化した知識の記述には、以下の2つが考えられる。

- (a)正帰還を無視して、不連続な2状態（HIGH, LOW）のみを考える。
- (b)正帰還を含めて、連続的な4状態（HIGH, LOW, FB1, FB2）を考える。

本方式では(b)の連続的な記述を採用する。(a)の不連続な記述は、状態数が少ないため、この知識を用いた推論時の処理時間が短くなるという利点がある。しかし、定性推論は変数の連続的な変化の解析に基づいているので、不連続な記述をした知識を、従来の連続的な記述に基づく知識に併用して利用した場合に不都合を生じることがある。以下にその根拠を詳述する。

第一の理由は、連続的な変化を解析する定性推論で不連続な変化を扱うには、例外処理を行わなければならない、推論系が複雑になるためである。例として、不連続な表現を用いたシュミット・トリガ回路の出力側に他の回路を接続した場合を考える。接続点における電流と電圧の不連続な変化は容易に求めることができる。しかし、この不連続な変化の影響を回路の他の部分に伝播するには、因果解析などを必要とするため難しい。特に、フィードバックが入るなど問題の規模が大きくなると非常に困難である[Iwasaki 88]。この結果、複雑な例外処理が必要であるにもかかわらず、汎用性が低い。

第二の理由は、不連続な記述の知識と連続的な記述の知識とを併用した場合、正しく推論を行なうことができないことがあるためである。例として、図7に示すように、単調な変化の後に不連続変化が起こる入出力特性を持つモジュール(M1)と単調増加の後に飽和特性を持つモジュール(M2)を直

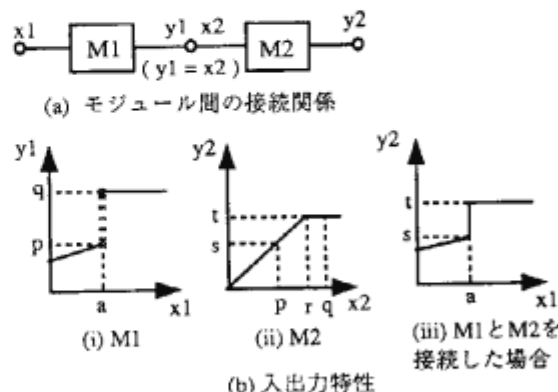


図7 不連続な記述が矛盾する例

列に接続した場合を考える。M1の入力を増加させるシミュレーションを行なうと、M1の挙動が変化する点である $x_1=a$ までは、全体の出力である y_2 を求めることはできる。このとき、 y_1 と x_2 の値は同じであるので、 $y_2=s$ となる。次に、M2の挙動は $x_2=r$ の点で変化するの、推論系は $x_2=r(=y_1)$ のときの全体の挙動を解析する。しかし、M1の $x_1=a$ での変化を不連続に記述していると、M1は $y_1=r$ を満たす x_1 の値がないため、推論が矛盾してしまう。一方、この部分を連続にとらえて、 $p < y_1 \leq q$ と記述しておけば、このような矛盾は起こらない。

4. 階層的知識の作成

4.1 全状態の導出

深い知識をモジュール化して作成する階層的知識は、どのような状況でも利用できるようにするために、全ての入力に対応できなければならない。そこで、モジュールのとり得る全ての状態の入出力関係とその境界を求める必要がある。ところが、一般的に全ての入力変数の範囲を網羅的に調べることは困難であると考えられる。例えば、図2のDTLをモジュール化する場合、トランジスタやダイ

オード等、複数の状態を取り得る素子を含んでいるので、入出力関係も複数の状態を持つと考えられる。しかし、このモジュールの知識を作成する前に入出力特性がわかっているとは限らない。その場合、状態の数や状態間の境界がどこにあるかはわからない。そこで、全ての状態とその状態間の境界を求める方法が問題になる。

