

2)

# 定性推論に基づく動的システムの 因果理解に関する研究

平成7年12月

秋吉政徳

## 内容梗概

計算機技術の進歩は、複雑化する人工物を設計・構築し、運用していくためには必要不可欠となっている。これまでのところ「人間の代行」という自動化の側面の利用技術の研究開発が先行してきたが、「人間との協調」という側面の利用技術についても確立が急がれるようになってきた。人間の知的活動を支援するインタフェースあるいはヒューマン・コンピュータ・インタラクションの構築技術として、本論文では動的システムの因果理解という視点から、人間の思考に整合する機能の構築を目指し、そのための要素技術として定性推論を位置付ける。

まず、動的システムの因果性の取り扱いに関して、従来の定性推論における手法を整理し、大規模な対象への適用を困難にしている問題点を考察して、定量的解析との対応操作による定性シミュレーション手法を提案する。ここでは、対象システムにおける変数間の局所的な影響伝播の過程をシミュレーションすることによって、人間の直感に適合する因果的説明の導出を行なう。その際に、対象システムに含まれる変数の全てを用いた粒度の細かい影響伝播ではなく、人間の注目している変数群に対する影響伝播を行なう。具体的には、対象の定量モデルに含まれる変数間の依存構造に対して縮約を実行し、また定量データを変換した定性データを用いて定性シミュレーションに発生する曖昧さを解消する手法の適用結果から、提案する手法の有効性や問題点を明らかにする。

次に、因果理解という認知プロセスをメンタルモデルの形成という立場から考察し、そのために必要な機能をもとに教育・訓練システムを構成する重要な技術として因果理解支援方式を提案する。まず、認知科学からの知見に基づき、理解のための方略を整理し、次に階層型定性モデルをもとにそれらを支援する機能を構築する。階層型定性モデルは、対象システムの物理階層を自然に表現したものであり、学習者の視点がモデルの構成に反映されるとともに、それを用いた定性シミュレーションが実行される。このような階層型定性モデルならびに定性シミュレーションは、学習者のメンタルモデルの部分的表象であるとの立場であり、納得のいく因果的説明が得られるまで学習者がツールとして用いることを目指している。このような「表象-吟味のループ」によってメンタルモデルの形成を支援する。発電プラントに適用した結果を述べる。

最後に、事象間の因果理解が前提となっている故障木解析について述べる。故障時点と観測時点が異なるうえに故障状態が変化する大規模な対象については、専門家といえどもしばしば故障木解析が困難となる。このような問題に対して、設計情報をもとにしたモデル・ベース推論を実行すれば、システムティックな処理が可能となることを示す。状態の変化を取り扱いつつ観測兆候から可能性のある状態を推論する際に、定性推論が応用できることを示し、電力保護リレーの故障木解析に適用した結果を述べる。

# 定性推論に基づく動的システムの 因果理解に関する研究

## 目次

第1章 序論	1
1.1 本研究の背景と目的	1
1.2 従来の研究ならびに問題点	3
1.3 本論文の構成	5
第2章 定性推論による動的システムの解析	7
2.1 緒言	7
2.2 定性推論で用いられる基本的な表現ならびに考え方	8
2.3 架空の因果性を用いた挙動解析	12
2.4 構造方程式を用いた因果解析	15
2.5 結言	20
第3章 定量的解析との対応操作による定性シミュレーション手法	21
3.1 緒言	21
3.2 従来の研究と本手法のアプローチ	22
3.3 定性/定量的解析の対応づけの概要	24
3.4 因果順序づけを利用した定性モデルの構築	26
3.4.1 定性モデルの依存構造	26
3.4.2 定性モデルにおける定性関係	28
3.4.3 定性関係の割り当て	28
3.5 定量データから導入される解釈のための時間	29
3.5.1 定量データの変換	30
3.5.2 解釈のための時間の生成	31
3.6 解釈のための時間を組み込んだ挙動シミュレーション	33
3.6.1 挙動シミュレーション	33
3.6.2 因果的説明の導出	39
3.7 結言	40
第4章 発電プラントの定性シミュレーション	41
4.1 緒言	41
4.2 発電プラントの定量モデル	42
4.3 発電プラントの依存構造	44
4.4 発電プラントの縮約された依存構造	52

4.5 負荷変化に対する因果的説明 . . . . .	56
4.6 定性モデルの修正例 . . . . .	58
4.7 結言 . . . . .	62
第5章 定性推論に基づくプロセス現象の因果理解支援方式 . . . . .	63
5.1 緒言 . . . . .	63
5.2 因果理解支援システムの設計思想 . . . . .	66
5.2.1 認知科学からの知見 . . . . .	66
5.2.2 因果理解支援方式 . . . . .	67
5.3 因果理解支援システムの構築 . . . . .	70
5.3.1 階層型定性モデル . . . . .	70
5.3.2 定性シミュレーション . . . . .	74
5.3.3 ユーザインタフェース . . . . .	78
5.4 適用例 . . . . .	81
5.5 結言 . . . . .	85
第6章 定性推論を用いた因果伝播による故障木作成手法 . . . . .	86
6.1 緒言 . . . . .	86
6.2 故障木作成手法 . . . . .	88
6.2.1 対象の状態変化をとらえた故障木作成の基本的な考え方 . . . . .	88
6.2.2 対象のモデルおよび推論処理 . . . . .	90
6.2.3 モデルにおける故障の定義およびその処理 . . . . .	93
6.2.4 ソフトウェア構成 . . . . .	94
6.3 電力保護リレーへの適用 . . . . .	95
6.3.1 保護リレーの設計情報およびモデルの生成 . . . . .	96
6.3.2 制約伝播処理ならびに状態遷移処理 . . . . .	99
6.3.3 動作結果 . . . . .	102
6.4 結言 . . . . .	107
第7章 結論 . . . . .	108
謝辞 . . . . .	111
参考文献 . . . . .	112
付録 . . . . .	120
著者の研究発表一覧 . . . . .	136

# 1 序論

## 1.1 本研究の背景と目的

近年の計算機利用技術のめざましい発展は、工学分野における設計、診断、制御、教育・訓練などのさまざまな実問題にたいして、有効な解決手段を提供してきている。例えば、数値処理の側面では、CADによる設計の効率化、複雑な制御アルゴリズムのリアル・タイム実行による制御の高度化、精緻な数値シミュレータによる訓練の逼真化などがあげられる。一方、記号処理の側面でも、専門家の知識や対象物に関する構造、機能、因果関係などを表現したモデルをもとに、設計物の不具合の検出、数学モデルの記述が困難な制御対象に対する人間の判断の取り込み、仮説生成・検証のような人間の情報処理過程を模した診断など多くの手法が提案されている。さらに、対象となるシステムの複雑化に伴って、これら数値処理と記号処理の融合を図ることにより実問題を取り扱うといった計算機の高度利用がいつそう望まれている。

システムの複雑化に対する有効な手法として、挙動解析を記号処理によって実行し、それとともに因果的な説明を導出することを一つの目的としてかける定性推論が注目されている。定性推論は、1970年代の後半に「人間の定性的な思考、特にダイナミクスに関する理解の形式を明らかにし、計算機上にモデル化する」という動機から研究が始められた<sup>[7,8,43]</sup>。ここでいうダイナミクスに対する理解の形式化は、存在論 (Ontology) の立場からの機構理解を意味しており、対比されて用いられる認識論 (Epistemology) の立場とは大きく異なる<sup>[75]</sup>。存在論的アプローチでは、不変なものとしての対象世界の存在の形式をまず規定し、その形式からわれわれの知的活動を構成するとされる。一方、認識論的アプローチでは、われわれの対象世界に対する心的な構造化をもとにして、問題解決の際の知識や認識を形式化するとされる。このような哲学的論議も交え

て、定性推論については、定性モデル<sup>[1, 6, 12, 15, 16, 17, 24, 74]</sup>、定性シミュレーション<sup>[5, 11, 19, 20, 29, 30, 31, 32, 33, 34, 41, 46, 59, 62]</sup>、因果解析<sup>[25, 26, 27, 28, 52, 70, 82]</sup>、定量解析との融合<sup>[37, 38, 44, 53]</sup>、各種の問題解決への応用<sup>[14, 18, 21, 58, 60, 63, 81]</sup>に大別される項目で研究が活発に展開されている。しかしながら、大規模な実システムへの適用といった観点からは現時点では十分な検討が行なわれておらず、この問題をいかに解決するかということが重要な課題となっている。

一方、このような計算機利用技術を考えるにあたり、計算機をインテリジェント化するための方法論とともに、最近では計算機と人間とのインタフェースが重要視されている<sup>[40, 71]</sup>。すなわち、計算機と人間がそれぞれ得意とする情報処理を役割分担して、困難な問題の解決を達成する枠組みが注目され、次の観点からのインタフェースが求められている。人間の思考には、対象への抽象化や注視化などの操作によって、問題解決にとって本質的と考えられる情報から次第に精緻化していく面が多くみられる。すなわち、これからのインタフェースとしては、このような人間の思考に整合する機能を備えている点が望まれている。

本研究は、先に述べた定性推論の研究動向をふまえて、動的システムの因果理解という側面から人間の思考に整合する機能を備えた、計算機と人間とのインタフェースの構築へ取り組んだものである。具体的には、対象として電力システムを取り上げ、「運用」、「教育」、「故障診断」に関する問題を議論する。「運用」に際しての問題は、運転員は通常運転時における状態把握のために膨大なデータを因果的に解釈する必要があるにもかかわらず、最近の複雑化したシステムでは非常に困難となっている点である。この解決に向けて、定性推論における因果性の取り扱いならびに問題点を整理し、因果的説明を導出するための定性シミュレーション手法を提案することは重要である。「教育」に際しての問題は、対象システムの振る舞いに関する因果的説明の獲得プロセスが重要にもかかわらず、これまでの教育・訓練環境では十分な機能が提供され

ていない点である。このために、学習者自身による因果的説明の獲得プロセスを支援する教育・訓練環境を構築することは、非常に意義深いものであると考える。「故障診断」に際しての問題は、観測兆候から因果関係をさかのぼって故障原因候補を求める故障木解析において、観測時点と故障時点の異なる場合には専門家といえども解析が困難となっている点である。このために、状態遷移がともなう故障木解析について議論し、さらに実用的な観点から支援システムとしてどのような機能が必要であるかを考察することは重要である。

## 1.2 従来の研究ならびに問題点

因果性そのものは、従来から人間の思考における時間という属性の認識の問題<sup>[83]</sup>とも関係づけられ哲学的な議論も多くなされているが、一般には「二つの事象XとYの間で、Xが時間的にYに先行し、XとYが構造的に近くに位置し、XからYが一意的に導出される」という観点から、動的システムの因果性ならびに因果理解の研究は行なわれている。特に工学分野では、「XからYが一意的に導出される」という表現に対して、「量の変化」を前提とした因果性を意味していると考えるのが自然である。

ところで、定性推論による動的システムの解析では、実数領域を有限個に分割した離散領域の上で、「量の変化」に基づいて記号処理が実行される。この記号処理による解析の利点の一つは、対象に関する可能な限りの挙動を導出すると同時にその因果的説明が得られるという点である。この際のモデルならびに推論方法を記述する段階で、基本原理の一つとして「構造から挙動へ、そして機能へ」という枠組み<sup>[9, 12]</sup>が、一般に認められている。この基本原理は、先ず構造記述と挙動推論を明確に分離し<sup>[9]</sup>、さらに個々の挙動をもとに機能的な観点からの記述にまとめあげる推論<sup>[12]</sup>も分離することによって、できるだけad hocな知識の混入を排除して、客観的で形式的な推論の体系を規定している。定性推論の初期の研究では、このような構造記述ならびに挙動推論の方法がいろ

いろと議論されたが、因果性を構造記述に明示的に含める<sup>[17, 29, 30]</sup>、あるいは含めない<sup>[12, 52]</sup>という立場の違いが生じている。また、挙動推論とは独立に因果性に関わる議論として、対象を非因果的に記述した方程式に含まれる変数間の依存関係を解析し、変数間に順序づけを行なうことにより因果構造をシステムティックに決定する研究<sup>[25, 26, 27, 28]</sup>もある。対象が複雑な場合にも解析が可能な反面、平衡方程式の場合には同時的に値が決まることから、仮に変数の値の求まる順序が決まったとしても、それに因果性をみいだすことは不自然といわざるを得ない。このことは、因果性を構造記述からのみたどることは不具合が生じる場合もあることを示している。

因果理解という認知プロセスは、認知科学の分野で展開されてきた「メンタルモデル」の研究<sup>[22]</sup>と密接に関係している。「メンタルモデル」とは、「人間が対象を理解しようとする際に、頭の中にいづく対象の心的表象である」と一般に定義される。このようなメンタルモデルについては、「不安定な存在で、しばらく使用しないと忘れ去ってしまう」という非常に曖昧なものであることが指摘されているが<sup>[39]</sup>、一方ではその形成自体の重要性が認められている。計算機利用による教育（CAI：Computer-Assisted InstructionもしくはITS：Intelligent Tutoring Systemと呼ばれる）に関する研究の中では、学習プロセスをメンタルモデルの形成としてとらえる立場から重視し、意味ネットワークやシミュレーションなどを用いたさまざまなアプローチによってシステム構築が行なわれている<sup>[49]</sup>。特に、因果理解は、動的システムに関するメンタルモデルの形成に大いに役立っている。しかしながら、因果理解を積極的に促進し、その結果としてメンタルモデルの形成を支援しようとする枠組みに関しては、十分な議論がなされているとはいえない。

故障診断においては、観測された異常兆候に対して、故障原因候補を推定し、さまざまな検証作業を通してそれらの候補を絞り込む際には、対象の構成要素間の因果関係を正確に把握する必要がある。このような作業を効率的に行なう

手法として、故障木解析がある。故障木解析とは、観測された事象から故障を引き起こす原因となる事象を順次調べることにより、考えられる故障原因候補を求める作業である。その作業過程でそれぞれの事象間の論理的関連が明確となる利点から、故障診断においてはしばしば用いられる。しかしながら、対象の規模が大きく、また時間的に故障状態が変化する場合には、専門家といえども故障木解析が困難なことが指摘されている。このような故障木解析を実行するシステム構築は、実用的な観点からも非常に重要であり、モデル・ベース診断として研究もなされている<sup>[78, 79, 80]</sup>。しかしながら、従来は故障状態は変化しないという仮定のもとに議論されている場合が多く、実用的には故障が発生した時点と観測時点が同時点でない場合の処理を考える必要がある。

### 1.3 本論文の構成

本論文は、以上に述べたそれぞれの課題について次の流れにそって取り組み、まとめたものである。第2章においては、定性推論で用いられる基本的な表現ならびに考え方、因果性の取り扱いに関する代表的な手法<sup>[12, 25, 26, 27, 28]</sup>を概説し、それらを通して因果的説明を導出する際の問題点を論じる。第3章においては、定性推論の大規模な系への適用を可能とするために、第2章における考察に基づいて、定性推論における構造記述ならびに挙動推論を定量的解析と関連づける手法を新たに提案する。第4章においては、第3章での提案手法から発電プラントの現象の因果的説明を導出し、提案手法の有効性を示す。第5章においては、認知科学からの知見をもとにメンタルモデルの形成支援の枠組みを論じ、その枠組みの中での定性推論の利用を考察し、発電プラントの教育・訓練システムの構築に取り組む。第6章においては、故障木解析における故障時点と観測時点が異なることに起因する問題を考察し、定性推論の利用による解決方法を議論する。提案する手法を電力保護リレーの故障木解析に適用し、さらに故障木解析支援システムとしての機能についても考察する。第7章においては、

2章から6章までで得られた結果をまとめるとともに、今後の研究課題について述べる。

## 2 定性推論による動的システムの解析

### 2.1 緒言

本章では、第3章で取り組むべき課題を明らかにするために、「人間の定性的な思考、特にダイナミクスに関する理解の形式を明らかにし、計算機上にモデル化する」という観点から行なわれてきた定性推論の研究を概説する。定性推論は、従来から数式を用いたモデル化を通して解析的あるいは数値的に取り扱われてきたダイナミクスに対して、抽象化や注視化などを通して因果的説明を試みる人間の思考を、記号処理によって表現する方法論の確立を目指してきた。その中心課題は、実数領域を有限個に分割した離散領域の上で記述される変数の「定性的変化」に基づいて、対象モデルに対する定性演算を行ない、挙動推論ならびに因果解析を形式化することにある。

定性演算は、定性変数間の演算としての加法や乗法を定義したものである。定性変数は、もともと連続領域の変数の値域に対して境界標と呼ばれる離散値とそれらで記述される開区間をもとに定性化した値域を用いて表現され、一般に定量的な情報は含んでいない。このような定性化した値域を記号処理することの利点としては、「量の情報が不完全な場合」にも挙動を求めることが可能となる点が挙げられるが、一方では「量の情報の欠落」による演算の曖昧性から、現実には起こりえない挙動を推論してしまう欠点もある。

定性演算は2変数間の一般的な演算則であり、挙動推論や因果解析のためには対象モデルの記述が必要である。このようなモデル記述に関する議論は、定性推論の基本原理の一つとして初期の研究段階に取り組み、de KleerとBrownが提案した「構造から挙動へ、そして機能へ」という枠組み<sup>19,12)</sup>が、一般に認められている。この基本原理は、先ず構造記述と挙動推論を明確に分離し、さらに個々の挙動をもとに機能的な観点からの記述にまとめあげる推論を分離する

体系を規定している。de KleerとBrownはさらに、因果性をあらかじめ構造記述に明示的に含めることなく挙動推論と同時に解析する手法として、「架空の因果性」を用いた方法を提案し、圧力レギュレータや電気回路の挙動解析に適用した<sup>[12]</sup>。

また、因果解析についてはIwasakiとSimonが挙動推論とは別に、対象を非因果的に記述した方程式（構造方程式と呼ぶ）に含まれる変数間の依存関係を解析し、変数間の順序づけを行なうことにより因果構造をシステムティックに決定する手法を提案し、プロセス・プラントなどに適用している<sup>[25, 26, 27, 28]</sup>。

以下、2.2で定性推論で用いられる基本的な表現ならびに挙動推論を実行する際の中心となる考え方を説明する。2.3では、因果的説明の導出方法として「架空の因果性」を用いた挙動解析手法の特徴を、2.4では、与えられた構造方程式を用いて挙動解析とは独立に因果的説明を導出する「因果順序づけ」の手法を説明し、それぞれの問題点を考察する。

## 2.2 定性推論で用いられる基本的な表現ならびに考え方

動的システムを状態変数 ( $x \in R^n$ ) で表わすときに、これに対して定性推論で用いられる基本的な表現は、以下のようにまとめられる。

### ・量の定性化

実数上の連続領域で定義される変数  $x_i$  ( $i=1, \dots, n$ ) の値域を、境界標と呼ばれる離散値の集合 ( $L_i = \{l_{i,1}, \dots, l_{i,m_i}\}$ , ただし,  $l_{i,1} = -\infty$ ,  $l_{i,m_i} = +\infty$ ) から有限個の開区間に分割した領域により、定性変数  $[x_i]$  は表現される。また、変数の微分値の表現として、 $dx_i/dt$  が '+', '0', '-' の場合に対応して「増加, 安定, 減少」という記号を用いる。

定性領域を  $Q$  とすると、

$$Q = \{ l_{i,1}, (l_{i,1}, l_{i,2}), l_{i,2}, \dots, l_{i,m_i} \} \quad (2.1)$$

となり、したがって、動的システムの定性領域  $Q$  は、

$$Q = \prod_{i=1, \dots, n} Q_i \quad (2.2)$$

と表わされる。

### ・定性演算

定性変数  $[x_i]$  どうし、あるいは定性微分値  $dx_i/dt$  どうしの二項演算として、表2.1のように加法と乗法が定義されている。ただし、ここでは境界標として、 $\{-\infty, 0, +\infty\}$  をとり、 $(-\infty, 0)$  を  $[-]$ ,  $(0, +\infty)$  を  $[+]$  として表中に記述している。

表2.1 定性演算としての加法ならびに乗法

$[x]+[y]$	$[-]$	$[0]$	$[+]$	$[x] \times [y]$	$[-]$	$[0]$	$[+]$
$[-]$	$[-]$	$[-]$	$?$	$[-]$	$[+]$	$[0]$	$[-]$
$[0]$	$[-]$	$[0]$	$[+]$	$[0]$	$[0]$	$[0]$	$[0]$
$[+]$	$?$	$[+]$	$[+]$	$[+]$	$[-]$	$[0]$	$[+]$

表中 '?' が示す意味は、定性的に不定であるとされる演算であり、このような曖昧さが演算に含まれることが、可能性のある挙動を全て求めることができるという利点と、現実には起こりえない挙動を推論してしまう欠点を生じさせている。

### ・定性的状態に基づく時間軸の定性化

定性的状態は、瞬時的状態と持続的状态に分類してとらえられる。瞬時的状態とは、 $[x] = (\dots, [x_i], \dots)$  の中に少なくとも一つは、

$$[x_i] = l_i \text{ かつ } \left[ \frac{dx_i}{dt} \right] \neq \text{安定} \quad (2.3)$$

となる変数が含まれていると定義される。逆に、そのような変数がない場合には、持続的状態と定義される。このような立場から動的システムの挙動は、図2.1が示すように定性的には瞬時的状態と持続的状態が交互に繰り返しているものであるととらえられる。この時の、状態の移り変わる時点は定性的状態を解釈する上で重要と考えられるが、あらかじめ与えることはできず、定性的状態が求められた後に、 $\{t_0, \dots, t_n\}$ のように決定されるものである。

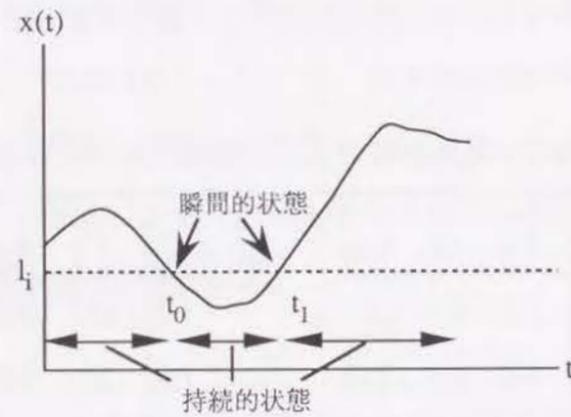


図2.1 時間軸の定性化

### ・定性的連続性

変数 $x_i \in R$ の連続性は、定性化された場合には次のように扱われる。

例えば、 $[x_i] = (l_{i-1}, l_i)$  かつ  $\left[ \frac{dx_i}{dt} \right] = \text{増加}$  ならば、次の時点で  $[x_i] = (l_i, l_{i+1})$  にいきなり変化することはなく、 $[x_i] = l_i$  にまず変化する。

以上の基本的な表現を用いて動的システムの構造記述と挙動推論を形式化するという事は、「特徴量としての定性変数を決定し、その定性変数間の関係に基づいて連続性などの制約のもとに定性変数の値の推移を推論すること」に