全ての状態は、単に知識として作成するモジュールの挙動をシミュレーションするだけでは、求めることができるとは限らない。その理由を図8に示すシュミット・トリガ回路の入出力特性を例として説明する。入力が増加する時と、減少する時では挙動が異なるので、増加、減少の2種類のシミュレーションをすれば、全ての状態を求めることができると考えられる。しかし、この方法では、変化の方向も考慮した全ての状態を求めることができない。つまり、(a)と(d)の部分は、入力が増加、減少両方の状態が求まっているのに対し、(b)と(c)の部分は、入力が増加、または、減少の片方の状態しか求めることができない。したがって、(b)の部分で入力が増加する場合等には対応できない。このような結果からシュミット・トリガ回路の知識を作成し、推論に用いると間違いを起こす。例えば、入力電圧を0ボルトから増加させ、 $x \sim y$ の間の(b)の状態に減少させる場合、本来なら(b)の状態のままである。しかし、入力電圧が $x \sim y$ の範囲で減少している条件は(c)の状態しかないの、(c)の状態へ遷移するという誤った推論結果を導いてしまう。

この問題解決するため、本方式では、Quprasのエンヴィジョニングの機能[Ohki 92b]を用いて全ての状態を求める。モジュールの知識を作成する際にエンヴィジョニングを利用するには、モジュールの入力変数の値を未定義にして定性推論を行う。入力変数に関する制約がないため、エンヴィジョニングにより、入力変数の値の違いに基づく全ての状態を求めることができる。さらに、Quprasの持つ数式処理能力により、各状態を成立させるための入力変数の値の範囲を求めることができる。

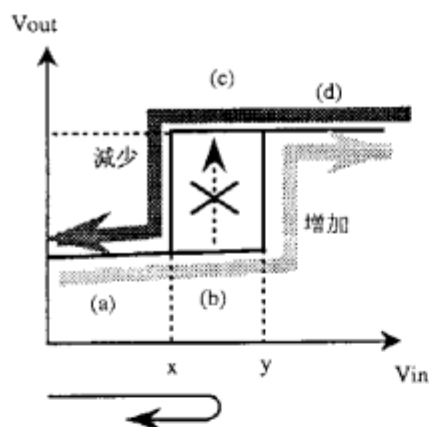


図8 シミュレーションで全ての状態を求めることができない例

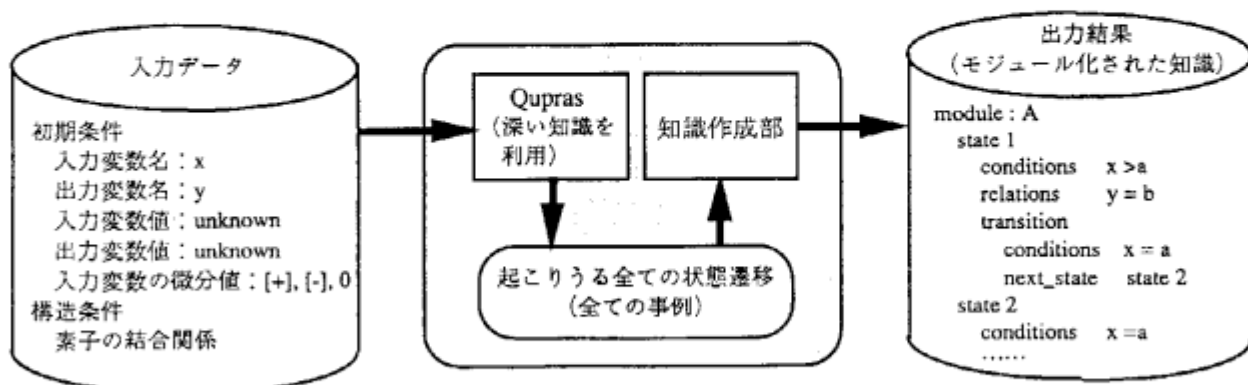


図9 階層的知識作成のシステム構成

4.2 階層的知識作成過程

本方式のシステムの構成を図9に示す。入力データとして、入出力変数名、入力変数の変化状態、モジュールの構造の定義を与える。変化状態を与える理由については、4.2.1で後述する。また、エンヴィジョニングを行なうために入出力変数の値は未定義にしてある。システムは、まず Qupras を用いて全ての状態を求め、その結果をまとめることによって、知識を作成する。得られた知識は、そのモジュールの持つ特徴的な状態ごとに、入力変数の条件と入出力関係を表わす式、及び状態遷移に関する情報を持っている。

知識作成過程について、図10を用いて説明する。全体は次の3つの段階にわかれる。

- step 1: 入力値を未定義、入力の変化状態を定義して、定性推論を行ない、全ての状態と、そこで求めた各状態からの状態遷移を求める。
- step 2: 同じ状態に対して、成立条件のORをとることで状態を統合することにより、冗長な情報を除去する。
- step 3: step 1で求めた状態遷移に基づいて、step 2

で統合した各状態間の順序関係を求め、浅い階層の知識として記述する。

step 1はQurasを用いて行なわれ、step 2とstep 3は知識作成部で行なわれる。以下にそれぞれについて詳述する。

4.2.1 全ての状態と状態遷移の導出

まず全ての状態を求め、各状態から遷移可能な状態を求めることによって、起こり得る全ての状態遷移を得る。全ての状態を求める方法は、4.1で示したように、シミュレーションの初期条件として、モジュールの入力変数の値を未定義にしたエンヴィジョニングを用いる。ただし、各状態間の遷移関係を求めるため、入力変数の微分値を $[+], [-], 0$ の三種類の初期条件を加えたものそれぞれについて、シミュレーションを行なう。微分値の $[+], [-], 0$ とは、入力変数が増加、減少、一定の変化状態を表すものである。このように、全ての状態から入力を3種類に変化させることにより、各状態に隣接する全ての状態遷移を求めることができる。なお、図10では入力一定の場合は省略している。