ほかならない。このことにより、図2.2が示すような定量的解析との類似性のもとに定性推論を位置づける考え方が、一般に認められている。

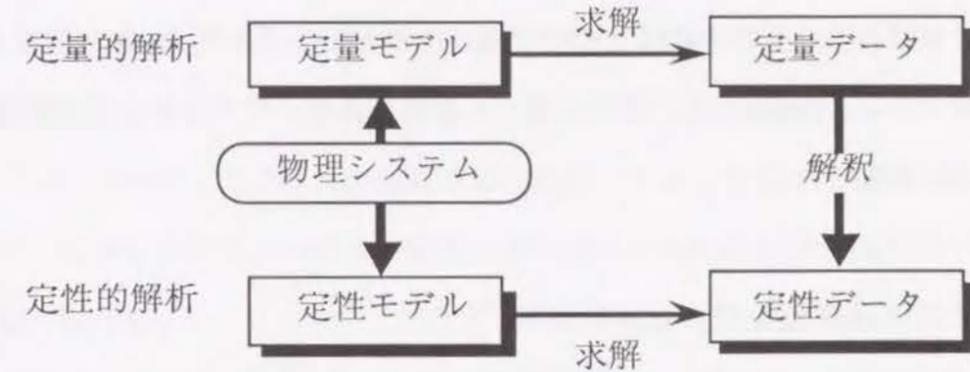


図2.2 定性/定量解析の類似性

ところで、構造記述に関する考え方は大別すると、Device中心の記述<sup>[12]</sup>、Process中心の記述<sup>[17]</sup>、Constraintsに基づく記述<sup>[30]</sup>の三つになる。これらは、対象に関して何を中心にとらえてモデル化するかによる違いであるが、結果として得られる記述には因果性を明示的に含まないものと含むものとの違いが生じている。すなわち、Device中心の記述ではあらかじめ因果性を明示的に準備する必要がなく、Process中心の記述とConstraintsに基づく記述では因果性を理解した上で構造記述を行なう必要がある。対象が大規模で複雑になれば、後者のようにあらかじめ因果性を理解することは困難であり、前者の立場が望ましく考えられる。ただし、このことはあくまでも構造記述に関する点であり、Device中心の記述が、挙動推論と同時に因果的説明の導出を保証するものではない。

挙動推論に関しては、瞬時的状態と持続的状態の交互の繰り返しという考え方に基づき状態間の遷移規則がまとめられている<sup>[30]</sup>。この規則に基づいて、各々の定性変数に関して先ず現在の定性値から遷移可能な定性値を求め、次に定性変数間の関係に基づき可能な組み合わせを決定するという「生成-テスト法」

により、挙動推論の実行が可能となる。しかしながら、その場合には因果的な説明はえられない。これに対して、ある定性変数の状態が変化したときに、その定性変数と同時的制約関係にある定性変数への影響を因果伝播という形でとらえて挙動推論を実行すれば、その伝播をたどることにより因果的説明をえることができる。Device中心の記述に基づく挙動推論は、このような因果伝播の考え方を提案した。

### 2.3 架空の因果性を用いた挙動解析<sup>[12]</sup>

de KleerとBrownが提案した構造記述ならびに挙動推論は、Device-centered Ontologyと呼ばれる考え方に基づいている。これは、対象の構成デバイス（コンポーネント・モデル）を最小の記述単位として、それらのデバイスの接続関係（デバイス・トポロジー）から、構成デバイスに含まれる状態変数間に成立する関係を合流型定性微分方程式（Confluence）と呼ぶ形で求める。合流型と呼ばれる由縁は、デバイスどうしは導管によって結合され、導管どうしの合流する点をターミナルと呼び、ターミナルにおいてはキルヒホッフの法則と同じ考えで変数間の関係がとらえられるからである。この点から状態変数としては、電圧のような基準点をもつ圧力や電流のような流れのタイプの変数が扱われる。

以上の考えに基づく挙動解析は、状態変数の変化が伝播することによって実行される。本来同時的制約関係にある変数間に、このような伝播の考え方を導入したときに、de KleerとBrownは「架空の因果性」と呼んだ。このことは、「人間が事象Xと事象Yの間を因果的に理解する際に、時間軸にそって考える」という立場から、たとえ同時的に決定される事象であっても、変化が伝播された部分とこれから伝播される部分に分けてとらえるという考え方である。ただし、定性演算のために伝播が行き詰まることもあり、その際には次に挙げる三つのヒューリスティクスを用いて伝播を続行している。

#### (i) コンポーネント・ヒューリスティクス

コンポーネントの一端の圧力型変数が変化することがわかり、かつそのほかの値の変化に関する情報が得られない場合には、コンポーネントはそれらの未知の値の変化が無視できるかのように振る舞う。

#### (ii) 導管ヒューリスティクス

コンポーネントの圧力型変数の値が決定できないとき、コンポーネントにつながれた導管への流れ型変数の値の変化にその圧力型変数の値の変化は一致する。

#### (iii) 合流ヒューリスティクス

三つ以上の変数を持つコンポーネントにおいて、値の変化のわかっていない変数に関しては、そのうちの一つを除いて残りの変数はあたかも値の変化が0であるかのように扱い、値を伝播する。

これらの因果伝播ならびにヒューリスティクスについて、de KleerとBrownが取り上げた例をもとに簡単に説明する。

#### [圧力レギュレータの例]

図2.3が示すような圧力レギュレータは負のフィードバック特性を有しており、流入側の圧力が上昇したあとの伝播の様子をたどってみる。

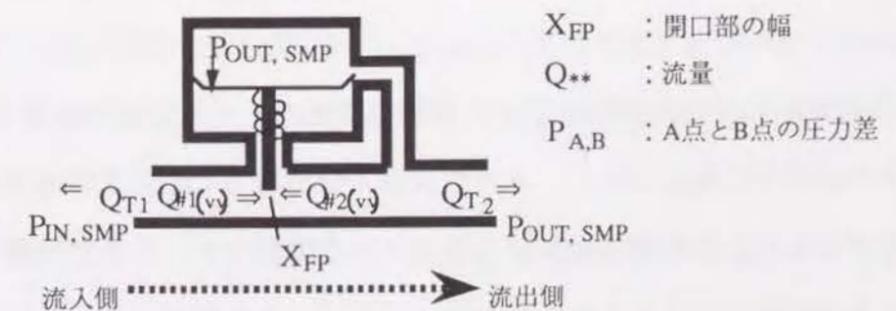


図2.3 圧力レギュレータ

なお、Confluenceとしては、次のものが用いられている。

$$\partial P_{IN, SMP} = \partial P_{IN, OUT} + \partial P_{OUT, SMP} \quad (2.4)$$

$$\partial P_{IN, OUT} - \partial Q_{\#1(vv)} + \partial X_{FP} = 0 \quad (2.5)$$

$$\partial Q_{\#1(vv)} + \partial Q_{\#2(vv)} = 0 \quad (2.6)$$

$$\partial X_{FP} + \partial P_{OUT, SMP} = 0 \quad (2.7)$$

$$\partial Q_{T1} + \partial Q_{\#1(vv)} = 0 \quad (2.8)$$

$$\partial Q_{T2} + \partial Q_{\#2(vv)} = 0 \quad (2.9)$$

(1)  $\partial P_{IN, SMP} = [+]$

(2) (2.4)式において、 $\partial P_{IN, OUT}$  と  $\partial P_{OUT, SMP}$  の値の伝播に行き詰まる。このような場合に、コンポーネント・ヒューリスティクスが用いられ、

$$\partial P_{IN, OUT} = [+]$$
 をえる。

(3) (2.5)式において、 $\partial Q_{\#1(vv)}$  と  $\partial X_{FP}$  の値の伝播に行き詰まる。このような場合に、合流ヒューリスティクスが用いられ、 $\partial Q_{\#1(vv)} = [+]$  をえる。

(4) (2.6)式より  $\partial Q_{\#2(vv)} = [-]$ 、(2.9)式より  $\partial Q_{T2} = [+]$  をえる。

(5) ここで伝播が行き詰まると、弁に対する導管ヒューリスティクスが用いられ、 $\partial Q_{\#2(vv)} = [-]$ より  $\partial P_{OUT, SMP} = [+]$  をえる。

(6) (2.7)式より、 $\partial X_{FP} = [-]$ をえる。

以上の(1)から(6)を自然言語を用いて再構成すれば、「入力側の圧力の上昇により、弁の両側で圧力差が生じ、その結果流入量が増加する。この結果、センサの圧力が高まり、弁の開口部が狭くなる。」と解釈され、弁の開口部の狭まりが流入量の増加を抑えるという負のフィードバックを説明することになる。なお、最終的な挙動推論の結果は、このような因果伝播を繰り返して行ない、状態遷移図の形で提供される。

圧力型や流れ型の変数でコンポーネントが表現される場合には、このようなヒューリスティクスを用いて因果伝播を行ない、挙動推論と同時に因果的説明の導出がある程度可能なことが今までに示されている。このような因果伝播という考え方は、われわれの因果理解に近いものである。ただし、実問題に適用するためにヒューリスティクスが十分であるかの議論が不十分かつ困難であり、因果伝播を実行するより強力な手法が必要とされる。

## 2.4 構造方程式を用いた因果解析<sup>[25, 26, 27, 28]</sup>

IwasakiとSimonが提案した「因果順序づけ」の手法は、対象の構造記述として構造方程式と呼ぶ数式表現を用い、この数式に表現された変数間の依存構造を解析する。構造方程式とは、各変量の間で成立する物理法則やその適用条件をもとに作成された方程式であり、それぞれの式が対象の機能を構成するメカニズムを表現するとされている。平衡方程式ならびに微分方程式に対して、以下の定義がなされている。

### 定義2.1 平衡構造

平衡方程式の集合Sにおいて、次の条件を満足する場合に平衡構造という。

(1) Sに含まれる任意のk個の方程式には、少なくともk個の変数があらわれる。

(2) 任意のk個の方程式において、 $m(\geq k)$ 個の変数があらわれているとき、任意の $(m-k)$ 個の変数の値がわかると、この方程式は一意に解ける。

### 定義2.2 完結構造

平衡構造となっているk個の方程式で、k個の変数があらわれるとき、完結という。

### 定義2.3 極小完結集合

平衡構造の部分集合で、それ以外の完結集合を真部分集合として含まない

ものを極小完結集合という。

#### 定義2.4 0次の極小完全集合

完結構造に含まれる全ての極小完結集合の集まりを、0次の極小完全集合という。

#### 定義2.5 派生構造

完結構造Aに対して、極小完全集合A'があたえられるとA'を一意に解くことができ、その値を代入した(A-A')は完結構造となり、これをAから得られる派生構造という。0次の極小完全集合から繰り返すことにより、1次、2次、...、k次の派生構造が導かれる。

#### 定義2.6 k次の完全集合

k次の派生構造の極小完全集合を、k次の完全集合という。

#### 定義2.7 外生変数ならびに内生変数

k次の完全集合に含まれる変数 $x_i$ が、kより低次の完全集合に含まれるときこれを外生変数といい、含まれないとき内生変数という。

#### 定義2.8 完結な微分構造

n変数を含んだn個の一階の微分方程式の集合Dにおいて、次の条件を満足する場合に完結な微分構造という。

- (1) Dに含まれる任意のk個の方程式には、少なくともk個の一階の微分変数があらわれる。
- (2) 任意のr個の方程式において、 $r(\geq k)$ 個の一階の微分変数があらわれているとき、任意の $(r-k)$ 個の一階の微分変数の値がわかると、この方程式は一意に解ける。

平衡方程式と微分方程式が混在した場合の「因果順序づけ」を構成する処理を、以下に説明する。

#### [因果順序づけ]

n変数n方程式の集合Mの中で、微分方程式の集合をDynamic(M)、平衡方程式の集合をStatic(M)とする。ただし、Dynamic(M)の要素は、 $x_i' = f_i(x_1, \dots, x_n)$ という形式の表現に変換されているとする。

- (1) Dynamic(M)においては、左辺にある変数の値は右辺の変数の値によって決定されるので、 $x_i'$ に対して右辺のn変数から依存リンクを張る。
- (2)  $x_i'$ から $x_i$ は計算されるので $x_i$ は $x_i'$ に依存していると考え、積分という意味での依存リンクを $x_i'$ から $x_i$ の間に張る。この時に、 $x_i = \text{constant}$ という式をStatic(M)に加える。
- (3) Static(M)においては、極小完全集合を順次求める過程で変数間に順序づ

けがなされる。n次 ( $n \geq 1$ ) の極小完全集合 $S_n$ は、 $(S_n - \bigcup_{i=0}^{n-1} S_i)$ において $S_{n-1}$ に含まれる全ての変数を既知だとした場合の方程式の集合をもとに導出される。このときに $S_n$ の変数の中の外生変数から $S_n$ の内生変数へ依存リンクを張る。

この処理の(3)で用いられる“k次の完全集合”に含まれる変数群は、同時的に値が決定されるとしているので、仮に変数の値の求まる順序が決まったとしても、それに因果性をみいだすことは不自然といわざるを得ない。すなわち、de Kleerらが「架空の因果性」として求めたものを、「因果順序づけ」では議論していない。

次に、このアルゴリズムの適用例を示し、その中で議論された問題点を説明する。

#### [バスタブの例]

図2.4が示すようなバスタブは、流入量の変化が流出量の変化となって現われる簡単な例であるが、構造方程式を(a)平衡状態のみから記述する場合と、(b)非

平衡状態を含めて記述する場合で、「因果順序づけ」の結果がわれわれの直感に基づく理解とくい違いが生じる。図2.5と図2.6にそれぞれの場合の因果順序づけの結果をグラフで表現する。図中矢印のアーチが因果的な依存関係を示している。

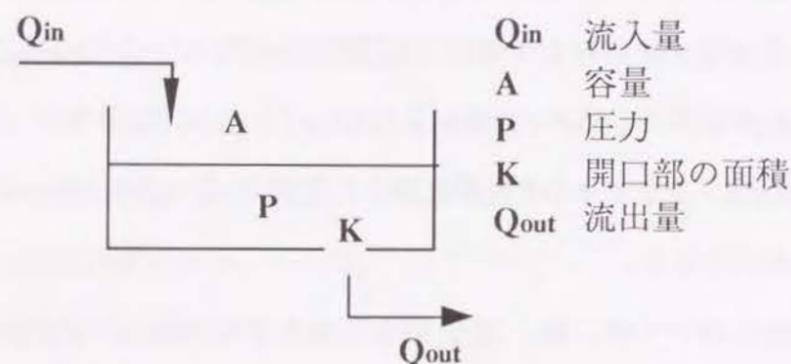


図2.4 バスタブ

(a) 平衡状態に基づく構造方程式ならびに因果順序づけ

$$Q_{out} = K P \quad (2.10)$$

$$A = c_1 P \quad (2.11)$$

$$Q_{out} = Q_{in} \quad (2.12)$$

$$Q_{in} = c_2 \quad (2.13)$$

$$K = c_3 \quad (2.14)$$

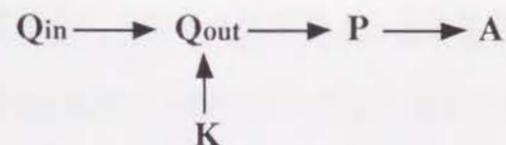


図2.5 因果順序づけ (平衡状態)

(b) 非平衡状態に基づく構造方程式ならびに因果順序づけ

$$Q_{out} = K P \quad (2.15)$$

$$A = c_1 P \quad (2.16)$$

$$\frac{dA}{dt} = Q_{in} - Q_{out} \quad (2.17)$$

$$Q_{in} = c_2 \quad (2.18)$$

$$K = c_3 \quad (2.19)$$

$$A = c_4 \quad (2.20)$$

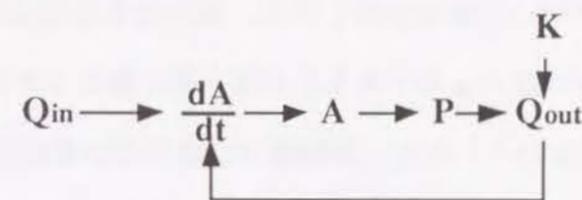


図2.6 因果順序づけ (非平衡状態)

明らかに図2.5が示す因果順序は、われわれの直感 (AやKから $Q_{out}$ が決定される) に合わない。これは、平衡状態だけを記述した式からは完全な因果性の情報を得られない場合があることを示している。この点に関して、例えば熱の伝導における非可逆性などの物理法則固有の仮定のように、方程式の上からは失われた情報は、「因果順序づけ」だけからは回復できないということが指摘され、物理法則固有の仮定に基づく因果性をヒューリスティクスとして表現し、変数間の順序づけを行なう際に、よりきめ細かな導出方法で因果構造を求める研究<sup>[82]</sup>もある。

以上のように、定量的解析でしばしば用いられる数式をもとに、「因果順序づけ」を実行することにより、変数間の依存構造をシステムティックに導出することができる。しかし、その依存構造が挙動に対するわれわれの因果理解との整合性を保証するものではなく、実挙動との対比による数式 (構造記述) への何らかのフィードバックが必要とされる。

## 2.5 結言

本章では、定性推論で用いられる基本的な表現を説明し、さらに定量的解析との類似性の考え方から構造記述や挙動推論の形式化が行なわれている点を説明した。特に、因果性を議論した代表的な研究として、「架空の因果性を用いた挙動解析」と「構造方程式を用いた因果解析」について例をもとに説明し、それぞれの問題点を明らかにした。

「架空の因果性を用いた挙動解析」では、局所的な因果伝播という考え方がわれわれの因果理解に近いものでありながら、用いられるヒューリスティクスの十分性が検証できないことから、実問題への適用がいまだ行なわれていない。しかしながら、これは一般的な議論を行なう場合の問題であり、曖昧さを解消できる情報が得られれば、「二つの事象XとYの間で、Xが時間的にYに先行し、XとYが構造的に近くに位置し、XからYが一意的に導出される」という観点から因果性を導出する、十分意義深いアプローチであると考えられる。

「構造方程式を用いた因果解析」では、方程式に含まれる変数間の依存構造のシステムティックな導出という点で、実問題への適用は可能である。ただし、依存構造そのものに挙動の因果的説明を完全に求めることはできない。

本論文の第3章においては、以上の手法の利点を取り込み、かつ欠点を補うという立場で定性推論を実問題へ適用するために、定性推論における構造記述ならびに挙動推論を定量的解析と関連づける手法を新たに提案する。

## 3 定量的解析との対応操作による定性シミュレーション手法

### 3.1 緒言

定性推論における構造記述ならびに挙動推論は、実数上の連続領域で定義される変数を定性化した定性変数をもとに行なわれる。実際には、

- (1) 定性変数を決定,
- (2) 定性変数間どうしの関係を記述,
- (3) 遷移可能な定性値の組み合わせを導出,

という三点をどのように記述するかという問題になる。この際の構造記述の観点として、Device中心の記述、Process中心の記述、Constraintsに基づく記述などが提案されている。例えば、ENVISION<sup>[12]</sup>やQSIM<sup>[30]</sup>では定性変数間の制約として定性微分方程式が用いられ、QPE<sup>[17]</sup>では定性変数間の関係をプロセスとして含んだシナリオモデルを記述する。挙動推論は、これらの定性変数間の関係に基づいて、初期状態からの遷移可能な定性値を求める。ただし、この挙動推論を実行する際に、定性演算による曖昧さから実世界では起こりえない状態が含まれてしまうことが、はやくから指摘されている。この問題に対しては、例えば量のオーダ<sup>[41]</sup>や位相面上での大局的な制約<sup>[36]</sup>、あるいは高階の導関数<sup>[11]</sup>などの新たな制約を加えて、解消を試みる研究が行なわれている。また、最近では定量的解析と定性的解析の相補的な点に着目して、対象システムの大域的挙動や定常解の性質を明らかにしたり<sup>[37,38,44,53]</sup>、定量モデルが複雑な大規模系における問題解決<sup>[60]</sup>を行なおうとする試みもある。特に、Forbus等の提案する手法<sup>[19,20]</sup>は、定性モデルとそれぞれの定性関係に対応して定量的に記述された数値計算モデル・ライブラリ (math-model library) を入力として、定性的解析の挙動とともにその挙動を厳密に計算する定量モデルを自動的に作成するという観点から、定性的解析と定量的解析を融合するという点で非常に興味深い。

定性推論においては挙動に関する因果的説明の導出も大きな目的となっている。このための代表的な因果解析の手法としては、de Kleerが述べた架空の因果性を用いた手法<sup>[12]</sup>、TQ解析<sup>[52]</sup>、因果関係流解析<sup>[70]</sup>などがある。因果解析の手法に関連した議論として、「内的発展型因果性」、「機能関連的因果性」、「外的駆動型因果性」、「目的論的因果性」の4種類に因果性の分類がなされ<sup>[75]</sup>、動的システムの挙動の解釈における「目的論的因果性」以外の3種の因果性の関係を明らかにし、架空の因果性の導出を行なった研究<sup>[82]</sup>もある。

以上述べたように、定性推論の構造記述ならびに挙動推論、因果解析としてさまざまな手法が提案されている。しかしながら、大規模な動的システムに適用する際には、構造記述をいかにシステムティックに実行するか、定性的挙動が実挙動と異なることを判断し、構造記述をどのように修正するかという2点が、大きな問題となる。本章では、このような問題点に対して、第2章で明らかにした論点も合わせて、動的システムの定量的挙動の因果的説明を導出するという観点からアプローチした定性シミュレーション手法について述べる。

### 3.2 従来の研究と本手法のアプローチ

Process中心の記述<sup>[17]</sup>やConstraints<sup>[30]</sup>に基づく記述では、対象の因果性を十分理解した上で、それぞれの観点から記述する必要があり、構造記述に関してはad hocな点がある。一方、de Kleerが行なったDevice中心の記述<sup>[12]</sup>では、コンポーネント・モデルとデバイス・トポロジーからの自動生成となっている点で、対象が大規模であっても構造記述が可能となろう。最近では、FalkenhainerとForbusが、挙動に関係する記述を与えなくとも、ユーザの質問と対象に関する領域知識を反映した部品モデルをもとに、推論したい状況に必要な構造記述を自動生成する手法を提案し、大規模な系に適用している<sup>[15,16]</sup>。また、構造記述の再構築という観点からのアプローチとして、対象に関する複数のモデルを準備し、最初に設定したモデルの予測挙動が観測挙動と矛盾する際には、それぞれのモ

デル間の違いに対応するモデリングの仮定（摩擦の有無や剛体か変形体かなど）をもとに、自動的にモデルを変更する手法が提案されている<sup>[1]</sup>。溝口の提案している手法<sup>[74]</sup>では、部品間の接続という観点から対象を表現した構造記述言語をもとに定性式の集合を生成するとともに、推論に必要な変数に関する因果解析から定性式を選択し、さらに観測挙動と予測挙動の差によって、その観測挙動を解釈するために必要なモデル修正を実行する。

以上の手法においては、定量的解析とは独立に定性推論を位置づけて、対象に関する部品モデルを中心として全体をシステムティックに構築するという考え方である。したがって、定性モデルは部品モデルの記述、すなわちモデル作成者の知識におおいに依存している。これに対して、本章で述べる手法では、定量的解析で用いられる方程式をもとに定性モデル構築をシステムティックに実行しようとするものである。この背景には、例えばプラントなどの定量的挙動を模擬する定量モデルは、一般に方程式で記述されたものが得られている場合が多く、対象が複雑な場合にこれを利用すれば定性モデルに対するad hocな記述をさけることができると考えるからである。

定性的挙動が実挙動と異なることを判断し、構造記述をどのように修正するかという問題に対して、本手法においては定性シミュレーションの時間として導入する「解釈のための時間」を利用して実行する。