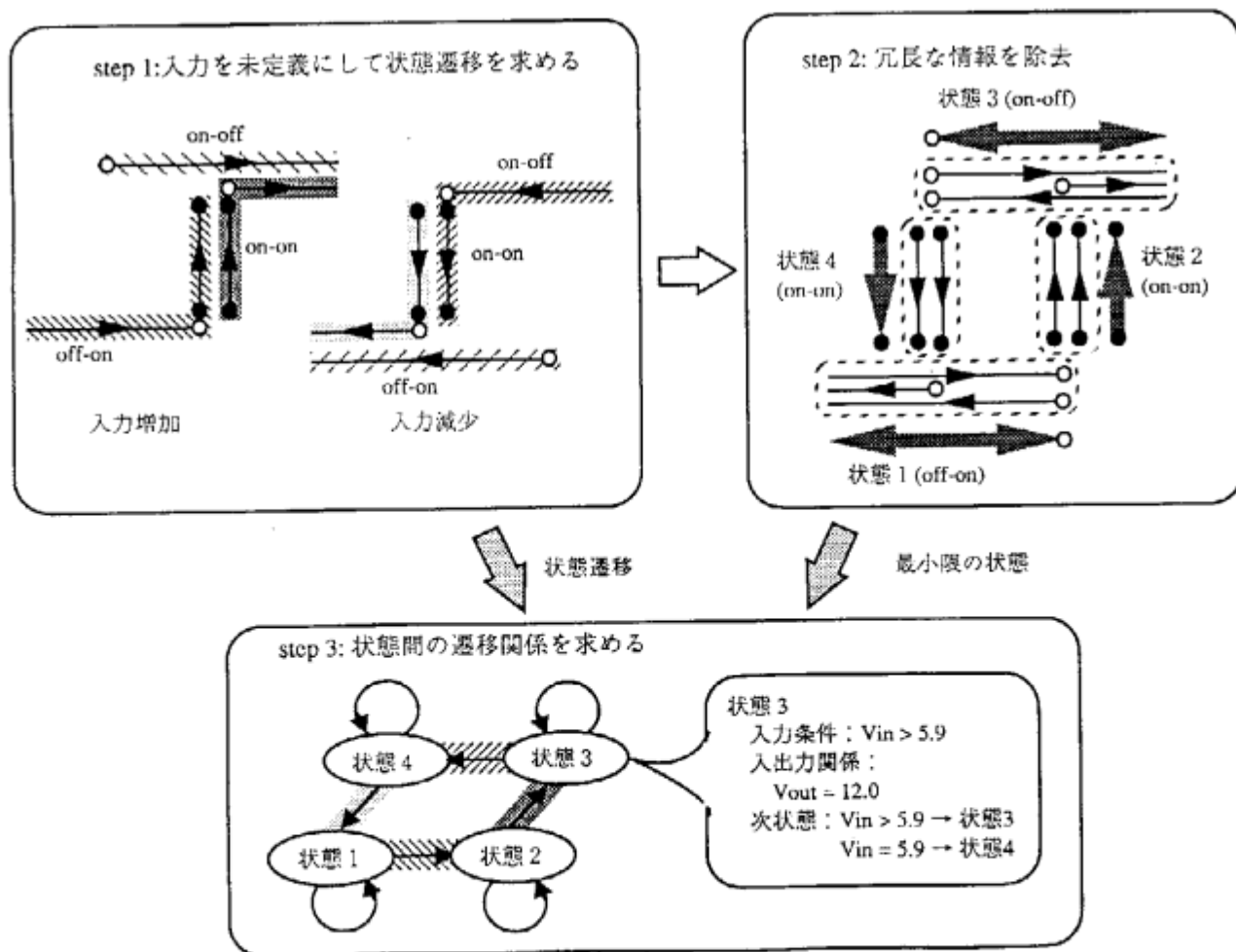


図10 階層的知識作成過程

4.2.2 冗長な情報の除去

4.2.1で求めた状態を全て記録して知識とするのは冗長である。したがって、挙動を表現するために必要最小限の状態を作成するため、同じ状態であるとみなせるものは統合する。状態の統合の基準は次の2つである。

- (1)出力変数の値が同じ
- (2)モジュールの状態を表現するモデルである連立方程式（不等式）が同じ

この2つの条件のうち少なくとも一方を満たす状態があれば統合の対象になる。これらの条件を満たせば、両者の状態の入力条件（成立条件）を比較する。変数値の範囲と微分値についてそれぞれORをとることにより、その状態についての入力条件を広げる。図10(b)の状態1や状態3のように、一つの状態について微分値が[+],[-],0の全てをとる場合には、微分値に関する条件を入力条件から削除して、入力値の範囲のみを入力条件とする。

4.2.3 状態間の遷移関係を求める

4.2.1で求めた状態遷移の情報に基づいて、4.2.2で統合した各状態間の遷移関係を求める。このようにして得られた各状態は次のように記述されている。

- (1)入力条件：入力変数の条件
(等式、不等式)
- (2)成立関係：出力変数の値（出力値が定数の時）
成立する関係式（出力値が定数でない時）
- (3)次状態に関する情報
 - (a)次状態での入力条件の候補
 - (b) (a)を満たす時の次状態

モジュール化した知識は入出力に注目して、内部をブラックボックスとして扱うため、出力値が定数の時は成立関係には出力値の定数のみを記述する。定数でない時は連立方程式全てを記述している。連立方程式全てを持っているので、推論の際、入力変数の値が与えられれば、その値を連立方程式に代入することにより出力値を求めることができる。

5. 評価実験

深い知識として、トランジスタ、抵抗の特性に関する知識と、キルヒホッフの法則を用いて、図2(a)に示すDTLと図6(a)に示すシュミット・トリガ回路のモジュールの知識を作成した。DTLは状態を複数持つ回路であり、3.1の(b)のモジュールに相当する。シュミット・トリガ回路はヒステリシスを持つため、状態間に順序関係がある回路であり、3.1の(c)のモジュールに相当する。表2にこの2つの回路のモジュール化前後における、知識表現や処理の軽減について示す。ここで、挙動推論とは、DTLとシュミット・トリガそれぞれについて、入力電圧を0(V)から増加させることによる挙動を求めたものである。モジュール化前は、深い知識に基づき素子一つずつ評価しながら、推論を行なった。モジュール化後は、作成した知識を用いて、モジュールの入力を評価することにより、出力を求めて推論を行なった。

知識をモジュール化することによって、評価する式の数が増え、挙動推論の処理時間も短縮されていることがわかる。高速化については、DTLで46倍、シュミット・トリガでは11倍を実現できた。

式の数が増える効果よりも、処理時間の短縮の効果の方が大きい。これはモジュール化前後では、知識の持つ式の意味が異なるからである。モジュール化前の式は、深い知識として記述されている式なので、このような式を用いて状態を決めるには、(1)論対象のモデルを構築するために、式一つずつ評価する(2)全ての式を評価した後、連立方程式を解く、という2つの処理が必要である。これに対し、モジュール化後の式は、構築されたモデルを表現する最小限の式なので、(2)の処理だけを行えばよいため高速になる。

6. 考察

定性推論では、挙動を状態の変化でとらえているため、本研究で作成したモジュール化した知識も、状態に注目した記述になっている。この結果、人間の理解に似た知識の記述になっている。シュ

表2 知識のモジュール化による処理の軽減

	DTL			シュミット・トリガ		
	モジュール化前	モジュール化後	比 (前)/(後)	モジュール化前	モジュール化後	比 (前)/(後)
一状態の解析で評価する式の数(平均)	107	25	4.28	85	21	4.05
挙動推論時に遷移する状態の数(入力増加時)	10	6	1.83	13	7	1.86
挙動推論時に評価する式の総数(入力増加時)	1070	149	7.90	1105	237	4.66
挙動推論時間(sec)	10702	232	46.13	7908	704	11.23

ミット・トリガの例では、LOW、HIGH、正帰還（上昇）、正帰還（下降）の4状態に自動的に分けられている。これは、電流や電圧といった個々の変数の変化から、シュミット・トリガの挙動全体に着目した上位の概念を獲得していると考えられる。

DTLをモジュール化した場合、本方式は、知識作成の際にモジュールの内部状態よりも入出力に着目しているため、上位の概念を獲得していることが、より明らかになる。DTLの挙動は、ダイオードやトランジスタといった内部状態に着目すると、表1に示されたように全体は5状態に分かれる。すなわち、定性推論における状態の変化に着目するという観点から見た場合は5状態である。したがって、DTLのモジュールの知識は5状態を持つと作成しても、個々の変数ではなく挙動に着目した上位の概念を得たといえる。ところが、入出力の変化に着目し、出力が同じものは統合して、5状態より3状態の方がより人間の理解に近いといえる。そこで、DTLをモジュール化した知識は次の3状態で構成されている。まず、表1の状態1から状態5に相当する部分と $V_{in} < 0$ の部分をひとまとめにしたものがHIGHの状態、状態6がMIDの状態、そして状態7から状態10までに相当する部分がLOWの状態である。本方式では、知識作成の際に、単に定性推論で得られた結果を統合しているだけでなく、入出力に着目し、内部状態などの挙動に関する情報を全て入出力変数にマッピングしている。その結果、人間が図2(b)の入出力特性を認識するように、3状態の記述を行なっている。これは内部状態などに基づく状態表現より上位の概念が得られていると考えている。

さらに、近似や省略の機能を本方式に持たせることができれば、より上位の概念も得ることができる。DTLのMIDの状態に注目すると、この状態のみが他の2つの状態に比べて入力範囲が非常に狭いことがわかる。この状態を無視して、知識表現を変更することができれば、ONとOFFの2状態のみとなる。これは、回路の機能を表現していることになり、入出力特性より上位の概念を獲得することになる。