以上のような考え方に基づいて、定性モデルの構築から定性的挙動のシミュレーション方法、さらに現象の因果的説明の導出までの統一的な枠組みを提案する<sup>[3,4,55]</sup>。具体的には、3.3で定性/定量的解析の対応づけの概要を説明する。3.4で「因果順序づけ」を利用した定性モデルの構築について説明し、3.5で定性シミュレーションの時間として導入する「解釈のための時間」の生成、3.6でそれを組み込んだ挙動シミュレーションについて説明する。

### 3.3 定性／定量的解析の対応づけの概要

第2章の図2.2に示された動的システムに対する解析の流れにそって、これらから生成されるデータ（解）やその解析の特徴を比較し、定量的解析と定性的解析の対応づけを考える。

表3.1 定性／定量的解析の特徴

	モデルの表現	導出データ	問題点
定性的解析	定性変数と定性関係	定性的挙動解	曖昧さの発生, 時間の欠落
定量的解析	常微分方程式, 平衡方程式	数値解あるいは 解析解	説明機能の欠落

表3.1が示すように、定性的解析においては、モデルの性質上厳密な数値を用いずに推論を実行し、定性的挙動を導出することができる。この解析で用いられる定性的記述は、人間の直感に適合し、因果的説明の作成につながるという反面、定性的演算における曖昧さから実挙動と異なる定性的挙動が導出される可能性があることや、時間が陽には表現されないといった問題を引き起こす。時間が陽に表現されないことは、定性的挙動と定量データとの対応づけを困難にしている。一方、定量的解析において用いられる方程式は、各変量の間で成立する物理法則やその適用条件に基づいて作成されることから、その表現には物理システムを解析した様々な知識が内包されている。実はこのような知識は、定性的解析のモデル構築の際にも用いられているものでもある。ただし、定性モデルの構築としては、一般に定量モデルに表現された全ての物理変数を定性モデルの変数とすることはない。例えば、発電プラントなどに異常現象が発生した場合には、オペレータは先ず運用上危険とされる事象の発生を防ごうとする。したがって、そのために把握しなければならない因果の流れを示した変数群に注目することが多い。対象とするシステムならびに解決しようとする問題

にもよるが、このように人間が選択した変数群に対して定性モデルの構築がなされることが望ましく、また実際に人間が解釈可能な変数の数はそれほど多くないと考える。

では、問題解決のための変数群が選択された場合、定性的解析においてどの変数間にどのような定性関係の定義を行なう必要があるかを明らかにするにはどうすればよいであろうか。ここでは、常微分方程式や平衡方程式などで表現された物理システムの変数に対する因果順序づけの手法<sup>[25, 26, 27, 28]</sup>を応用する。この因果順序づけの手法は、定量的解析の方程式に表現された変数群の依存構造を抽出するものである。

次に、表3.1の導出データ間の対応づけに関して、次のように行なう。一般に定性シミュレーションから導出された定性的挙動を示す定性データ（これは定性値と定性微分値を用いて表現される）が、定量データの示す挙動と一致しているかどうかは、人間の判断によっている。このことは、人間が定量データを定性的に見直して、それを定性データと比較して判断しているといえる。この考えに基づいて、定量データに対して人間の与えた定性的な見方と定性シミュレーションから導出した定性的挙動データとの対応づけを計算機内部で実行する。

以上の考えをまとめたものが、図3.1である。この図が示すように、定量モデルから得られる変数間の依存構造が、縮約という操作をもとに定性モデル構築の際に利用され、定量データを定性的視点で変換したデータが定性シミュレーションの際に曖昧さの解消に利用される。ただし、後述するように定性モデルの変数群の選択や定性関係の定義およびそれらの修正は、モデル作成者が実行する。

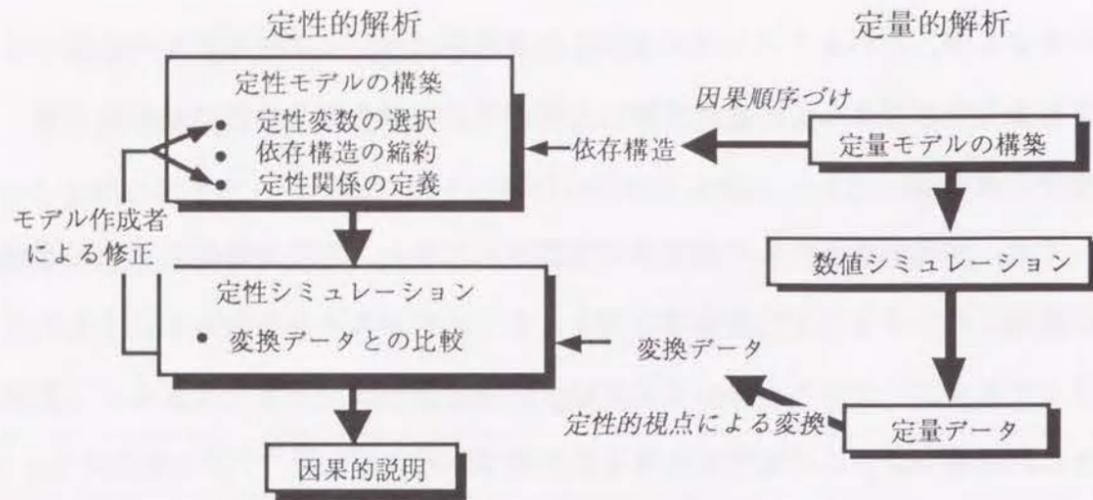


図3.1 定量的解析との対応操作による定性シミュレーション

### 3.4 因果順序づけを利用した定性モデルの構築

#### 3.4.1 定性モデルの依存構造

定量モデルの変数群を $U$ 、定性モデルに組み込む変数群を $Q_{var}$  ( $\subseteq U$ ) とし、 $Q_{var}$ に関する依存構造について説明する。第2章の2.4で述べたように「因果順序づけ」の手法は、定量的解析の方程式に表現された変数群 $U$ の依存構造を抽出するアルゴリズムである。その結果を、例えば $x_i$ が $x_j$ へ直接に影響を及ぼす場合には、依存関係 $\rightarrow$ を用いて $x_i \rightarrow x_j$ と記述する。ここで、 $x_i \rightarrow x_i$ は常に成立すると約束すると、この関係 $\rightarrow$ は反射的かつ推移的であることから擬順序関係となる。ここで、 $Q_{var}$ に対して次の関係 $\prec$ を定義する。

#### 定義3.1 関係 $\prec$

$\forall x_i, \forall x_j \in Q_{var}$  に対して、 $x_i \prec x_j$

$\Rightarrow$

$x_i, x_j \in U$  に対して

(i)  $x_i \rightarrow x_j$

もしくは、

(ii)  $\exists x_k \in U$  かつ  $\exists x_k \notin Q_{var}$ ,  $k=i_1, \dots, i_n$  ( $i_i$ は自然数) に対して

$$x_i \rightarrow x_{i_1} \rightarrow \dots \rightarrow x_{i_n} \rightarrow x_j$$

ここで同様に、 $x_i \prec x_j$ は常に成立すると約束すると、この関係 $\prec$ は反射的かつ推移的であることから擬順序関係となる。すなわち、 $\{U, \rightarrow\}$ ,  $\{Q_{var}, \prec\}$ という変数群とその二項関係においては、ともに擬順序関係が成立している。

この定義に基づく $Q_{var}$ の依存構造の求め方として、 $U$ と依存関係 $\rightarrow$ による有向グラフ $G = (U, E)$ 上で、次のような有向パスから決定する。

#### [ $Q_{var}$ に対する依存構造の求め方]

$\forall x_i \in Q_{var}$  に対して、有向グラフ $G$ 上で変数 $x_i$ からの有向パス $P_{i,k}$ として、 $Q_{var}$ に含まれる変数を終点とするパスを求める ( $k$ は自然数で、 $k$ 番目のパスを示している)。有向パスがあれば、 $Q_{var}$ での依存関係とする。

以上によって、例えば図3.2に示すように、任意の与えられた変数群に対する依存構造を作成することができる。

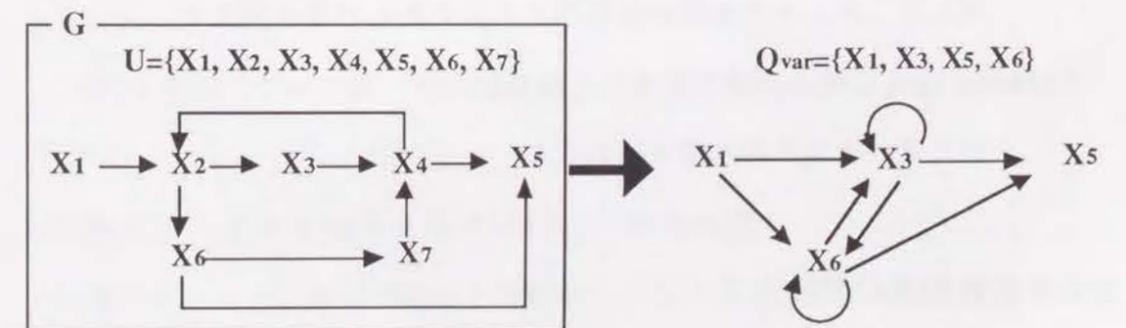


図3.2  $Q_{var}$ における依存構造の例

### 3.4.2 定性モデルにおける定性関係

定性関係に関しては、図3.3に模式的に示した形で2変数間に成立する因果関係を記述する。ここでは、3種類のタイプをもとに記述することとしている。このことにより、例えば「温度がある閾値をこえるとヒータがOn/Off動作をする」関係や「タンクの水位の変化率がタンクからの流出量から決まる」関係などを表現することが可能である。

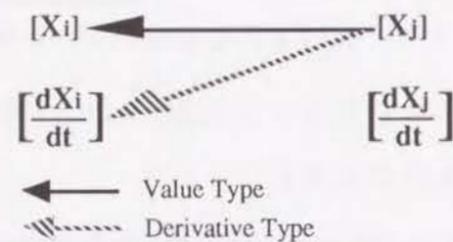


図3.3 定性関係のタイプ

#### [定性関係のタイプ]

##### (1) Value Type (定性値間に定義される関係)

例えば、温度によるヒータのOn/Off動作など

##### (2) Derivative Type (定性微分値と定性値間に定義される関係)

例えば、タンクの水位の変化率とタンクからの流出量など

##### (3) Mixed Type ((1), (2)の混合した関係)

例えば、タンクの水位と圧力など

### 3.4.3 定性関係の割り当て

3.4.1で求めた依存構造の2変数間に対して、3.4.2で述べた定性関係を割り当てる。この際の定性関係の決定方法として、例えば対象システムのいくつかの動作点近傍での挙動を定量モデルから求め、それらのデータから定性関係を決

定することや、あるいは専門家の保有する知識を利用するなどが考えられる。前者は、対象システムの性質や動作点の選び方に大きく左右され、後者は専門家からの知識獲得が困難な場合に問題がある。

この二つの考えは、選択された定性変数群に対する定性モデルをもとに挙動予測を行なった結果が実挙動と一致するようなモデルを、一回の試行で得ることを前提としている。ここでは、図3.1に示すようにモデル構築に修正のプロセスを導入する。例えば、定性シミュレーションの結果が実挙動と一致しない変数に関しては、変数間の依存構造に基づいて関係する変数との定性関係をモデル作成者が見直すことによって、定性モデルの修正を実行する。

図3.4に、このようにして依存構造と定性関係の割り当てによって構築される定性モデルの例を示す。

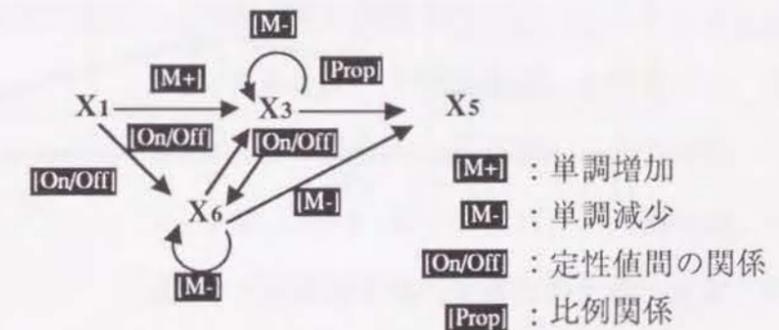


図3.4 定性モデルの例 (依存構造ならびに定性関係を構成要素とする)

### 3.5 定量データから導入される解釈のための時間

定性シミュレーションの結果と実挙動を比較するときに、人間は定量データを定性的にとらえなおしている。このような定量データの変換をもとにして、定性シミュレーションの時間として導入する「解釈のための時間」について述べる。

### 3.5.1 定量データの変換

人間が、定量データを定性的に捉えなおす際に用いる定量データの特徴点として、ここでは次のものを定義する。

#### 定義3.2 定量データの特徴点

- (i) 増減の変化する時点
- (ii) 定常動作時の値をとる時点
- (iii) 閾値をとる時点

各々の変数に関して、その定量データをトレンド・グラフとして表示し、定義3.2に基づく特徴点を用いて定量データに対する分節化を実行すると、図3.5に示すようにある時区間は一定の傾向を示すデータが得られる。以下、これを定性的制約データ  $Q_{di}$  ( $i$ は自然数) と呼ぶ。 $Q_{di}$ はその区間の両端の時間と定量値を属性として持つことで、定性的な増減状態を表現している。すなわち、

$V_s < V_e \Rightarrow [x'] = \text{増加}$  (別表現では,  $[+]$ とも書く)

$V_s = V_e \Rightarrow [x'] = \text{安定}$  (別表現では,  $[0]$ とも書く)

$V_s > V_e \Rightarrow [x'] = \text{減少}$  (別表現では,  $[-]$ とも書く)

となる。なお、定量データの両端は、必ず特徴点とする。

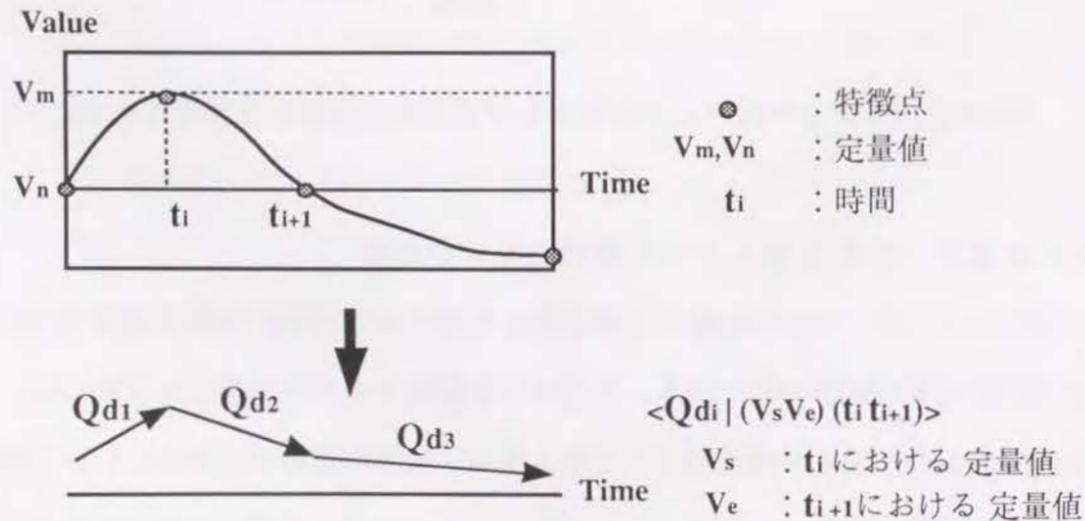


図3.5 定量データの変換

### 3.5.2 解釈のための時間の生成

一般に、定性推論では挙動は瞬時的状態と持続的な状態の繰り返しとして現われ、時間はこの瞬時的状態によって定義される。これは定性的状態を区別するための時間であり、数値シミュレーションで用いられる時間には対応していない。前述の定性的制約データとの対応をとるには、共通の時間軸上での関連づけが必要である。

ここでは定性状態をとらえるための時間を、前節で述べた定性的制約データを用いて次のように作成する。例えば、図3.6が示すように2変数の定性的制約データの持つそれぞれの時間を、同じ時間軸上にまとめると、新たな時間記述を生成することができる。

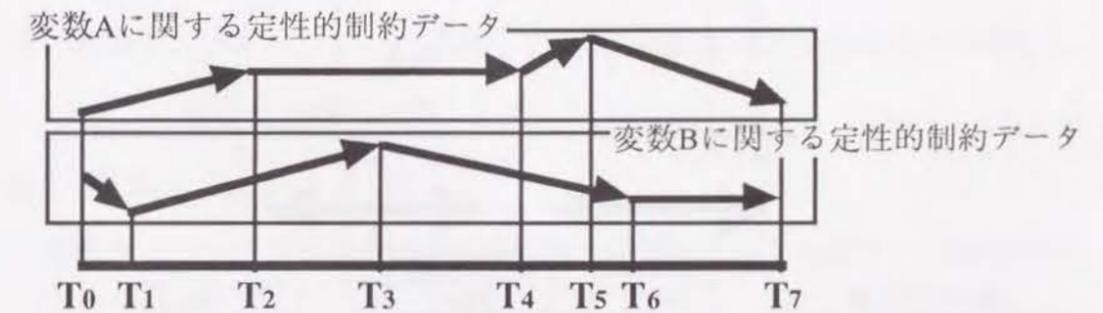


図3.6 「解釈のための時間」の生成

ただし、このような合併操作においては、時間にわずかに差がある定性的制約データどうしをそのまま処理すると、生成される時間記述は非常に細かくなる。このような細かい時間記述をもとに定量データを解釈しなければならない現象もあるが、一般にはわずかな差は同一視して解釈するほうがわれわれにとっては自然と考えられる。また、この操作によって、後述する定性シミュレーションに基づき、人間にとって自らが与えた定性的視点に直感的に対応する因果的説明を作成することができる。ただし、現象の解釈が定性的制約データに基づいていることから、わずかな差を同一視した結果がもとの定性的制約データの

集合の持つ性質を変更してはならない。このような点を考慮した合併操作を、以下に説明する。

[時間に関する選択的合併操作]

図3.7に示すように、定性的制約データに表現された全ての絶対時間 $t_i$ を同じ絶対時間軸上で並べたものを $M_{t_i}$  ( $i$ は自然数)とすると、次のようなグループ分け操作により選択的に合併する。

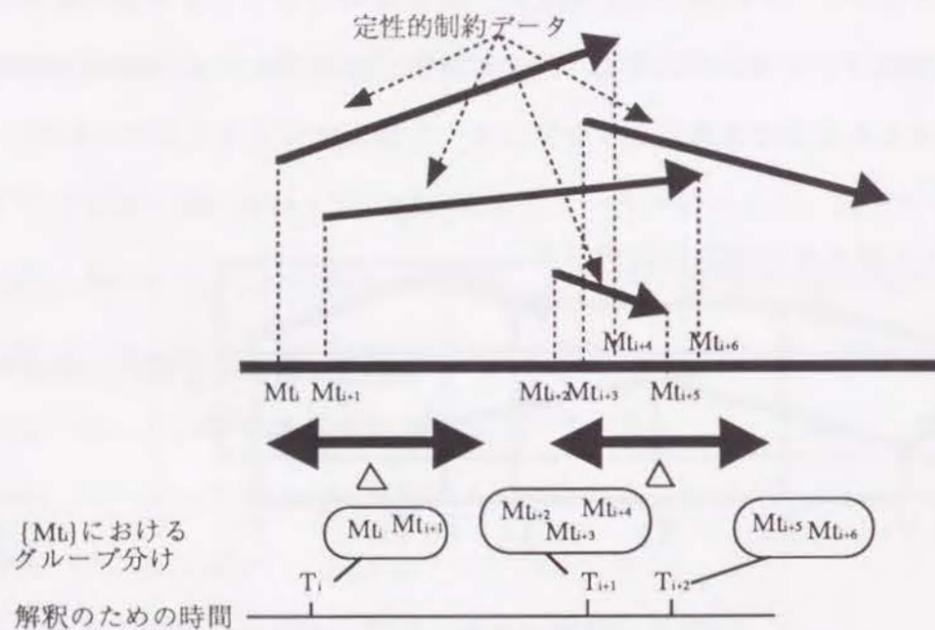


図3.7 選択的合併操作

- (手順1) 定量データの観測終端までの時間の $1/D$  ( $D$ : 整数)を基準値 $\Delta$ として,  
 $M_{t_i}, M_{t_{i+\alpha}}$  ( $\alpha$ は自然数)の差に関して $M_{t_{i+\alpha}} - M_{t_i} \leq \Delta$ のときに合併可能かどうかを調べる。合併可能な場合は、これらにいったん同じグループ属性を与えて、さらに $M_{t_{i+\alpha+1}}, M_{t_{i+\alpha+2}}, \dots$ の合併可能性を調べる。合併可能性は、 $M_{t_i}, M_{t_{i+\alpha}}$ がある定性的制約データの両端の時間になっていない場合である。
- (手順2) 時間 $M_{t_i}$ は(1)の操作によって、グループ分けされる。それぞれのグループの平均値をそのグループの代表値としたリストの要素に番号を割り当て

て $T_i$  ( $i \geq 0$ )とし、「解釈のための時間」とする。この時間記述を用いて定性的制約データに時区間を割り当てる。

この操作を定性モデルに含まれる全変数について実行後作成された時間記述の任意の時区間 ( $T_i, T_{i+1}$ ) においては、対象システムの定量データを一つの定性状態に対応づけることができる。したがって、この時区間ごとに定性シミュレーションの生成する挙動と比較することが可能となる。作成した $T_i$ を「解釈のための時間」と呼ぶ。

3.6 解釈のための時間を組み込んだ挙動シミュレーション

3.4にて作成した定性モデルと3.5にて導入した「解釈のための時間」をもとに、挙動シミュレーションを実行して因果的説明を導出する処理を説明する。

3.6.1 挙動シミュレーション

挙動シミュレーションでは、定性変数に初期値を与えた後に、定性変数間に発生する影響を伝播し、その伝播された影響に基づいて定性変数の状態遷移を実行する。この際に「解釈のための時間」の進行は、次のようになる。

$$T_0 \rightarrow (T_0, T_1) \rightarrow T_1 \rightarrow (T_1, T_2) \dots$$

ただし、 $T_i$ :  $i$ 番目の解釈のための時間

以後、時点から時区間への進行をPI遷移、その逆をIP遷移と呼ぶ。それぞれの前提条件ならびに処理は次のとおりである。

[PI遷移]

定義: 時点 $T_i$ から時区間 ( $T_i, T_{i+1}$ ) への状態遷移

前提条件:  $T_i$ においては、少なくとも一つの変数の定性的状態が定量データにつけられた特徴点に一致している



表3.2 PI遷移における状態遷移表 ( $l_j, l_{j+1}$ は定性値)

現在の定性状態	遷移可能な定性状態
$\langle l_j, \text{steady} \rangle$	$\langle l_j, \text{steady} \rangle$
$\langle l_j, \text{increase} \rangle$	$\langle l_j l_{j+1}, \text{increase} \rangle$
$\langle l_j l_{j+1}, \text{increase} \rangle$	$\langle l_j l_{j+1}, \text{increase} \rangle$
$\langle l_j, \text{decrease} \rangle$	$\langle l_{j-1} l_j, \text{decrease} \rangle$
$\langle l_j l_{j+1}, \text{decrease} \rangle$	$\langle l_j l_{j+1}, \text{decrease} \rangle$
$\langle l_j l_{j+1}, \text{steady} \rangle$	$\langle l_j l_{j+1}, \text{steady} \rangle$