本方式では、異なる階層の知識が一貫した記述で表現されているため、異なる階層の概念を併用して推論することができる。例えば、個々の構成要素の変数の変化などの深い知識で表現された概念と、モジュールの入出力特性といった上位の概念を同時に利用することができる。また、本方式で用いているQuprasは定性推論において数式処理を行なっているため、値の変化に着目した定量的な概念と、状態の変化に着目した定性的な概念を併用できる。これは、定性的な記述のもとでも、入力変数の正確な値が与えられれば、状態だけでなく、出力変数の値も求めることができるためである。さらに、不連続な変化にも対応しているため、アナログの概念とデジタルの概念を併用して扱

うことも可能になる。このような特徴は、同じ推論対象の中で、部分的に解析の詳細度を変えて推論したい時に役立つ。

従来の深い知識に用いていた知識表現では、成立条件とその時の成立関係のみを記述していたので、状態遷移に順序関係のあるものは扱えなかった。推論対象の状態遷移の順序性となる内部状態は、このような順序性を持たない深い知識によって表現されているため、深い知自身の状態遷移まで考慮する必要がなかったからである。しかし、複数の知識をモジュール化する場合には、状態遷移の順序性の扱いが問題になる。そこで、本方式では、作成した知識に状態遷移の情報を持たせることにより、この問題を解決した。さらに、状態遷移の情報を知識が持つことのもう一つの意味は、推論時に行なう次状態への変化の解析を、知識自身が情報として持っているため、推論を高速化することができるということである。

学習という観点から考察すると、本方式は

(1)学習に必要な事例をシステム自身が生成し、

(2)それを一般化する能力を持つ

学習システムとみなすことができると考えている。さらに、(2)は定性推論の浅い知識を作成するという必要性から、(a)モジュールの持つ状態の学習と(b)状態遷移の学習に分けられる。本方式は、入力にモジュールの入出力変数名、及びモジュールの構造を与えるだけで、出力としてモジュールの各状態ごとの記述を持つ知識を作成する。得られた知識を用いると、それまで行なっていた内部の状態を解析することなく、モジュールの挙動を求めることができるので、学習をしたといえる。しかし、学習に用いる事例に相当するものは与えていない。本方式では、図10のstep 1でQuprasのエンヴィジョニング機能を用いて事例を生成している。以降の過程は、ここで得られた各事例を一般化することにより学習を行なっている。

step 2では得られた事例を入出力関係に着目して同じ概念（状態）ごとに分類し、各概念ごとに事例を一般化して、その概念を説明する仮説を求める。ここでは、各状態の入力条件が仮説にあたる。すなわち、この過程では「状態」を学習する。しかし、通常、例題に基づく学習は、一般化に用いた事例以外については、必ずしも常に真であるとは限らない。ところが、本方式はstep 1において全ての事例を求めることを前提としているので、作成された知識は、健全性が保証される。

モジュールの挙動は、状態のみでは説明できないため、step 3で「状態遷移」の学習を行なう。学習のための事例はstep 2と同様、step 1で求めた全ての状態遷移である。ただし、step 2では各状態の入出力に着目したのに対し、step 3では状態間の遷移に注目する。ここでも、全ての状態遷移が求められているので、健全な一般化が行なえる。

ただし、本方式は全ての推論対象を扱うことが

できるわけではない。Quprasは数式処理を用いて変数の範囲などを求めているため、適用範囲は数式を用いるものに限られる。また、現在非線形連立不等式（方程式を含む）を実用に耐え得る時間で解くことのできるアルゴリズムは存在しない。Quprasの数式処理システムも非線形2次不等式（方程式を含む）の一部しか解くことができないため、推論対象がこの範囲に限定されることになる。

今回の方式は、モジュール内のパラメータが全て既知のものしか扱うことができない。そこで、今後の課題としては、適用範囲を拡張するため、調整可能な未定義なパラメータを持つモジュールの知識を作成する方式を検討している。

7. 結言

定性推論を実用化するための一段階として、複数の深い知識をモジュール化して、浅い階層の知識を作成する方式を提案した。本方式は知識とすべきモジュールの範囲の指定はユーザーが行なうが、モジュールの挙動を表現するための知識の記述は自動的に行なうことができる。そして、本方式を電子回路に応用した結果、DTLとシュミット・トリガ回路の知識を作成することができた。

本方式の特徴は以下の通りである。

- (1)モジュール化した知識は入力と出力のみに着目して利用でき、推論を高速化できる。
- (2)作成した知識は、従来用いていた深い知識と同じ記述方式をとっているため、併用可能であり、多重の階層化が可能である。
- (3)不連続な変化にも対応できる。
- (4)モジュール化した知識が状態遷移の情報を持つことにより、状態遷移に順序関係を持つものに対しても入力と出力のみに着目して利用できる。
- (5)学習に必要な事例をシステム自身が生成し、健全に一般化を行なう学習システムとみなすことができる。

謝辞

本研究は、第五世代コンピュータプロジェクトの一環として、新世代コンピュータ技術開発機構（ICOT）からの委託により行ったものである。ご支援頂いた新田克巳第7研究室長に感謝致します。また、本研究に有益な指導・議論をしていただいた京都大学西田豊明助教授、研究をまとめるにあたって貴重なコメントをいただいた日立製作所基礎研究所の吉田健一研究員、ならびにシステムの試作にご協力いただいた（株）日立情報システムの川口俊治さん、土岐尚子さん、磯川澄江さんに感謝致します。

参考文献

- [Forbus 84] Forbus,K.D.:Qualitative Process Theory, Artificial Intelligence, 24, pp. 85-168 (1984).
- [Falkenhainer 88] Falkenhainer,B. and Forbus,K.D.:Setting Up Large-Scale Qualitative Models, In Proceedings AAAI-88, pp.301-306, American Association for Artificial Intelligence (1988).
- [Iwasaki 88] Iwasaki,Yumi:Causal Ordering in a Mixed Structure, In Proceedings AAAI-88, pp.313-318, American Association for Artificial Intelligence (1988).
- [Kuipers 84] Kuipers,B.: Commonsense Reasoning about Causality:Deriving Behavior from Structure, Artificial Intelligence, 24, pp. 169-203 (1984).
- [Ohki 92a] Ohki,Masaru, Sakane,Kiyokazu, Sawamoto,Jun, and Fujii,Yuichi: Enhanced Qualitative Physical Reasoning System:Qupras, New Generation Computing,10 pp.223-253 (1992).
- [Ohki 92b] Ohki,Masaru, Oohira,Eiji, Shinjo,Hiroshi, and Abe,Masahiro: Range Determination of Design Parameters by Qualitative Reasoning and its Application to Electric Circuits, In proceedings FGCS'92 (to appear).
- [岩佐 91] 岩佐英彦, 馬場口登, 手塚慶一: 定性推論におけるオブジェクト指向モデルの一考察, 平成3年度後期情報処理学会全国大会講演論文集 (3), pp.27-28, 情報処理学会 (1991).
- [大木 88] 大木 優, 藤井裕一, 古川康一: 物理法則に基づいた定性推論, 情報処理学会論文誌, Vol.29, No.7, pp.694-702 (1988).
- [西田 88] 西田豊明: 定性推論に関する最近の研究動向 (I) 基礎技術の進歩, 情報処理学会誌, Vol. 29, No. 9, pp. 1009-1022 (1988).
- [吉田 89] 吉田健一, 元田 浩: 階層的定性推論のための浅い知識の合成法, 人工知能学会誌, Vol. 4, No. 4, pp. 447-455 (1989).