表3.2においては左側の定性状態にある変数の状態は、右側の定性状態に遷移することを表している。また、Value Typeのような状態遷移を組み込んだのは不連続な変化を表現するためであり、さらにMixed TypeはValue TypeおよびDerivative Typeの拡張となっている。

### [IP遷移]

定義：時区間  $(T_i, T_{i+1})$  から時点  $T_{i+1}$  への状態遷移

前提条件：全ての変数の定性状態が  $(T_i, T_{i+1})$  における定性的制約データに一致する

状態遷移処理：

定義と前提条件から、全ての変数の時点  $T_{i+1}$  への状態遷移の結果が満足すべき状態として、定性微分値についてはあらかじめ与えられていない。従って、前後の時区間の定性的制約データから求める必要がある。ここでの処理について、図3.9における  $(T_{i-1}, T_i)$  から  $T_i$  へのIP遷移をもとに説明する。

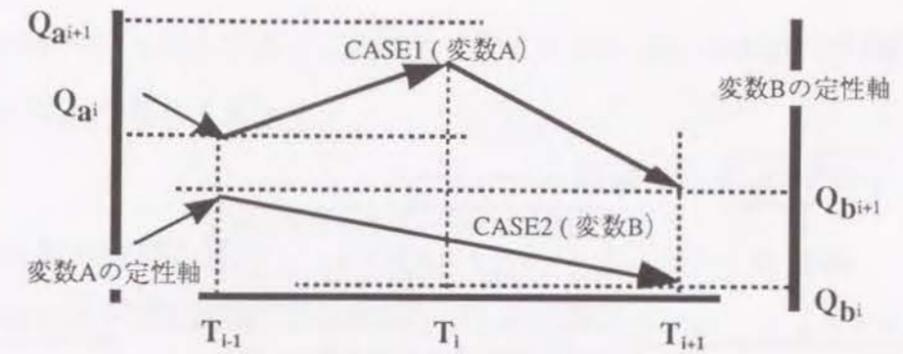


図3.9 IP遷移の処理における場合分け

シミュレーションが進行すべき時間  $T_i$  と定性的制約データとの間には二つの場合があり、このことから変数の定性軸上での定性微分値のとりえ方に違いが生じる。図3.9で、CASE2では、 $(T_{i-1}, T_{i+1})$  での定性微分値と同じになるが、CASE1のように、定性的制約データが  $(T_{i-1}, T_i)$  で「増加」、 $(T_i, T_{i+1})$  で「減少」の場合に、定性微分値は「増加」から「安定」を経て「減少」に変わる。この遷移をシミュレーションする際に、本来IP遷移  $(T_{i-1} \rightarrow T_i)$  で「増加」から「安定」、PI遷移  $(T_i \rightarrow T_{i+1})$  で「安定」から「減少」ととらえるべきところを、IP遷移では「増加」から「増加」、PI遷移で「増加」から「安定」を経て「減少」ととらえた処理を行なっている。このようにとらえても「なぜ増加から減少に変わっていったか」に関してシミュレーションから得られる因果的説明は同じであり、しかも  $T_i$  での満足すべき定性微分値を二つの定性的制約データから求める必要はない。また、この場合のIP遷移においては定性微分値の変化がなく、定性値も特徴点からの遷移でないことから、現象を解釈するうえでの因果的説明にも特徴的なものが現われない。この時にそれぞれの変数は相互の影響を含みながらも自身の定性微分値に従って遷移するとみなせる。IP遷移ではこの考えに基づいて、求めた定性値と  $(T_{i-1}, T_i)$  の定性微分値に定性変数の状態を書き換えるようにする。

以上(1)と(2)の処理の流れをまとめたものが、図3.10である。なお、定性状態が書き換えられた変数には、その時区間での状態が同定されたとしてJustifiedと

いうマークを付ける。

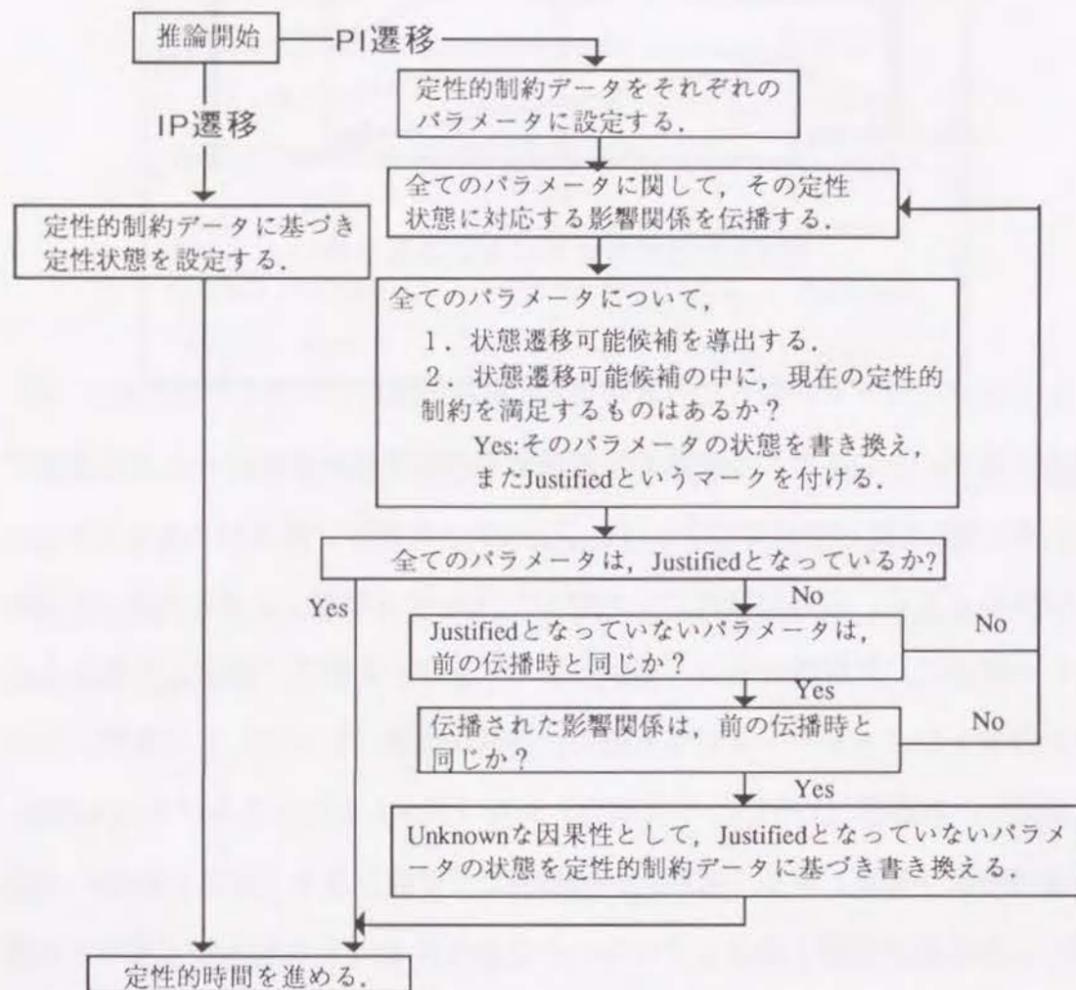


図3.10 状態遷移処理の流れ

ただし、PI遷移において影響を伝播しても、定性変数の状態がその時区間の定性的制約データを満足しない可能性がある。このことは、定性モデルとしての構造記述に不備があると考えられ、シミュレーションを停止し、定性モデルに修正を行なう必要がある。しかしながら、PI遷移において時区間ごとに与えた定性的制約データへの遷移をシミュレーションする本手法では、図3.10に示すようにJustifiedでない定性変数の状態を定性的制約データを用いて書き換えることによって、隣接する時区間とは独立してシミュレーションが実行可能とな

る。この際に、Justifiedでない定性変数に関する因果性を未確認として記録し、まとめて処理するようにする。

### 3.6.2 因果的説明の導出

人間が定量データに対する因果的説明を行なうときには、それぞれの変数が示す定量データの特徴（例えば、増加しているとかある値になったとか）をもとに、それらの「決定-被決定関係」を時間軸にそって明らかにしようとする。

3.5.1で述べた定量データに対して与えた定性的視点から定性的制約データを生成することは、上述の定量データの特徴を決めることに対応する。3.6.1で述べた挙動シミュレーションは、それら定性的制約データの「決定-被決定関係」を、定性モデルを用いて明らかにする過程である。従って、影響の伝播を時区間ごとに記録することによって、因果的説明を作成することができる。この結果得られる因果的説明は、本手法においては図3.11に示すような影響伝播のシーケンスとしてとらえられる。

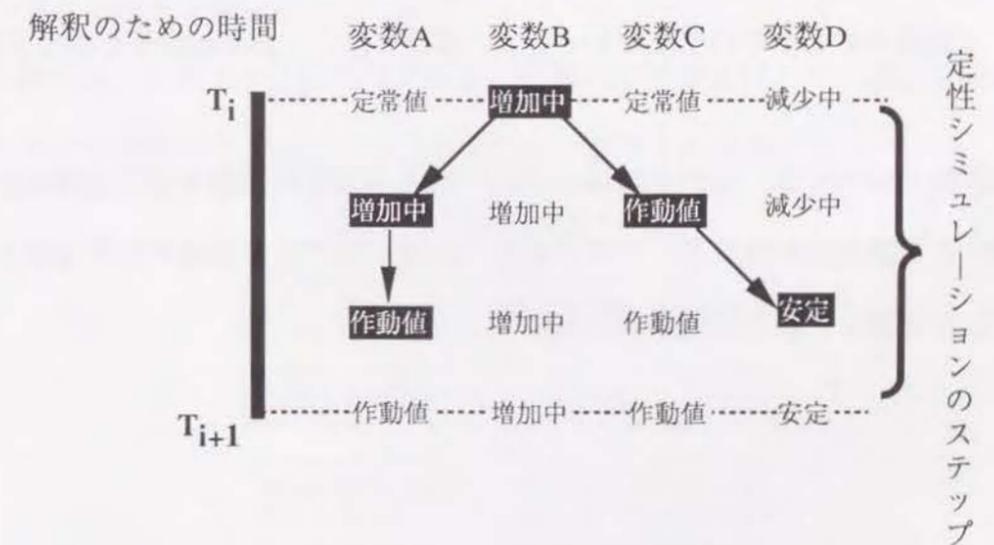


図3.11 因果的説明を示すシーケンス

### 3.7 結言

本章では、大規模な動的システムに対する定量的解析の実行が可能な場合において、定性的解析を実行する意義として、現象を因果的に説明することに焦点をあて、計算機上での具体的な実現方法を論じた。すなわち、定性モデルの構築、定性シミュレーション手法、因果的説明の導出を定量的解析と関連づけて述べた。その要点を、以下にまとめる。

- (1) 定性モデルの構築に際しては、定量モデルから導かれた依存構造を利用することにより、モデル作成者の視点を反映する変数群に関して定性関係を定義すべき変数の対が求まる。
- (2) 定量データに対する定性的視点をもとに導いた定性的制約データと「解釈のための時間」を用いて、影響の伝播による定性シミュレーションに発生する曖昧さの解消を行なう。
- (3) 定量データとの対応づけを組み込んだ定性シミュレーションを、影響の伝播に基づいて実行することによって、現象に内在する変数間の相互の影響を明らかにし、そのシミュレーション過程に基づく因果的説明を導出する。

本手法においては、定性変数群の選択に対する規範や適確な定性関係の定義に対する支援方法が課題として残されているが、専門家の知識の活用などをもとにした有効な手法の開発が考えられる。

## 4 発電プラントの定性シミュレーション

### 4.1 緒言

第3章では、大規模な動的システムに関して定量的な関係を記述した方程式とそれに基づく定量的挙動が与えられたときに、定性推論を適用して因果的説明を導出する手法を述べた。本章では、そこで提案した手法により発電プラントに対する定性シミュレーションを実行する。

発電プラントにおいては、発電機に接続した電力系統の負荷が急激に変化する際に、制御系が正確に作動し負荷に追従するかが重要な問題である。このため運用時には計測装置からえられた定量データに対して、第1章でも述べたように運用員が因果的解釈を実行し、プラントが定常状態に移行しつつあるかを監視し、異常を判断しようとする。しかしながら、プラントでは時定数の大きく異なる機器内のプロセス変数が複雑に相互作用しているために、詳細な因果関係から全てを解釈するよりは、特定のプロセス変数群に着目した因果的説明がなされる場合が多い。

本章では、このような発電プラントの負荷遮断現象に対して、図4.1に示す流れにそって定性シミュレーションを実行した結果を述べる。

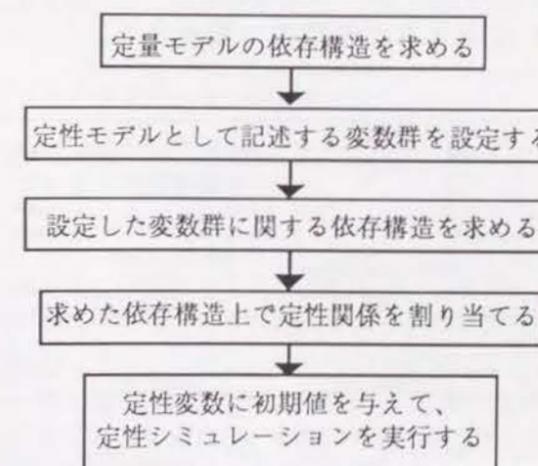


図4.1 因果的説明導出の流れ

#### 4.2 発電プラントの定量モデル

図4.2に示すような発電プラントの定量モデルは、方程式の数が75（微分方程式：38，平衡方程式：37），変数の数が113となっている。方程式の例を図4.2中に示す。

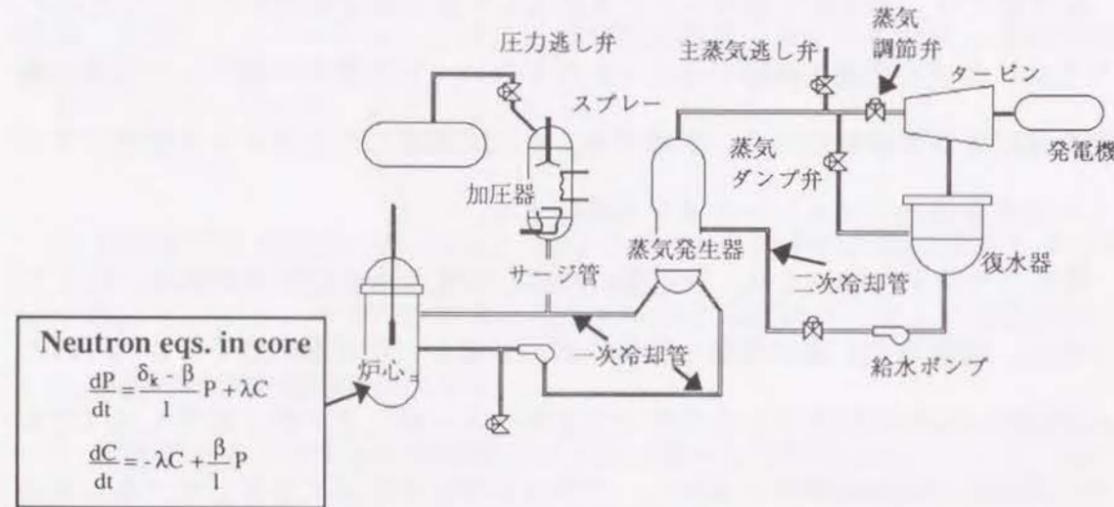


図4.2 発電プラントの系統図

対象とした発電プラントにおいて電力系統の変動から発電機の負荷が急激に減少したときには、タービンに流入する蒸気量を弁の急閉によって制限する。このことを原因として、図4.3に示すような定量データがえられ、「冷却水の温度の上昇」や「バイパス系の蒸気ダンプ弁の開閉」などの現象が発生する。なお、表4.1は、方程式に表現された変数の一覧である。

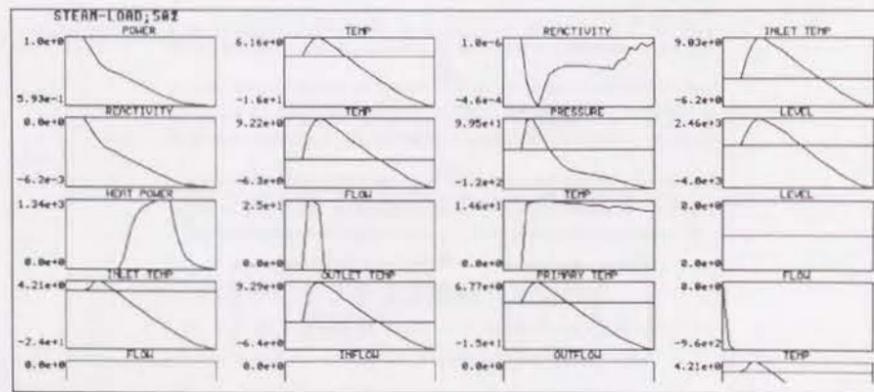


図4.3 負荷変化に対する定量データの例

表4.1 発電プラントのプロセス変数

機器名	プロセス変数名
蒸気発生器 (12)	SG内水温度偏差, SG内入口水温度偏差, SG内出口水温度偏差, 圧力流量偏差, 飽和蒸気流量, 飽和蒸気容量, 飽和水容量to蒸気, 飽和水容量, SG内圧力トリップマージン, 冷却水3温度偏差, 冷却水4温度偏差, 二次冷却水質量偏差
加圧器 (11)	スプレー質量, フラッシング質量, 加圧器内冷却水エンタルピー偏差, バックアップヒータ出力, バックアップヒータ出力偏差, 加圧器内冷却水質量偏差, 加圧器内冷却水内部エネルギー偏差, 加圧器圧力偏差, 加圧器内蒸気質量偏差, 加圧器内蒸気容量偏差, 加圧器内冷却水容量偏差
炉心 (20)	炉心内冷却水温度偏差, 炉心内入口冷却水温度偏差, 炉心内出口冷却水温度偏差, 先行中性子濃度, 総炉心出力偏差, 炉心出力, 炉心出力偏差, 即時炉心出力偏差, 遅発炉心出力偏差, 総反応度, 制御棒外乱反応度, 炉心内圧力トリップマージン, 実効燃料温度偏差, 燃料1温度偏差, 燃料2温度偏差, 燃料2温度偏差測定値, 冷却水1温度偏差, 冷却水2温度偏差, 制御棒速度, スクラム反応度
蒸気調節弁 (1)	負荷流量偏差
蒸気ダンプ弁 (2)	SG蒸気ダンプ弁流量, 蒸気ダンプ弁流量偏差
主蒸気逃し弁 (1)	安全弁流量偏差
圧力逃し弁 (1)	逃し弁による質量
一次冷却管 (2)	容器入口温度偏差, 容器出口温度偏差,
二次冷却管 (3)	二次側流量偏差, 二次冷却水温度, 二次冷却水温度偏差
サージ管 (2)	サージエンタルピー, サージ質量
給水ポンプ (1)	フィードウォーター流量偏差
制御系など (19)	参照温度信号, 比例ダンプ温度誤差信号, フィルター後の比例ダンプ温度誤差信号, 制御棒位置による誤差温度信号, 制御棒速度による誤差温度信号, 二次側圧力, 誤差温度信号, 炉心内冷却水温度偏差測定値, 二次側流量偏差測定値, 加圧器圧力偏差補償値, $\delta$ , 二次側内部エネルギー偏差, 平均温度偏差, 平均温度偏差補償値, 炉心出力偏差補償値, 制御棒反応度, ステップ外乱SG内水温度偏差補償値, 平均温度偏差遅れ値, ステップ外乱SG内水温度偏差

### 4.3 発電プラントの依存構造

4.2で述べた定量モデルに対して、まず因果順序づけにより依存構造を求めると図4.4に示すような結果がえられ、これをグラフ表示したものが図4.5である。

#### 0次

制御棒外乱反応度，負荷流量偏差，炉心出力，先行中性子濃度，遅発炉心出力偏差，炉心内冷却水温度偏差測定値，燃料2温度偏差測定値，燃料1温度偏差，燃料2温度偏差，炉心内冷却水温度偏差，SG内水温度偏差，冷却水1温度偏差，冷却水2温度偏差，冷却水3温度偏差，冷却水4温度偏差，SG内入口水温度偏差，容器入口温度偏差，炉心内入口冷却水温度偏差，容器出口温度偏差，二次冷却水質量偏差，二次側内部エネルギー偏差，二次冷却水温度偏差，フィルター後の比例ダンブ温度誤差信号，二次側流量偏差測定値，蒸気ダンブ弁流量偏差，加圧器内蒸気質量偏差，加圧器内冷却水質量偏差，加圧器内冷却水内部エネルギー偏差，加圧器圧力偏差，加圧器内蒸気容量偏差，加圧器内冷却水容量偏差，バックアップヒータ出力偏差，平均温度偏差，平均温度偏差補償値，加圧器圧力偏差補償値，炉心出力偏差補償値，制御棒反応度，ステップ外乱SG内水温度偏差補償値，平均温度偏差遅れ値，ステップ外乱SG内水温度偏差，

#### 1次

炉心出力偏差，実効燃料温度偏差，炉心内出口冷却水温度偏差，SG内出口水温度偏差，二次冷却水温度，安全弁流量偏差，比例ダンブ温度誤差信号，参照温度信号，加圧器内冷却水エンタルピー偏差，スプレー質量，逃し弁による質量，バックアップヒータ出力，SG内圧力トリップマージン

#### 2次

即時炉心出力偏差，飽和水容量，二次側圧力，誤差温度信号，フラッシング質量，制御棒位置による誤差温度信号，制御棒速度による誤差温度信号，炉心内圧力トリップマージン

#### 3次

総炉心出力偏差，飽和水容量to蒸気，SG蒸気ダンブ弁流量， $\delta$

#### 4次

飽和蒸気容量，制御棒速度，サージエンタルピー&サージ質量

#### 5次

飽和蒸気流量

#### 6次

圧力流量偏差

#### 7次

二次側流量偏差

#### 8次

フィードウォーター流量偏差，スクラム反応度

#### 9次

総反応度

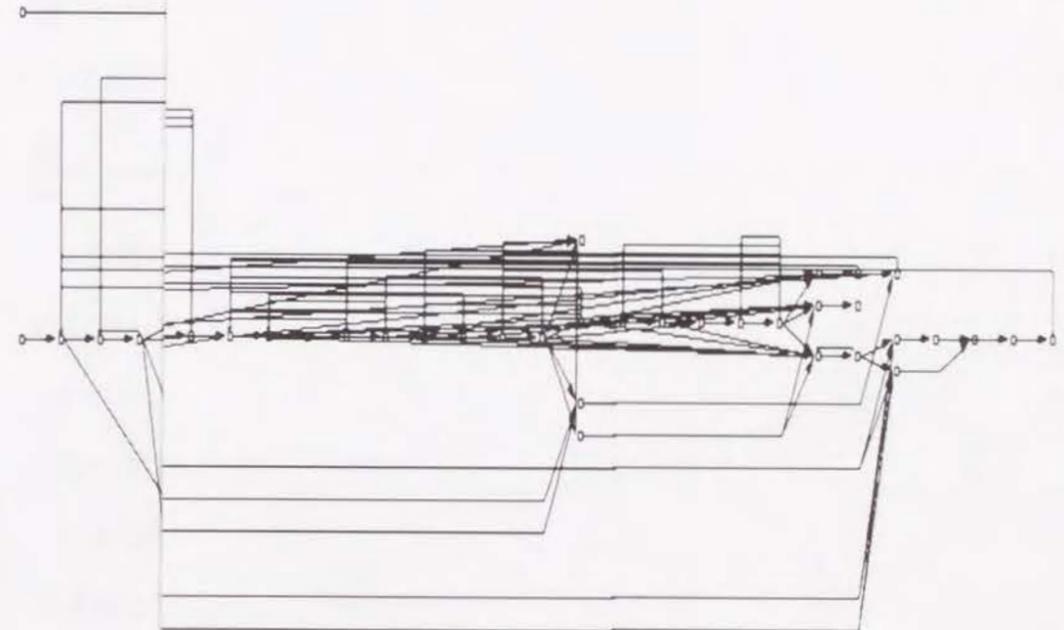
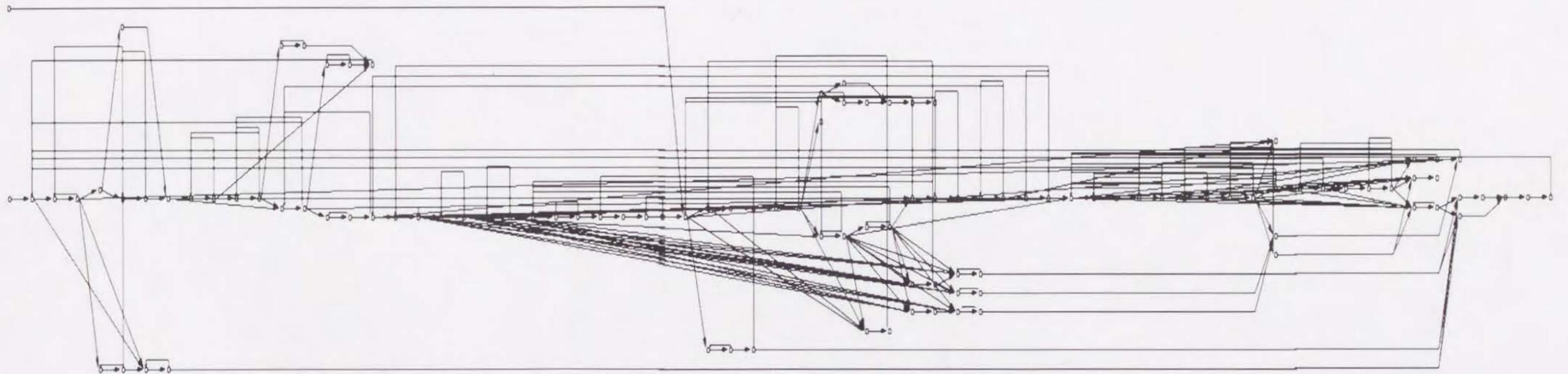


図4.4 因果順序づけの結果



□ 変数  
 → 依存リンク

図4.5 発電プラントの依存構造

図4.5に示すように、113の変数で表現される定量モデルの依存構造は複雑である。因果順序づけの実行について、以下の22個の方程式に関する部分をもとに詳しく説明する。なお、この22個の部分は発電プラントを構成する主要な部分の機器に関連している。

#### 炉心における出力など

$$\begin{aligned} \frac{d}{dt} \text{炉心出力} \\ = (\text{総反応度} - \text{定数C}) / \text{定数C} \cdot \text{炉心出力} + \text{定数C} \cdot \text{先行中性子濃度} \end{aligned} \quad (4.1)$$

$$\begin{aligned} \frac{d}{dt} \text{先行中性子濃度} \\ = \text{定数C} \cdot \text{炉心出力} - \text{定数C} \cdot \text{先行中性子濃度} \end{aligned} \quad (4.2)$$

$$\begin{aligned} \text{炉心出力偏差} \\ = \text{炉心出力} - \text{定数初期炉心出力} \end{aligned} \quad (4.3)$$

$$\begin{aligned} \text{総炉心出力偏差} \\ = \text{即時炉心出力偏差} + \text{遅発炉心出力偏差} \end{aligned} \quad (4.4)$$

$$\begin{aligned} \text{即時炉心出力偏差} \\ = \text{炉心出力偏差} \cdot (1 - \text{定数C}) \end{aligned} \quad (4.5)$$

$$\begin{aligned} \frac{d}{dt} \text{遅発炉心出力偏差} \\ = (\text{定数C} \cdot \text{炉心出力偏差} - \text{遅発炉心出力偏差}) / \text{定数C} \end{aligned} \quad (4.6)$$

#### 冷却水の温度など

$$\begin{aligned} \frac{d}{dt} \text{燃料1温度偏差} \\ = (\text{定数C} \cdot \text{総炉心出力偏差} - \text{定数C} \cdot (\text{燃料1温度偏差} - \text{燃料2温度偏差})) / \\ \text{定数C} \end{aligned} \quad (4.7)$$

$\frac{d}{dt}$ 燃料2温度偏差

$$= (定数C \cdot 総炉心出力偏差 + 定数C \cdot (燃料1温度偏差 - 燃料2温度偏差) - 定数C \cdot (燃料2温度偏差 - 炉心内冷却水温度偏差)) / 定数C \quad (4.8)$$

$\frac{d}{dt}$ 炉心内冷却水温度偏差

$$= (定数C \cdot 総炉心出力偏差 + 定数C \cdot (燃料2温度偏差 - 炉心内冷却水温度偏差) - 定数C \cdot (炉心内出口冷却水温度偏差 - 炉心内入口冷却水温度偏差)) / 定数C \quad (4.9)$$

蒸気発生器内の蒸気量など

$\frac{d}{dt}$ 二次冷却水質量偏差

$$= \text{フィードウォーター流量偏差} - \text{二次側流量偏差} \quad (4.10)$$

二次側流量偏差

$$= \text{負荷流量偏差} + \text{圧力流量偏差} + \text{安全弁流量偏差} + \text{蒸気ダンプ弁流量偏差} \quad (4.11)$$

負荷流量偏差

$$= \text{外生変数} \quad (4.12)$$

圧力流量偏差

$$= f(\text{飽和蒸気流量}) \quad (4.13)$$

安全弁流量偏差

$$= f(\text{二次冷却水温度偏差}) \quad (4.14)$$

比例ダンプ温度誤差信号

$$= \text{平均温度偏差} + 定数C \cdot ((\text{容器出口温度偏差} - \text{SG内入口水温度偏差}) / 定数C + 定数C \cdot (\text{SG内水温度偏差} - \text{冷却水4温度偏差}) / 定数C - (\text{SG内入口水温度偏差} - \text{冷却水3温度偏差}) / 定数C) \quad (4.15)$$

誤差温度信号

$$= \text{フィルター後の比例ダンプ温度誤差信号} - \text{参照温度信号} \quad (4.16)$$

参照温度信号

$$= 定数C \cdot \text{二次側流量偏差測定値} \quad (4.17)$$

SG蒸気ダンプ弁流量

$$= f(\text{誤差温度信号}) \quad (4.18)$$

加圧器の圧力など

$\frac{d}{dt}$ 加圧器圧力偏差

$$= 定数C \cdot (-(定数C \cdot (定数C \cdot (\text{フラッシング質量} - 定数C \cdot \text{スプレー質量} - \text{逃し弁による質量})) + ((定数C \cdot \text{スプレー質量} + \text{バックアップヒータ出力偏差} - 定数C \cdot \text{フラッシング質量} + \text{サージエンタルピー} \cdot \text{サージ質量}) / 定数C)) + \text{加圧器内蒸気質量偏差} \cdot ((定数C \cdot \text{スプレー質量} + \text{バックアップヒータ出力偏差} - 定数C \cdot \text{フラッシング質量} + \text{サージエンタルピー} \cdot \text{サージ質量}) / 定数C) - \text{加圧器内冷却水容量偏差} \cdot (\text{フラッシング質量} - 定数C \cdot \text{スプレー質量} - \text{逃し弁による質量})) / (定数C + \text{加圧器内蒸気質量偏差})^2) \quad (4.19)$$

スプレー質量

$$= f(\text{加圧器圧力偏差}) \quad (4.20)$$

逃し弁による質量

$$= f(\text{加圧器圧力偏差}) \quad (4.21)$$

バックアップヒータ出力

$$= f(\text{加圧器圧力偏差}, \text{加圧器内冷却水容量偏差}) \quad (4.22)$$

図4.6に上記の(4.1)から(4.22)に関する部分を示す。白ヌキで示した方程式は、微分方程式（8個）であり、これらによって図に示した変数群 {炉心出力, 先行中性子濃度, 遅発炉心出力偏差, 燃料1温度偏差, 燃料2温度偏差, 炉心内冷却水温度偏差, 二次冷却水質量偏差, 加圧器圧力偏差} に関して、それぞれ「変数名=定数」とする式が元の平衡方程式群に新たに加わる。以下、微分方程式に対応する定数式を(4.1)ならば(4.1)'のように記述する。蒸気調節弁による「負荷流量偏差」は、この場合には外生変数として与えられ、因果順序づけの中での取り扱いは定数式と同じとなる。定数式はそれ自身極小完結集合であり、したがって図4.6に示すように(4.1)', (4.2)', (4.6)', (4.7)', (4.8)', (4.9)', (4.10)', (4.12), (4.19)'のそれぞれの式が0次の極小完全集合の要素となる。次に、例えば炉心出力を右辺に持つ(4.3)式の炉心出力偏差は決定されることから、これは1次の極小完全集合の要素となる。このようにして図に矢印で示したような流れで、炉心や蒸気流量に関する方程式に含まれる変数の間に順序づけがなされる。なお、ここであげた以外の方程式に基づく順序づけについての矢印は図中に示してなく、どの段階（k次）の極小完全集合の要素であるかを示すにとどめる。なお、付録Aにこの依存構造の解析結果のデータファイルを示す。

また、図4.7は図4.5の一部を示したもので、矢印を付けたアークが変数間の依存関係を示している。ただし、図4.7においては図中に示された変数以外の変数からの依存リンクは省略し、逆にそれら残りの変数への依存リンクは破線で示している。

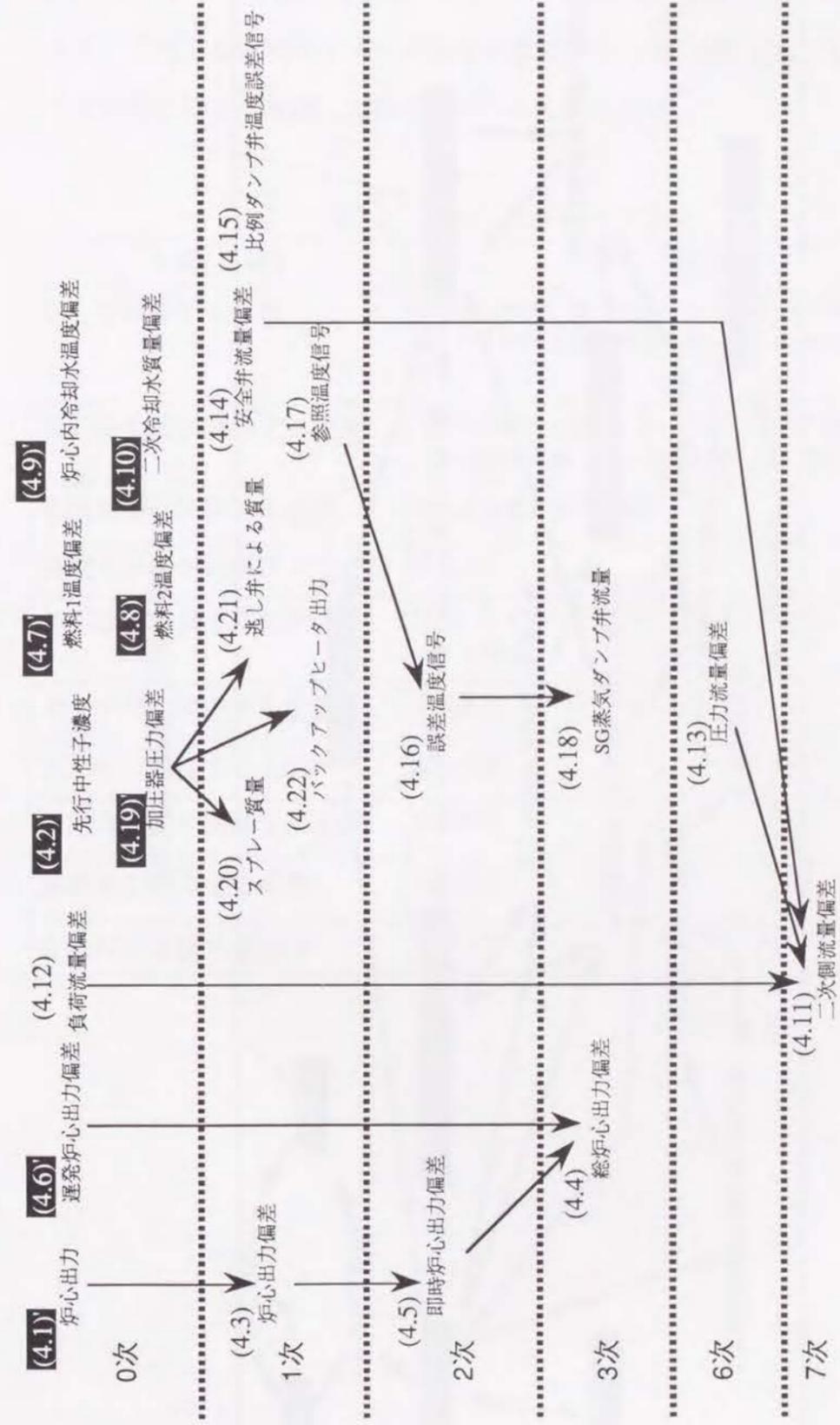


図4.6 因果順序づけの例

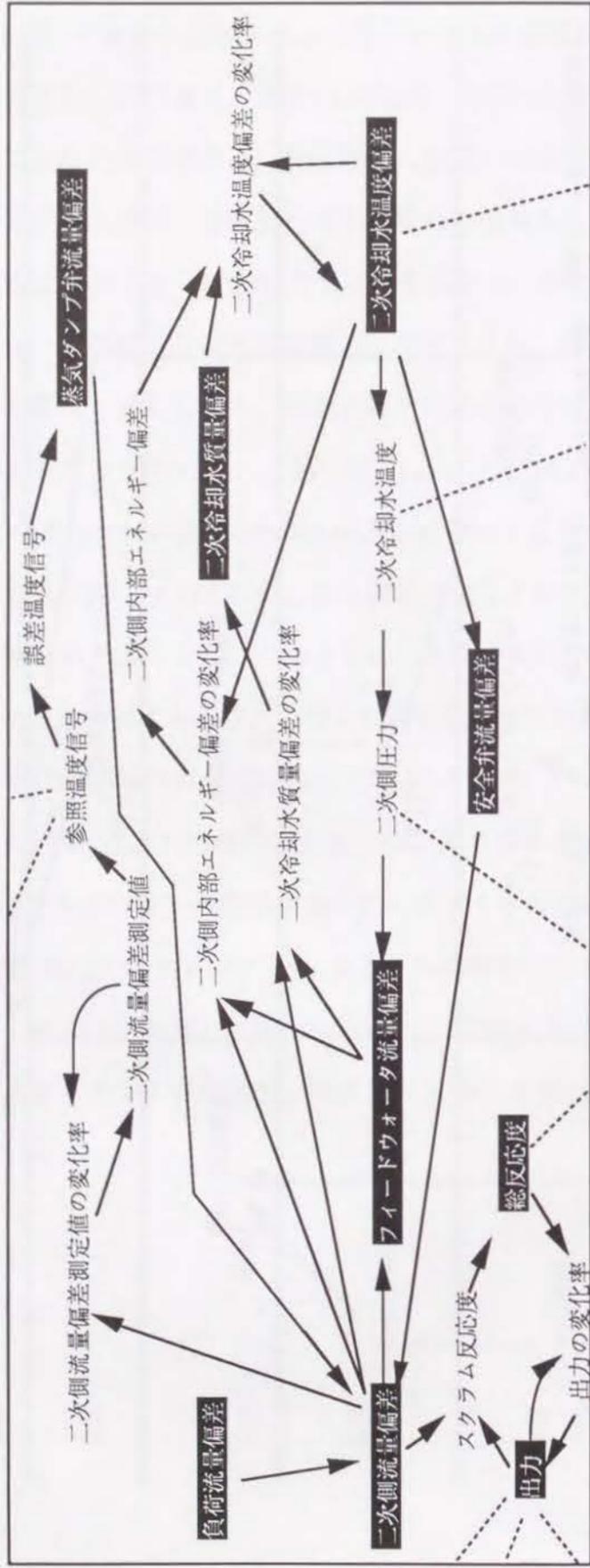


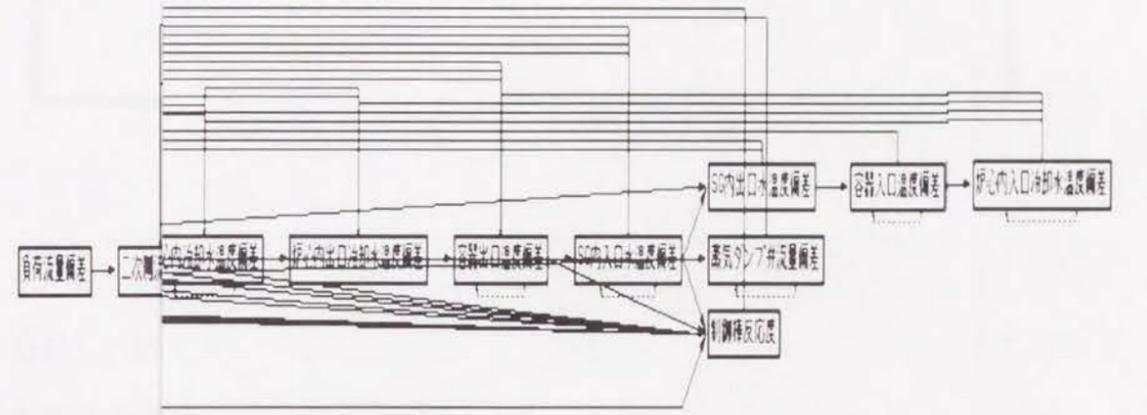
図4.7 依存構造 (定量モデル)

#### 4.4 発電プラントの縮約された依存構造

表4.2に示すような変数25個は、プラントの運転員が通常注視する重要なものであり、これらを定性モデルに表現する変数群とした。図4.8は、この25個について元の依存構造を縮約した結果を示したものである。

表4.2 選択したプロセス変数

関連する機器	変数名
炉心に関する変数	制御棒反応度, 総反応度, 出力, 炉心内冷却水平均温度, 炉心内流入側冷却水温度, 炉心内流出側冷却水温度
蒸気発生器に関する変数	一次冷却水平均温度, 流入側一次冷却水温度, 流出側一次冷却水温度, 2次冷却水量, 蒸気発生量
格納容器に関する変数	流入側温度, 流出側温度
循環水ポンプに関する変数	流量
加圧器に関する変数	圧力, スプレー流量, サージ流量, 蒸気量, 冷却水量, ヒータ熱量
ダンブ弁に関する変数	蒸気流量
逃し弁に関する変数	蒸気流量
圧力逃し弁に関する変数	蒸気流量
調節弁に関する変数	蒸気流量
2次系に関する変数	平均温度





例えば，図4.7にて白ヌキで表示した変数に対応する部分は，図4.9に示すものとなる。

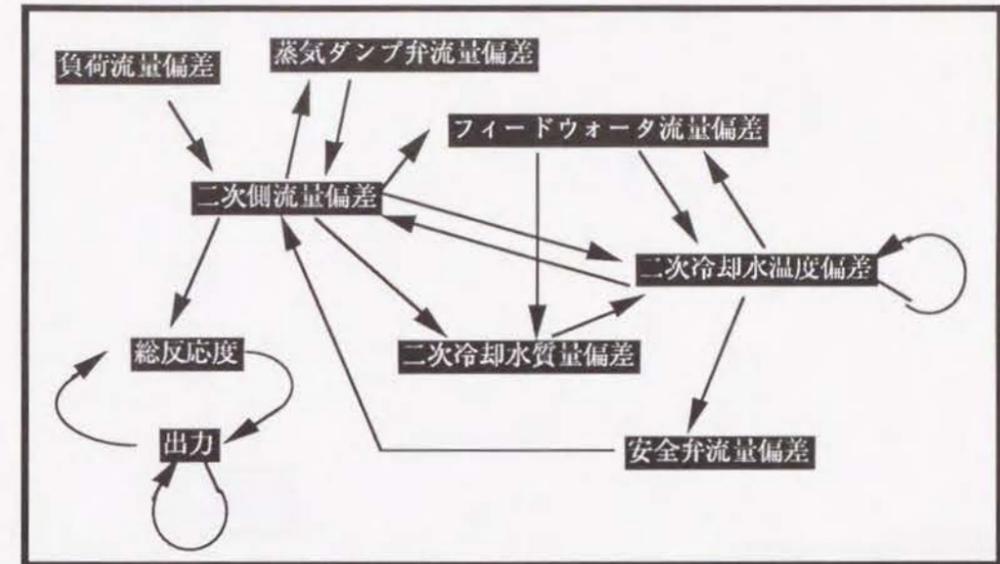


図4.9 縮約された依存構造

ただし，図4.8に示した依存構造ネットワークをモデル作成者がみた際に，多くの変数を省略したことによって，直接的な影響を表現していないと考えられる依存リンクが多数含まれていた。例えば，図4.8中に波線で示されたフィードバックは明らかに不自然である。また，加圧器内の圧力の変動は一次冷却水に影響が及び，その結果炉心内の反応度にも影響が及ぶと因果順序づけからは得られるが，そうならないように制御系が準備されているために考慮する必要がない。本実行例ではモデル作成者がこのようなものも含めて59個の依存リンクを対話的処理により影響のない依存関係として省き，依存構造を再構成した。図4.10は再構成した依存構造の全体図であり，最初にこの構造に基づいて定性関係を定義した。

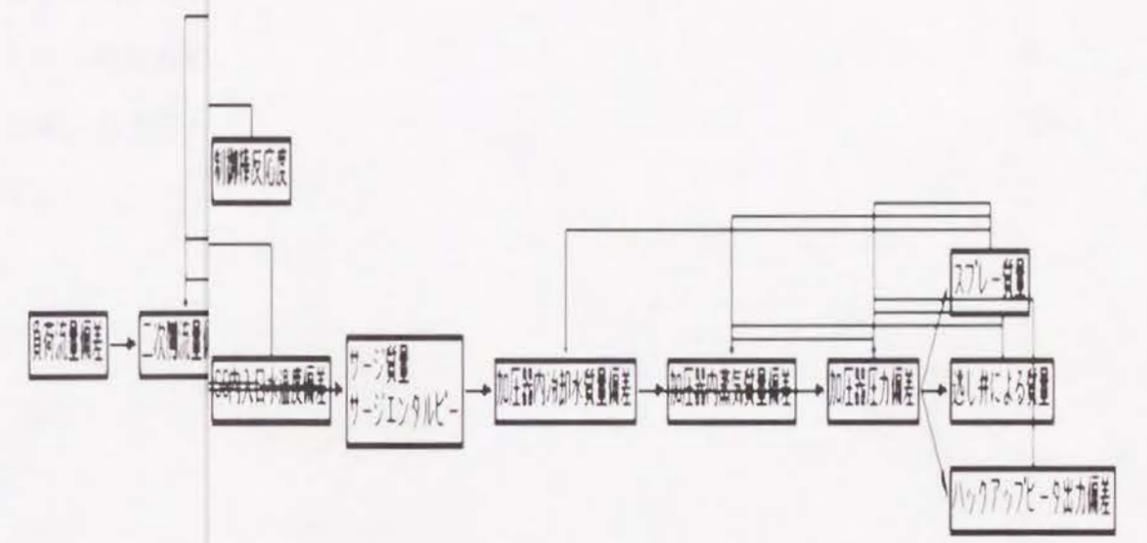
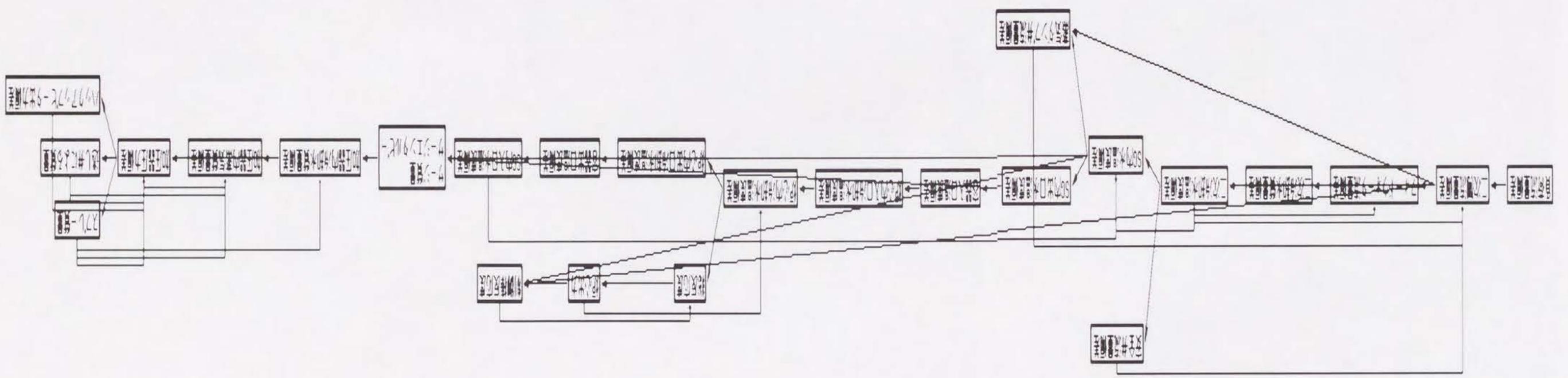


図4.10 再構成した依存構造



#### 4.5 負荷変化に対する因果的説明

4.4で作成した定性モデルにおいて、「蒸気調節弁の開動作」を初期状態（これ以外の機器は定常状態）として与える。なお、定性的制約データによって、この場合における「解釈のための時間」は11となる。

最終的に、定性モデルに2箇所の修正を必要としながらも図4.11-(a)のような因果ネットワークがえられた。図4.11-(a)では、縦軸が時間、横軸がプラントの機器に対応し、因果ネットワーク中の太線部や破線部のような因果シーケンスに関する因果的説明は図4.11-(b)のようになる。これは、負荷変化に対するプラントの「冷却水の温度上昇」や「蒸気ダンプ弁の開動作」の説明となっている。

なお、本手法で因果ネットワークの導出のために用いられたデータを表4.3に示す。

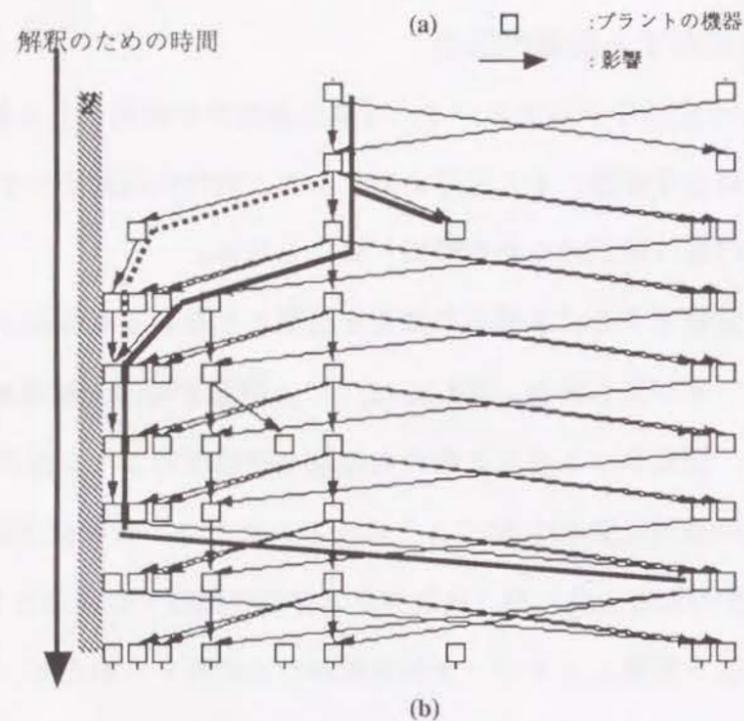


図4.11 導出された因果ネットワークならびに因果シーケンス

表4.3 実験データ

	定性モデル (縮約前)	定性モデル (再構成後)
変数の数	113	25
依存リンクの数	263	49
定性関係の数	***	159
関係伝播のステップ数	***	57

また、定性シミュレーション実行中に相反する影響が伝播された定性変数の状態遷移に関する曖昧さとしては、次のものが発生した。「蒸気発生器内の蒸発水量の減少」による「水位の上昇」という影響と「フィードウォーターポンプの噴出量の減少」による「水位の減少」という影響からは、「蒸気発生器内の水位が増加するか、減少するか、あるいは変化しないか」を決定できない。この場合に定性的制約データを用いることによって「水位が変化しない」ことがわかり、定性演算にあらわれる曖昧さは解消された。

#### 4.6 定性モデルの修正例

図3.1にて示した定性モデルの修正がどのように行なわれるかについて、4.5の因果ネットワークの導出過程で発生した例を用いて説明する。最初に作成した定性モデルを用いた定性シミュレーションでは、「解釈のための時間3」から始まるPI遷移の中で、変数間に影響を相互に伝播しても定量データの結果に一致する状態に遷移しないこと（図4.12-(a)）とシミュレーションの内容（図4.12-(b)）が計算機から出力された。これらに対する修正をそれぞれ以下に説明する。

(a)

定性的時間 3 から 4 で  
未確認の因果が、検出されました。以下の選択をしてください

(1) シミュレーションを続ける (スペース キー)  
(2) 現在の状況を調べる (TAB キー)

=====

以下の変数を調べてください ...

第 1 に  
デバイス : MAIN RELIEF VALVE,  
パラメータ : FLOW

第 2 に  
デバイス : PRIMARY HOT CONDUIT,  
パラメータ : OUTFLOW

(b)

時間 3

FLOW(24) in (MAIN RELIEF VALVE)

原因 : 二次側の温度高 TEMP(16) in (STEAM GENERATOR)  
⇒  
影響 : 安全弁流量増加

OUTFLOW(8) in (PRIMARY HOT CONDUIT)

原因 : 流出温度高 OUTLET TEMP(4) in (CORE VESSEL)  
⇒  
影響 : サージ量増加

原因 : 流入温度高 INLET TEMP(19) in (STEAM GENERATOR)  
⇒  
影響 : サージ量増加

原因 : 流出温度高 OUTLET TEMP(20) in (STEAM GENERATOR)  
⇒  
影響 : サージ量増加

原因 : SG内水温度高 PRIMARY TEMP(21) in (STEAM GENERATOR)  
⇒  
影響 : サージ量増加

図4.12 修正作業のための情報出力例

### (1) 「安全弁流量」に関する修正

取り上げた現象では「安全弁は作動せず、弁を通過する蒸気流量は0」であったのに対して、「解釈のための時間3」から始まるPI遷移のシミュレーション中では「二次冷却水の温度が定常時より高い」という状態から「安全弁の通過蒸気流量は増加する」という影響が発生する。この二つの変数間の関係記述をモデル作成者が見直した結果、「設定値を越える温度上昇」によって「安全弁が作動」という関係が記述されていなかった点に気づいた。実際に、定量データでは「二次冷却水の温度」は「定常時より高いが、設定値を越えていない」ことから、「二次冷却水の温度」を表す定性変数の定性軸に「安全弁作動設定

値」を付加し、それに基づいて「二次冷却水の温度が安全弁作動設定値を越える」ならば「安全弁の通過流量は増加する」という定性関係を付加した。この結果、定性シミュレーションでは「二次冷却水の温度」は「定常時より高いが、設定値を越えていない」という状態が導出され、「安全弁の通過流量は増加する」という影響は発生せず、この問題は解消された。

### (2) 「一次冷却水の加圧器側へのサージ量」に関する修正

「一次冷却水の加圧器側へのサージ量」に関する定量データは、「解釈のための時間3」からは「減少」に変化している。しかし、定性シミュレーションでは「サージ量を増加させる」という影響しか伝播されていない。この場合には図4.10の依存構造をモデル作成者が見直すことによって、「加圧器内の冷却水質量」から「サージ量」への影響を及ぼす依存リンクが必要であることに気づいた。これは、最初に縮約された依存構造をモデル作成者が再構成した際に誤って省いた依存リンクであり、2変数間に依存関係を定義することによって、この問題は解消された。

以上のように、作成された定性モデルに不備がある場合には、シミュレーションに基づいてどの変数に関して不備があるかを知ることができ、結果として図3.1に示されるようにモデル作成者が修正を行なうことができた。なお、図4.13が示す画面はモデル構築画面であり、プラントの系統図(図4.13-(a))、定量データ(図4.13-(b))、その定性的制約データ(図4.13-(c))、その変数が影響を及ぼす変数を示す依存グラフ(図4.13-(d))といった情報を表示する。図4.13-(e)には、図4.13-(d)においてマウスクリックで選択した変数との定性関係のリストが表示され、さらに図4.13-(f)によって選択された定性関係の内容を調べることができる。これらを用いて定性関係の修正を行なう。

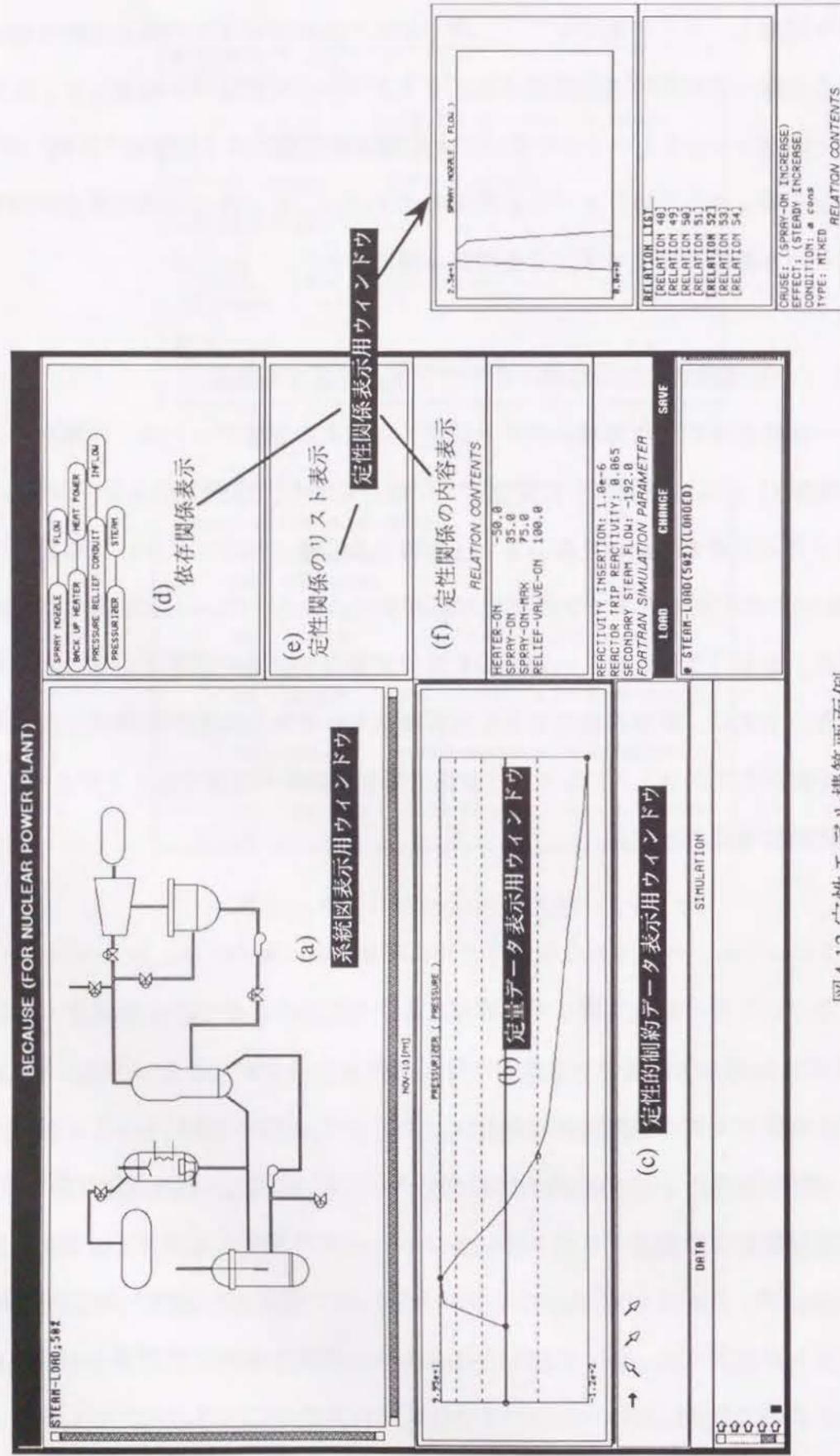


図4.13 定性モデル構築画面例

#### 4.7 結言

本章では、発電プラントに対する定量的解析の実行が可能な場合において、負荷遮断現象を因果的に説明することに焦点をあて、定性モデルの構築、定性シミュレーション手法、因果的説明の導出を述べた。その要点を、以下にまとめる。

- (1) 定性モデルの構築に際しては、求めた変数の対が直接的影響を表していると考えられない場合もあり、その際にはモデル作成者が、計算機から導出された依存構造をもとに、さらに依存リンクを除く必要もある。
- (2) 定量データに対する定性的視点をもとに導いた定性的制約データと「解釈のための時間」を用いて、影響の伝播による定性シミュレーションに発生する曖昧さが解消された。
- (3) 定量データとの対応づけを組み込んだ定性シミュレーションを、影響の伝播に基づいて実行することによって、現象に内在する変数間の相互の影響を明らかにし、そのシミュレーション過程に基づく因果的説明を導出することができた。

ただし、モデルの修正を行なう際の修正の手掛かりを計算機は提示するが、修正に関してはモデル作成者に依存していることから、今後モデルの修正を支援する方法の検討が必要である。

## 5 定性推論に基づくプロセス現象の因果理解支援方式

### 5.1 緒言

われわれの因果理解という認知プロセスは、プラントなどの動的システムを対象にすると、「機構理解のための一つ的手段」であると考えられる。因果理解を通して対象をより深く理解することによって、例えば、対象に何か操作を行えばどのようなことが動作系列として現われてくるかなどの推論を頭の中で行なうことが可能となる。このような「頭の中にいづく対象の心的表象」を、認知科学の分野では「メンタルモデル」と定義し、メンタルモデルの性質、構成要素、精緻化プロセスなどの観点から、いろいろと研究が展開されてきている<sup>[22]</sup>。

一方、1970年代から始まったCAI (Computer-Assisted Instruction) から現在のITS (Intelligent Tutoring System) にいたる研究の中では、学習者にとっては学習対象に関するメンタルモデルの形成が重要であると認識されている。その形成のために、教育システムとして何を機能として実現するかについていろいろと取り組まれた<sup>[49]</sup>。その結果、メンタルモデルの形成を目的とする教育システムとしては、STEAMER<sup>[23]</sup>やQUEST<sup>[50,51]</sup>といったシステムが開発された。

STEAMERの特徴的な機能としては、数値シミュレーションに基づくプラントの状態を観測あるいは操作するためのグラフィック・インタフェースが、ゲージなどのツールとして提供され、学習者はこれらのツールを用いて、プラントの現象を調べることを通してメンタルモデルの形成を支援する環境となっている。ただし、シミュレーション結果の因果的な説明を行なう機能は、実現されていない。QUESTでは、学習者のメンタルモデルの成長という点に焦点が当てられている。電気回路の構成部品に対して三つの観点から設定したレベルの異なる定性モデル群を用いて、学習者は自身のメンタルモデルの成長に対応す

るようにモデルを変換して、現象を理解する環境となっている。ただし、回路を構成し、定性シミュレーションを実行する定性モデル群は、現象に内在する因果よりは電気回路における電圧降下などの基本的概念の理解に焦点があてられた構成である。

ところで、プラントなどのように大規模で発生する現象が複雑な動的システムを目的にそった形で効率的かつ安全に運用していくためには、自動化技術の高度化とともに、運転員の知識の向上を目指した教育・訓練システムが望まれている<sup>[69]</sup>。しかしながら、現在の教育・訓練の形態は、運転マニュアルや機器ごとの機能説明書等を用いたクラス・ルーム・トレーニングと図5.1に示すような数値シミュレータを用いたレプリカ操作盤による模擬トレーニングが主体である。

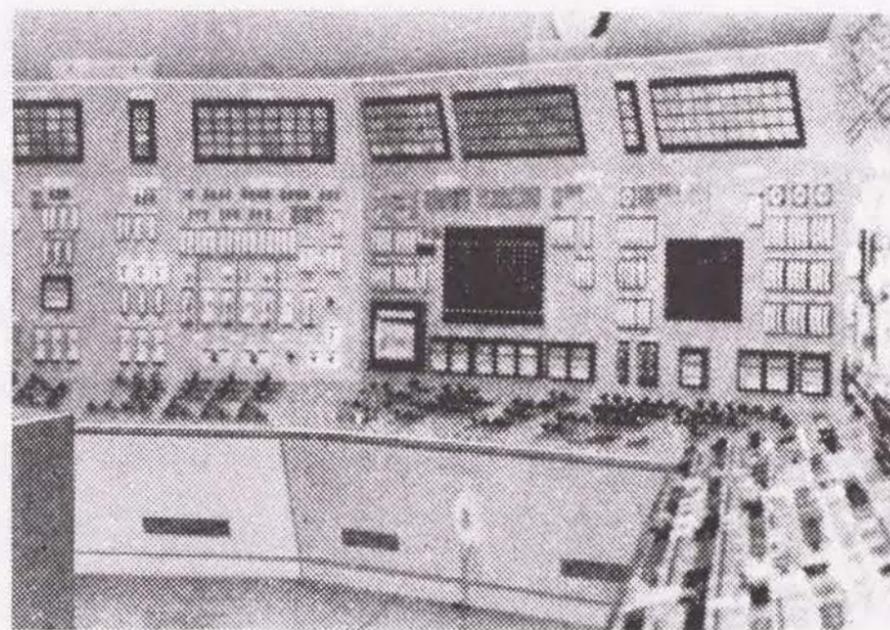


図5.1 レプリカ盤の例

これらは、運転員のタスク処理の観点<sup>[42]</sup>からみると、スキル・ベースもしくは手順に基づくルール・ベースの判断のための訓練であり、未経験の事象を処

理するナレッジ・ベースの判断のための訓練としては不十分なものであると考えられる。不測時の異常事態に対する的確な判断を実行できるためには、あるいはヒューマン・エラーの軽減のためには、対象システムに対する深い理解が必要である。今後の教育・訓練システムとしては、人間のナレッジ・ベースの判断を訓練すること、すなわち運転員自身の知識の向上を目的として研究されていかなければならない。

以上の観点から、本章では大規模な動的システムのダイナミクスに対する因果理解に焦点をあて、運転員の対象システムに対するメンタルモデルの形成支援を目指した教育・訓練の形態について考察する。ここでは、対象システムの定性モデルを学習者のメンタルモデルの部分的表象であるとの立場から、因果理解支援方式を提案する。すなわち、学習者は対象システムに関する自らのとらえ方に基づいて、計算機上の物理モデルの構成を行ない、この構成が反映された定性モデルを用いて与えられた現象の因果的説明（計算機内部では、因果シーケンスとして表現される）を導出する教育・訓練環境を構築する<sup>[2,54]</sup>。このための支援機能を実現することにより、学習者の理解の深化をうながし、その結果としてメンタルモデルの形成を支援できると考える。

本章では、5.2において、因果理解支援方式の設計思想について説明する。まず、人間の保有するメンタルモデルの特徴ならびにその形成に関連する理解のための方略に関して、認知科学からの知見を整理し、それに基づいて理解プロセスを重視した支援方式を提案する。5.3においては、支援方式に基づく階層型定性モデルの表現とそれを用いた定性シミュレーションについて説明し、学習者とのインタラクションのためのグラフィックインタフェースについても示す。5.4においては、本方式を原子力プラントに適用した結果を示す。

## 5.2 因果理解支援システムの設計思想

現象の因果理解を通して、学習者の対象に関するメンタルモデルの形成を支援するために、システムの機能として何をどの様な観点から構築するかは重要なことである。本節では、この立場からの設計思想を説明し、それに基づく支援方式を提案する。

### 5.2.1 認知科学からの知見

メンタルモデルの性質としては、次のような点が指摘されている<sup>[39]</sup>。

- (1) メンタルモデルは不完全であり、しかも不安定である。例えば、人間は利用していたシステムの詳細な部分をしばらく用いないと忘れてしまう。
- (2) メンタルモデルは人間によって異なり、しかも時としては思い込みによる場合もあり、科学的根拠のないものである。

このように曖昧で記述が困難なメンタルモデルを、対象に関する情報の獲得とともに不完全な部分を補完しながら、われわれは常に主体的に修正し、形成を繰り返していると認識されている。したがって、この形成に対しては人間が対象を理解しようとするときの方略が重要になってくる。そこで、理解のための方略として以下の点に注目する。

#### ・視点を用いた多重モデルの利用<sup>[64,76]</sup>

空間上にある物体に対して、人間はそれを理解するために様々な角度からそれを眺めて、いくつもの側面からの情報を活用している。ただし、この例は対象が物理的なものであり、それは知覚の範囲にとどまるが、より一般的には人間はある状況を設定して、その中で情報を例示化することによって、現象の一側面を把握しようとしていると考えられる。このように状況を設定して情報を例示化するという行為そのものが、理解プロセスにおける視点となる。視点を

用いることによって、同一の対象にたいして複数のモデルを構築し、さらにそれら複数のモデルを使い分けながらモデルそのものの洗練を繰り返している<sup>[61]</sup>。

#### ・定性情報の利用<sup>[22]</sup>

物理システムのダイナミクスを理解する場合に、そこに含まれる変数の定量情報を人間は保有する知識や過去の経験をもとに解釈する。この場合に、人間は定量情報を定性的にとらえなおしていることが多い。例えば、プラントのプロセス量の変化をとらえる場合において、厳密な観測値に対して正常値からの偏差が大きいとか小さいという定性量や、その変化の状態が増加から減少に変化するという定性変化量をもとにしている。さらに同じ物理量であっても、人によって特有の観点を加えてとらえなおすことがある。このような定性量をもとにして因果的説明を行なう。

以上のような理解のための方略を、因果理解支援方式として取り入れる必要があると考える。

#### 5.2.2 因果理解支援方式

ここでは、前節で述べた人間の理解プロセスにおける方略を反映する、次の二点からの支援機能を導入した因果理解支援方式の枠組みを提案する。

##### (1) 物理的階層に基づく視点の設定支援

大規模システムの現象が内包する情報は多様であり、それらを選択して利用するための視点の設定を次のように考える。例えば、大規模システムの構成要素間には、ある要素が他の要素の部品であるという物理的階層が存在している。この物理的階層にそくして構成要素ごとに階層を設定することによって、状況を設定することを考えることができる。また、これに対してある要素の集合が

機能的にみて一つの要素のように振舞い、それらの集合間に階層が存在しているとみることもできる。これを機能階層と呼ぶ。このような機能という側面から構成要素をまとめあげ、状況を設定することも可能である。ただし、機能階層に比べて物理的階層は構造関係として確定する場合が多く、ここでは視点設定とその移動にこの階層を用いる。具体的には物理的に詳細度の異なるデバイス・モデルを、学習者の視点に基づいて組み合わせることで、現象に対するモデルを構築する。また、このような視点を変えることによって生じる多重モデルの管理を計算機側で実行する。

##### (2) 定性情報の活用支援

定性情報を取り扱うことは、従来から定性推論の枠組みの中で行なわれてきた。物理的定性量として連続量を有限個の区間に分割し、区間と瞬間値の交互の繰り返しによる表現が用いられてきた。区間内では、定性値として同一視することによって、人間の常識的思考に適合するという立場に基づいている。このことから、支援方式として定性表現を基本として用いる。ただし、定性値の表現に対して学習者の観点を埋め込むことができる。例えば、連続量としてのプラント変数における正常値や周辺機器が作動する閾値等は計算機側が予め定性軸上に提示するが、さらに学習者側が考える概念的に重要な値（例えば、かつて異常が発生した値など）を、その軸上に表現することができる。ここで、その表現に基づく定性モデルを用いて定性シミュレーションを実行し、現象に対する定性変数間の因果関係の導出を支援する。

以上のような点に基づく支援方式の枠組みを、図5.2に示す。図5.2の中心部に示すように、数値シミュレータからえられた現象に内在する因果関係を、定性シミュレーションを用いて導出する流れの中で、学習者側と計算機側の処理するタスクがそれぞれ存在している。

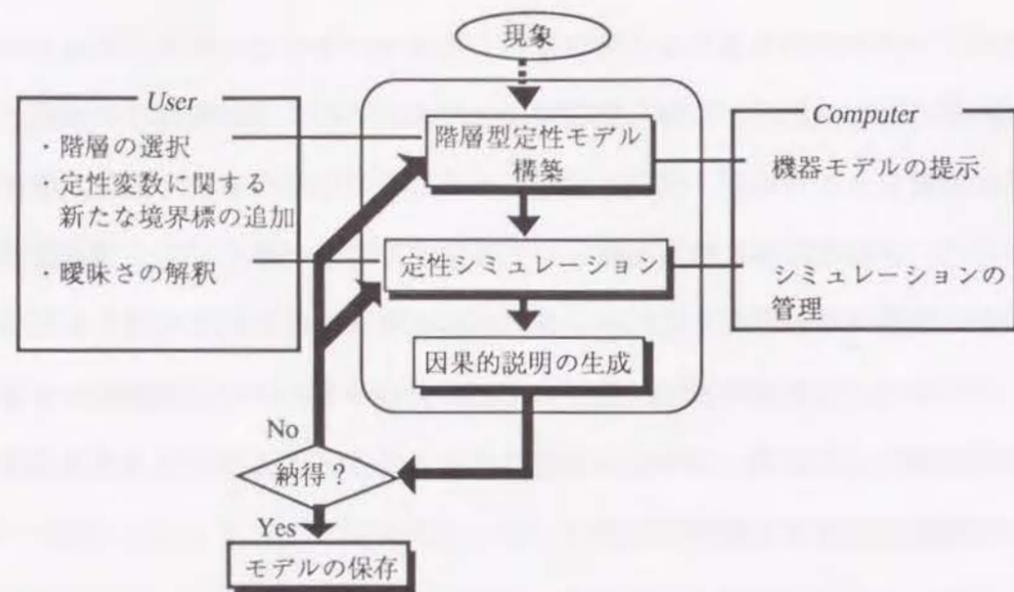


図5.2 因果理解支援方式

図5.2の流れにそって説明すると、現象に対して、先ず計算機に保存されている定性モデルが学習者側に提示される。次に学習者は自己の観点に基づいて、その定性モデルの物理的階層の選択や物理パラメータの定性的表現を変更して、階層型定性モデルを構築する。この階層型定性モデルの初期状態を学習者が設定した後に、定性シミュレーションが実行される。この場合に定性演算に基づく曖昧さが発生すると、計算機は学習者にその状況を提示し、その選択を学習者が行なう。最後に定性シミュレーションの結果に基づいて、現象に内在する因果関係が導出され、因果シーケンスを示すグラフィックが学習者に提示される。学習者はその因果関係が納得できない場合には、階層型定性モデルの構築あるいは定性シミュレーションの任意の時点に戻って、納得するまでこの流れを繰り返す。

本方式では、理解のための方略をもとにした学習者のメンタルモデルを計算機上の定性モデルに表象するという立場であり、単に定性情報や因果関係を提示するだけでなく、その導出過程をトランスペアレントにする点に特徴がある。

このことは、最近の「計算機の道具性」を考察する立場の認知科学の研究においても、「状況の知 (Knowing With)」という言葉で議論されており<sup>[57]</sup>、例えば上述のような主体的な方略をもとにして人間がおかれた状況とどうインタラクトするかが重要な検討課題となる。この点をふまえて教育・訓練システムにおいては、「表象-吟味のループ」の支援の重要性が指摘されている<sup>[69]</sup>。提案する支援方式のループは、このような観点からも学習者の理解の深化をうながすと考える。

### 5.3 因果理解支援システムの構築

本節では、5.2で述べた支援方式を実現する階層型定性モデルと、それを用いた定性シミュレーションによる因果シーケンスの導出について説明する。また、学習者とのインタラクションのためのグラフィックインタフェースについても示す。本システムは、Symbolics3650上でLispとFlavorsを用いて記述されている。なお、数値シミュレータはFortranを用いて記述された既存のものである。

#### 5.3.1 階層型定性モデル

デバイス中心の物理的階層を表現するために、オブジェクト指向の枠組に基づいて記述している。デバイスごとに物理的詳細度に基づく階層を設定し、各々のデバイスはその状態を表現するための定性変数を含んでいる。

##### (1) 定性変数の表現

定性変数は、定性値と定性微分値を用いる。定性値は瞬間値と区間をもとに表現されるが、記号的表現を用いずに連続量を $[-1, 1]$ ,  $[0, 1]$ ,  $[-1, 0]$ の区間に変換し、このリスト上に数値を用いて瞬間値と区間を表す。これは、学習者の観点を埋め込むための定性軸の変更を、計算機で扱いやすくするためであり、このような正規化によって、変数相互の定量的オーダは失われている。定性微分値

は、その定性値の変化方向を表現するために、一階の微分値として定性的に表現した「増加」, 「安定」, 「減少」を用いている。例として、原子力プラントにおける加圧器の圧力の定性値の表現をとりあげる。いくつかの周辺機器のスイッチングを引き起こす値が、瞬間値として定性軸上に表現されている。機器の稼働する区間は、図5.3に示す範囲のようになっている。定性軸は瞬間値のリストとして、表現される。

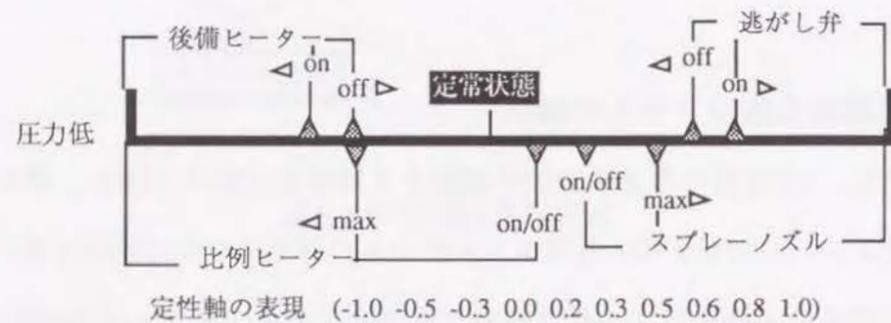


図5.3 定性変数の表現例

## (2) 定性変数間の関係

定性モデルを用いて挙動シミュレーションを行なうには、定性微分方程式を用いた制約充足の手法がある<sup>[30]</sup>。この手法においては、制約式を満足する定性値の組合せがすべて計算され、可能なかぎりの定性的挙動が導出される。しかしながら、制約充足の手法においては、因果的説明をえることはできない。ここでは、定性変数間の関係を表現するのに、制約式を用いるのではなく、定性変数間に存在する関係を二項関係として個別に記述する方法を行なっている。例えば、加圧器内部のヒーターの作動によって冷却水が加熱され、加圧器内部の圧力が上昇をはじめるという現象は、ヒーターの熱量を表すパラメータの定性値が加圧器の圧力の定性微分値に作用したとみなせる。この時に、ヒーターの熱量パラメータに、加圧器の圧力パラメータへの一次遅れの正（増加）の関

係を記述しておく。ここではある定性変数の定性値に影響を与える関係を比例関係、定性微分値に影響を与える関係を微分関係と呼ぶ。図5.4に示すように、関係記述は条件照合部と影響伝播部から構成され、識別用と状態記述用のリスト表現を含む。状態記述部の定性値として、「以上」、「以下」、「より大きい」、「より小さい」などの区間の表現も、前述のリストによる定性軸の表現を用いると容易に記述される。

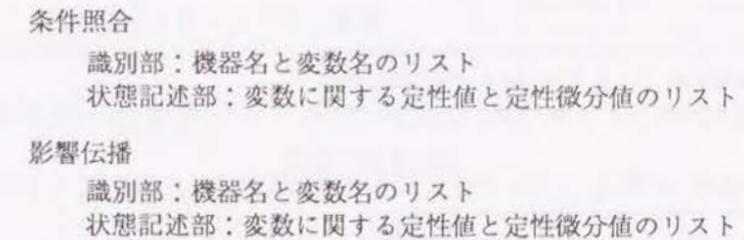


図5.4 定性関係の記述内容

## (3) デバイスの表現

デバイス・オブジェクトの内部には、デバイスを特徴づける定性変数のインスタンスを保持する。また、詳細化した記述のデバイス・オブジェクトへのリンクを保持することにより、階層を変更した際には同時に定性変数オブジェクトのシミュレーションモデルへの組み込みならびに取り除きが行なわれる。このように記述すると、簡潔性を失わずに、階層性のあるモデルの構成が容易となる。

## (4) プラントの表現

視点を反映したプラントのモデルとして、シミュレーション・オブジェクトというクラスを記述する。このインスタンス変数の値として、シミュレーション・オブジェクトを構成するデバイスのインスタンスを保持する。

図5.5は、(1)から(4)までの考えに基づいたFlavorsのクラス表現を示したものである。クラス名の下にそれぞれに応じたスロット名を図中に示す。以上の(1)から(4)に基づいて、階層を持ったオブジェクトから計算機上に構成したシミュレーション・オブジェクトを、階層型定性モデルと呼ぶ。

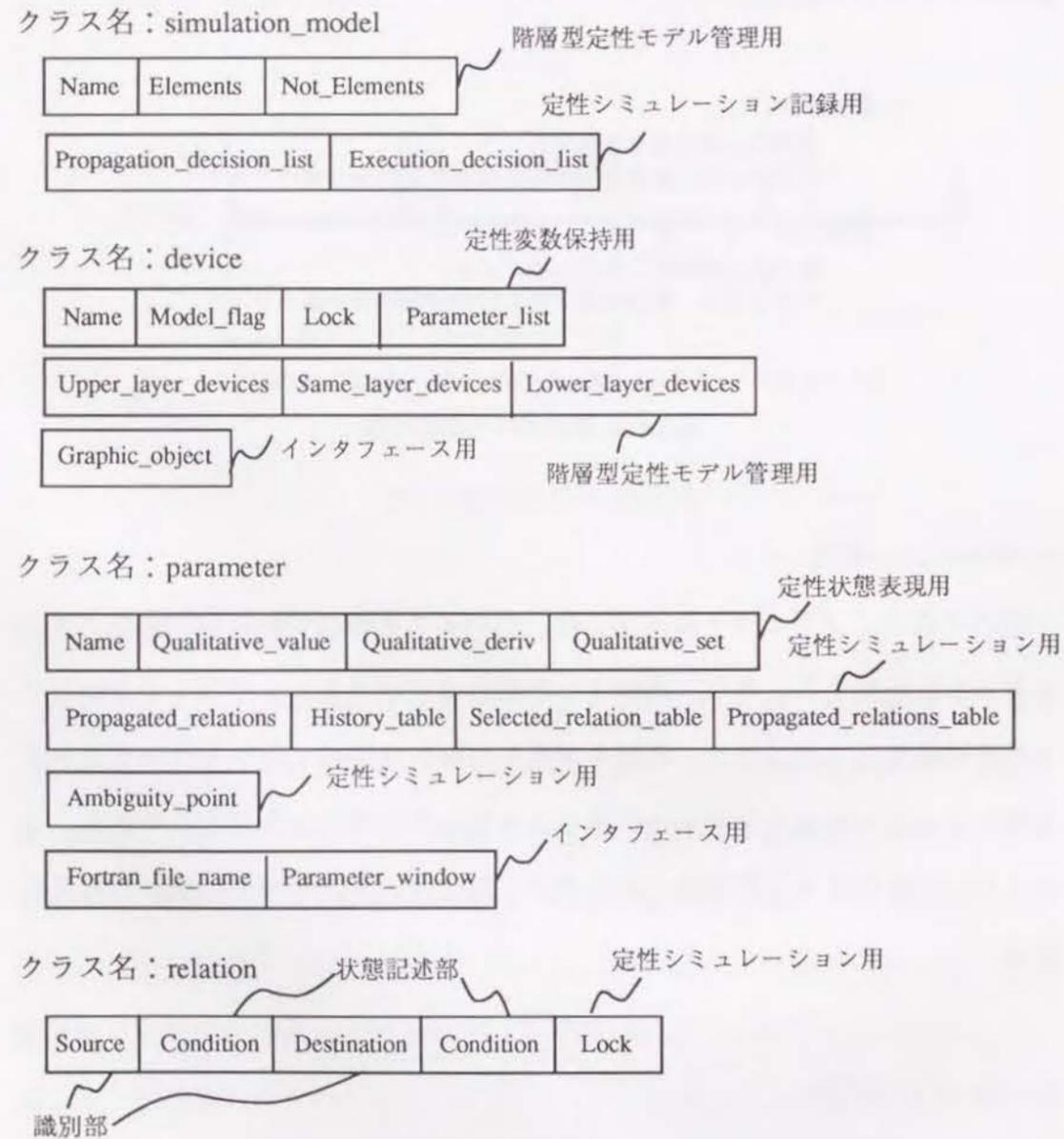


図5.5 クラス表現

### 5.3.2 定性シミュレーション

定性変数の定性状態（定性値と定性微分値の対）に対応して、各々の定性変数ごとに記述された関係オブジェクトが、デバイス・オブジェクト間を伝播される。この伝播された関係オブジェクトをもとに変数オブジェクトの状態遷移を実行する。以上の処理は、関係を伝播する処理と関係を実行する処理の二つの処理から構成される。

#### (1) 関係の伝播処理

以下、伝播元の変数オブジェクトを、Cause-パラメータ、その逆をEffect-パラメータと呼ぶことにする。この伝播処理は、図5.6に示す通りであるが、Effect-パラメータに複数の関係オブジェクトが伝播されることがある。この際には、Effect-パラメータの中の履歴管理テーブルに記録される。

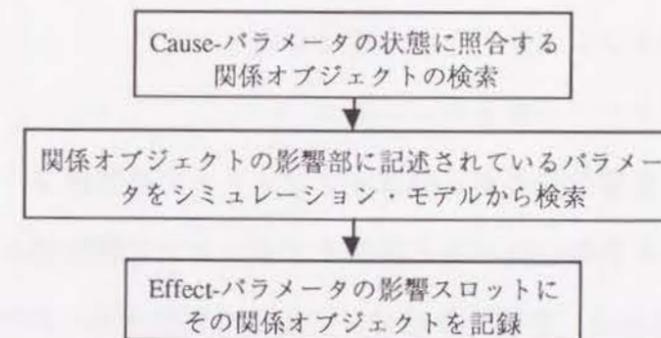


図5.6 関係の伝播処理の流れ

#### (2) 関係の実行処理

この処理は、関係オブジェクトの中の影響部の状態記述をもとに、各々の変数オブジェクトの状態（定性値と定性微分値）を更新する処理である。したがって、関係オブジェクトの伝播がない変数オブジェクトについては、現在の定性状態に基づいて状態遷移を実行する。関係オブジェクトが伝播された変数オブ

ジェクトに関しては、以下の流れにそった処理を行なう。場合としては二つあるが、学習者が行なう処理はほとんど同じであり、この時には定性シミュレーションは中断している。

#### (a) 伝播された関係オブジェクトが一つの場合

伝播された影響の定量的な強さによっては、変数オブジェクトの現在の状態に影響がでない場合もある。これを定性モデルだけから決定することは不可能なので、変数オブジェクトの現在の定性状態と伝播された関係オブジェクトの内容を学習者に提示し、その関係オブジェクトに記述された影響に従う状態遷移を行なうかどうかを学習者が選択する。

#### (b) 伝播された関係オブジェクトが複数の場合

(a)と同様に変数オブジェクトの現在の定性状態と伝播された関係オブジェクトの内容が提示される。この時に学習者の可能な選択は、次の4つである。

- (i) 関係オブジェクトを全く選ばない
- (ii) 関係オブジェクトを一つ選ぶ
- (iii) 同じ影響記述のある関係オブジェクトを複数選ぶ
- (iv) 異なる影響記述のある関係オブジェクトを複数選ぶ

(i)の場合は、因果的に影響をうけていてもその影響が弱い状況に対応する。  
(ii), (iii)の場合は、その影響記述に基づく状態遷移を行なう状況に対応する。  
(iv)の場合には相反する影響なので、その中でさらに学習者の解釈に基づいて選ばれた影響記述に基づく状態遷移を行なう状況に対応する。

(a), (b)ともに定性的表現だけでは学習者は適切な選択を行なうのが困難であるために、学習者に利用可能な情報として、各々の関係オブジェクトのCause-パラメータとEffect-パラメータに関してあらかじめ得られている数値シミュレータのトレンド・データが示される。

以上のように場合分けが発生した際には、学習者が定量的情報や保有する知識をもとにして関係オブジェクトを選択することによりシミュレーションを続行し、その場合分けの記録を残す。その記録をもとに学習者は、任意の状態に戻ることが可能となる。この定性シミュレーションでは学習者は、現象を定性的に解釈していく過程で、因果的に複雑な時点を認識し、計算機の提示する情報をもとに解釈を実行するという位置づけになる。これら関係オブジェクトの伝播の過程が、因果シーケンスとして後述するグラフィックインタフェースで表示され、現象に対する因果的説明となっている。

この関係の実行処理の流れをまとめたものを、図5.7に示す。

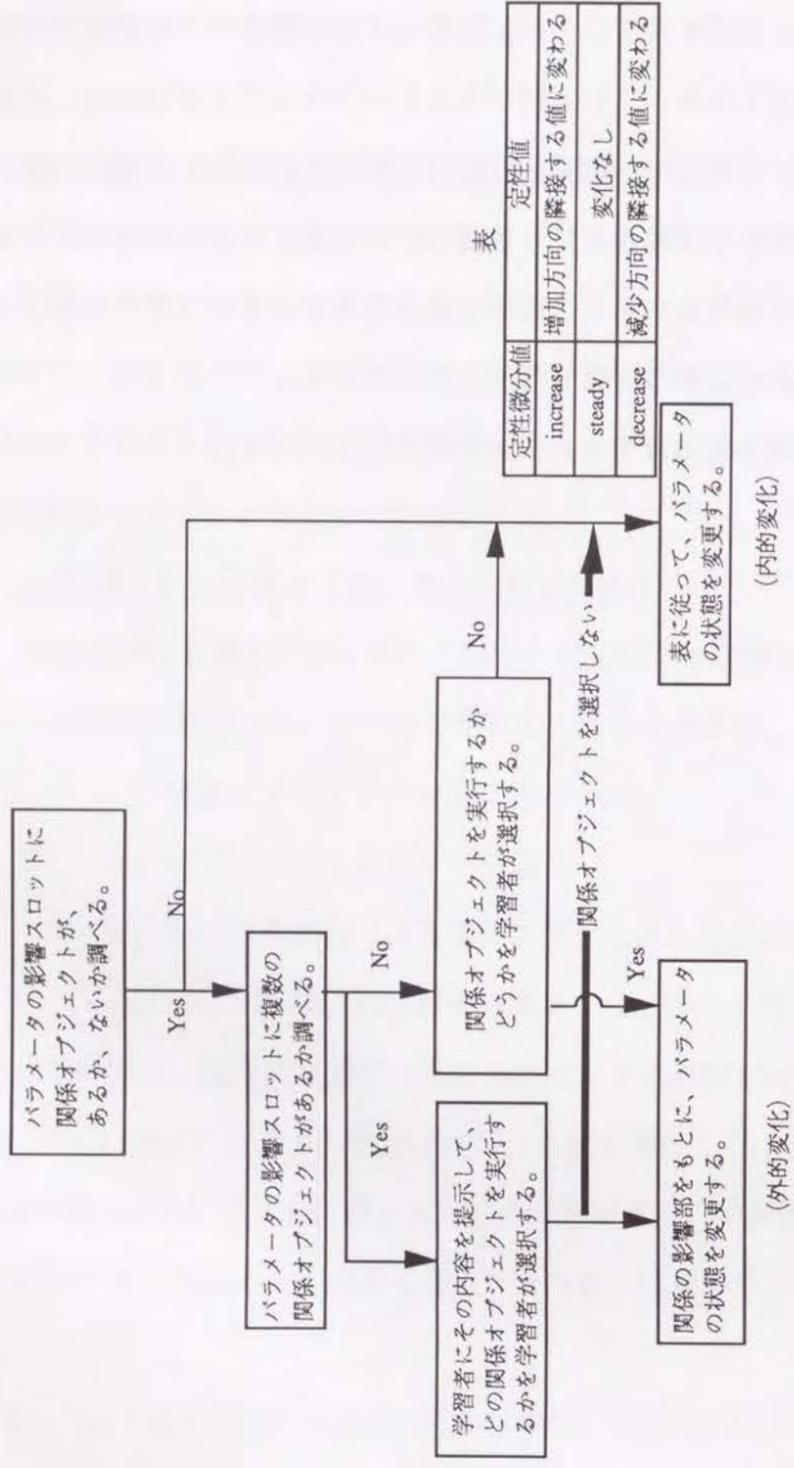


図5.7 関係の実行処理

### 5.3.3 ユーザインタフェース

ユーザと支援システムとのインタラクションは、基本的にマウス操作によるグラフィックインタフェースで実行される。以下、画面例をもとに階層型定性モデルにおける選択や定性シミュレーション実行時の操作を説明する。なお、定性シミュレーションからえられる因果シーケンスを表示するグラフィックインタフェースについては、適用例の中で示す。

#### (1) 階層型定性モデルの選択

学習者は、図5.8のようなプラントの系統図上のグラフィックスをマウスでクリックすることによって、次のような情報を得ることができる。例えば、機器の機能仕様書を見たり、階層を変更することができ、階層を変更した結果はグラフィックスとシミュレーション・オブジェクトに反映される。また、機器に含まれる定性変数の表現と現在の値は、図5.8が示すようにマウスクリックの後に変数リストのポップアップ・メニューが現れ、選択した結果がプラント系統図右横の変数表示ウィンドウに表示される。この定性変数の定性的表現を変更する場合や、その初期値を設定する場合には、変数表示ウィンドウ上で、マウスクリックにより新たな定性区間を作成したり、初期値の設定を行なう。定性変数間の関係を新たに定義したり、変更するには、この変数表示ウィンドウとは別に記述用のエディタが現れる。

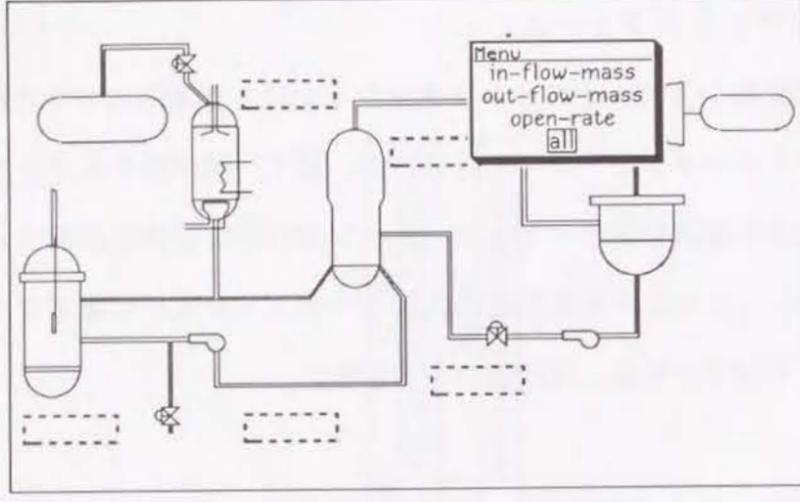


図5.8 定性変数表示のポップアップ・メニュー例

(2)定性シミュレーションの実行

初期状態を設定して定性シミュレーションを実行している際には、定性情報を観測するツールがある。これは、学習者が注目している定性変数の定性的変化の履歴と各々の時点ごとの定性状態を観測するためのものである。また、関係の実行処理の際に、学習者が関係オブジェクトの選択を実行する上で必要となる定量データを観測するツールがあり、図5.9において左にある二つのウィンドウがこれらを組み合わせたものである。また、関係オブジェクトの伝播の状況を表示するウィンドウをプラント系統図の下に備えている。関係オブジェクトの選択に関しては、別のウィンドウを用いて行なう。

以上の(1)ならびに(2)は、図5.9に示す画面上で行なうことができる。

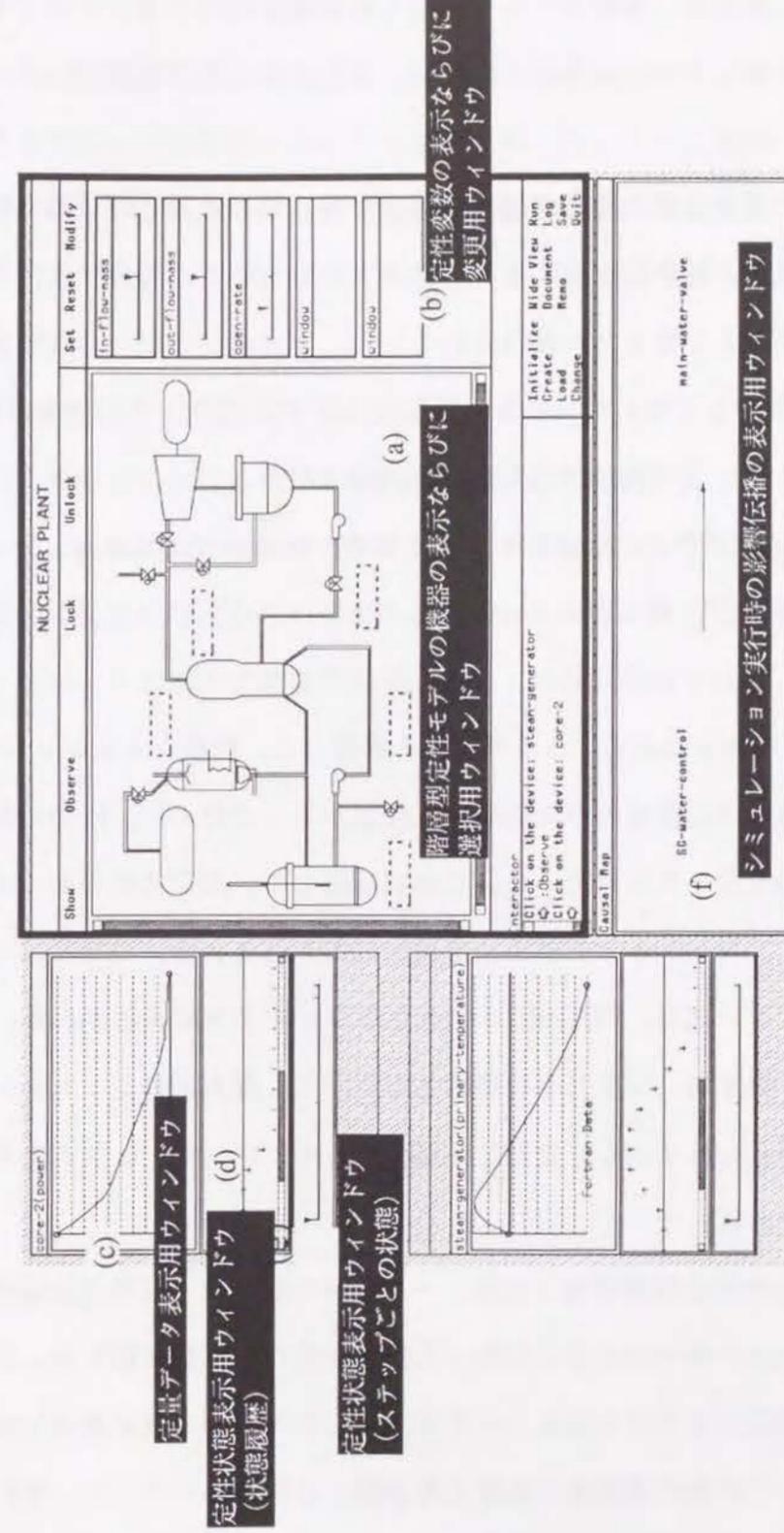


図5.9 階層型定性モデル構築ならびにシミュレーション実行画面の例

## 5.4 適用例

原子力プラントに対して、本システムを適用した結果を説明する。ここで構成したプラントの定性モデルは、機器の数が41、定性変数の数が120となっている。

発電機に接続した電力系統の負荷が急激に減少する、負荷遮断と呼ばれる現象がある。50%の負荷遮断が発生すると、プラントのいろいろな部分に変化が生じる。例えば、次のようなことがおこる。

- ① 一次側の一次冷却水において、いったん温度上昇が起こり、その後ゆるやかに下降していき、定常時よりも低い値になる。
- ② 二次側の蒸気ダンプ弁が、しばらくすると開き、その後閉じられる。
- ③ 原子炉内に制御棒が、挿入される。

この現象に対して学習者は次のようなモデルを構築した。数値シミュレーションの結果から加圧器の周辺機器の中で作動する機器があったために、圧力に対して詳細化された加圧器モデルを用いた。これに対応して、周辺機器のヒータやスプレー弁がシミュレーションモデルに自動的に組み込まれる。

導出した因果シーケンスは、図5.10に示すようなウィンドウに表示される。このウィンドウは、横方向にプラントの機器名を表示し、縦方向にシミュレーションのステップがとられている。ただし、図5.10はスクロールするウィンドウをつないだものである。

ここで導出した最終的な負荷遮断の因果シーケンスの図には、「蒸気加減弁の急閉に伴い、一次系の側で制御棒を制御する動作が起こり、制御棒を炉心に挿入して原子炉の熱出力を下げる因果シーケンス、二次系の側で蒸気量のフローの急減によって、二次系の循環水の温度上昇が起こる因果シーケンス、更に蒸気ダンプ弁が開く因果シーケンスが含まれている。また、時間がたつにつれて、蒸気発生器で一次系と二次系の間に相互作用が生じ、その結果が各々に影

響を及ぼしあう」ことが把握できる。

このシーケンスの導出には、29ステップを要し、シミュレーションの後半には多くの機器間に因果が発生し、学習者は数度定性シミュレーションのある時点まで戻って再解釈を実行する必要がある。なお、破線で示されている関係オブジェクトは、関係の実行処理中に影響が弱いとみて実行されなかったことを示している。さらに詳細に各々の因果関係を把握するために、次のような機能がある。図中において、ノードはデバイス・オブジェクトに、アークは関係オブジェクトに対応し、図5.11に示すように、ノードをクリックすることで、各々の時点における機器の定性変数の状態の情報を、アークをクリックすることで、伝播された関係オブジェクトの情報を得ることが可能である。また、学習者の任意の時点からの再解釈を行ないたいという要求に対して、シミュレーション・モデルの履歴管理機構を用いて容易に実行される。

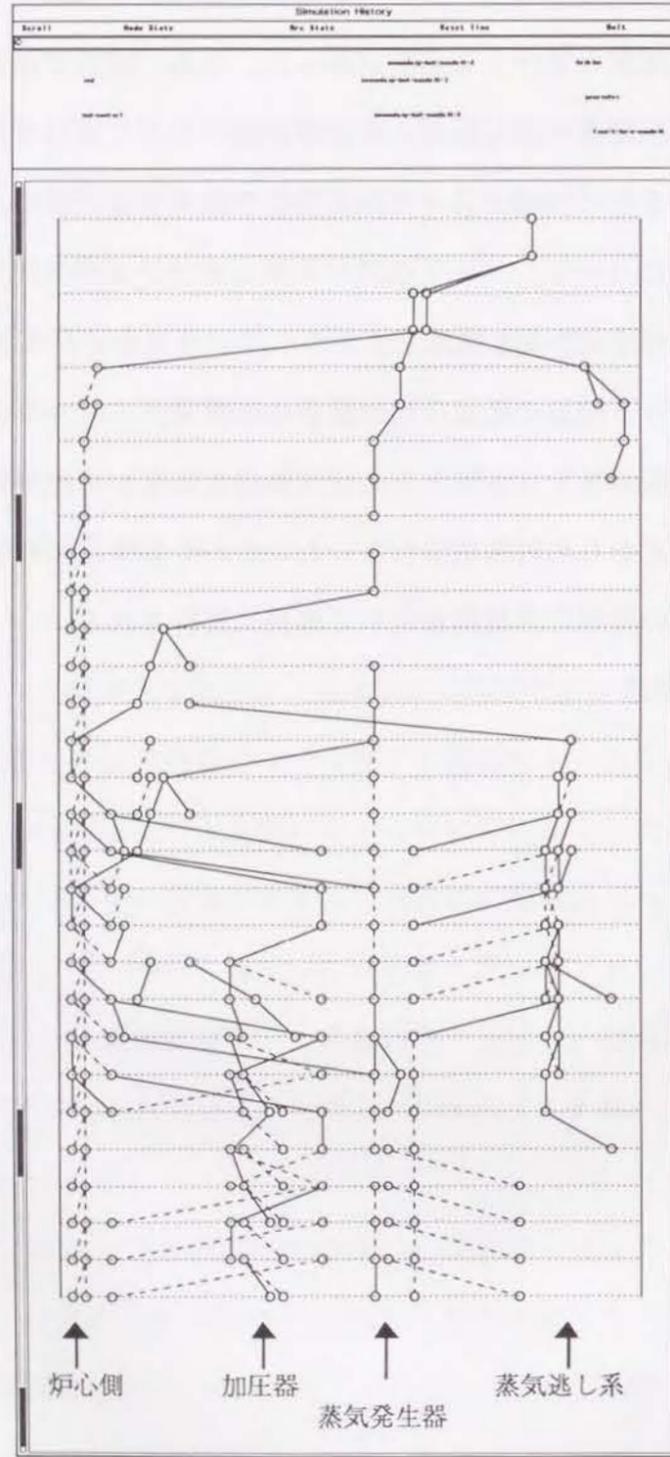


図5.10 因果シーケンスの例

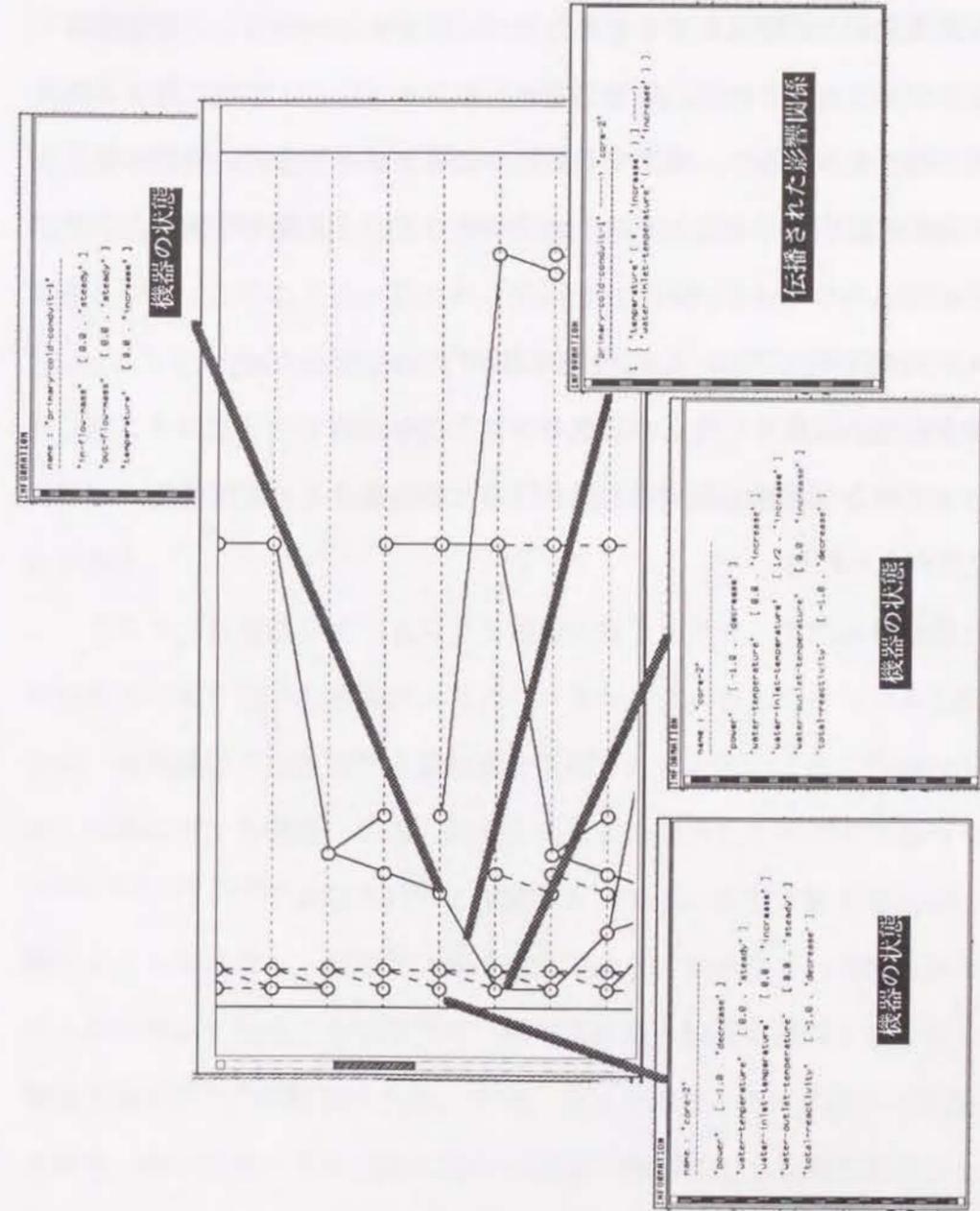


図5.11 シミュレーションの時点ごとの情報表示例

## 5.5 結言

本章では、因果理解がメンタルモデルの形成を促進するという立場から、階層型定性モデルならびにそれを用いた定性シミュレーションを提案した。階層型定性モデルは、学習者のメンタルモデルに基づく対象システムのとらえ方を階層的な機器をもとに構成したものであり、定性シミュレーションは局所的な影響伝播をもとに実行される。定性シミュレーションによる現象に対する因果的な説明を導出する過程で、相反する影響の伝播などの因果的に解釈が複雑な時点を学習者に認識させるとともに、納得のいくまで「表象-吟味のループ」を利用するツールとなっている。

本システムを原子力プラントを例に専門家が評価したところ、

- ・現象を定性的に捉える視点の設定ができる点が良い
- ・わかっているつもりでも、考え直す時のための道具として使いたい

という意見がえられた。

## 6 定性推論を用いた因果伝播による故障木作成手法

### 6.1 緒言

因果理解は、故障診断を行なう際には必要であり、実際故障箇所を同定するのみならず、設計へのフィードバックを行なうにはなくてはならないという点でも、工学的な立場から重視される。しかしながら、対象の規模が大きく、また時間的に故障状態が変化する場合には、専門家といえども因果関係の把握は困難となる。このことは、従来から信頼性工学で行なわれてきた故障木解析<sup>[66, 67]</sup>においても問題になっている。故障木解析とは、観測された事象から故障を引き起こす原因となる事象を順次調べることにより、考えられる故障原因候補を求める作業であり、その過程でそれぞれの事象の論理的関連が明確となる利点がある。

ところで、計算機システムによる故障診断としては、専門家の知識を記述した知識ベースを用いた手法がエキスパートシステムとしていろいろと開発されたが、知識獲得や知識管理の問題点が指摘されている。これに代わって最近では、対象に関する構造、機能、因果関係などを記述したモデルに基づく手法<sup>[10, 13, 47, 61, 63, 68, 72, 77, 78, 79, 80]</sup>がいろいろと提案されている。故障木解析を実行する計算機システムの構築も、信頼性工学では故障木を、あらかじめ準備された部分故障木をつないで構成する手法<sup>[45]</sup>や、観測兆候から因果を逆向きにたどることで構成する手法<sup>[35, 48]</sup>が提案された。また、最近上述のモデルに基づく手法の研究の中で、深い知識とよぶ対象に関する構造、物理法則、故障原理などをもとにした推論によって故障木を構成する手法<sup>[78, 79, 80]</sup>が提案されている。観測された兆候に対して、対象のモデルに基づいて故障原因候補を導出していく枠組みは、未経験の故障への対応、ならびに規模の大きなものや振る舞いが複雑なものにおいて熟練技術者が見落とす可能性のある故障原因候補のシステムティックな

導出などの点で大いに期待されている。

本章では、定性推論を用いた因果伝播による故障木作成手法について提案する<sup>156)</sup>。故障木作成の際には、観測兆候から対象の未観測部分の振る舞いを推定することが必要であり、計算機処理のための対象のモデルの表現ならびに振る舞いの推論方式として、次の点を十分に考慮しなければならない。

(1) 一般に、故障が発生した時点と観測時点が同時点である保証はなく、故障発生後の対象の状態変化を考慮した故障木作成が必要である。しかしながら、従来は故障状態は変化しないという仮定などによって、対象の状態変化を明確に取り扱う議論があまりなされていなかった。

(2) 推定した対象の振る舞いの中で故障状態と考えられる部分を同定する際には、正常動作モデルのみを用いる考え方<sup>160)</sup>と、正常動作モデルならびに故障動作モデルを用いる考え方<sup>13, 47)</sup>がある。故障動作モデルによって故障原因候補の特定が容易になることから、モデルにおいて考えられる故障を定義するのが望ましい。

また、故障原因候補の検証の際には、求めたそれぞれの故障原因候補からの波及する影響から観測されるべき兆候のチェックが、基本作業とされている。したがって、次の点についてもここでは考慮する。

(3) 作成した故障木に現われる故障原因候補の絞り込みのために、モデル上での故障の波及シミュレーションが望まれる。

6.2においては、以上の点に対して、故障発生後の対象の状態変化をとらえた故障木作成の基本的な考え方を提案し、モデル記述ならびに振る舞い推定への定性推論の利用、モデル上での故障の波及シミュレーションによる故障原因候補の絞り込み、モデルにおける故障の定義について説明する。また、システムのソフトウェア構成についても説明する。6.3においては、電力保護リレーに対して、本手法を適用した結果を説明する。

## 6.2 故障木作成手法

### 6.2.1 対象の状態変化をとらえた故障木作成の基本的な考え方

故障木を作成する際には、観測兆候、対象を構成する機器の正常ならびに故障動作に関する知識、機器間の因果関係をもとに、未観測の機器の部分の動作状態を求める過程で、故障と考えられる機器を同定する。この時に、故障状態は変化しないという仮定をもとに、未観測な部分の動作状態を求めて故障原因候補を導出すると問題が発生する。例えば、電力保護リレーのシーケンス回路のように復帰回路を含んでいると、遮断器へのトリップ信号を出力する回路の主リレーは、トリップ信号を出力する状態からしばらくして元の状態に復帰する。前述の仮定の下に「遮断器が開で主リレーがトリップ信号を出力していない状態」という観測兆候を与えると、遮断器の故障しか導かれない。しかし、実は遮断器も主リレーも正常で、観測時点より少し前の時点で主リレーへの作動信号を出力した補助リレーの動作回路に故障が発生していたかもしれない。

以上のように故障状態は変化しないという仮定によって、観測兆候に対して考えられる故障原因候補が網羅できないことになる。ここでは、観測部分から未観測部分の振る舞いを推定するということを、以下の二つに明確に区別し、このことから故障発生後の対象の状態変化をとらえ、その結果上記の問題点に対処することを提案する。

- (i) ある時点での対象を構成する機器に関して、既知と未知の状態が存在する場合に、機器間の構造や因果関係をもとにして未知の状態を推定する。
- (ii) ある時点での対象を構成する機器に関して、既知の状態の機器の中に状態遷移の可能性がある場合に、該当する機器に関する状態遷移処理によって少し前の時点における状態を求め、さらに(i)の処理によって未知の機器の状態を推定する。

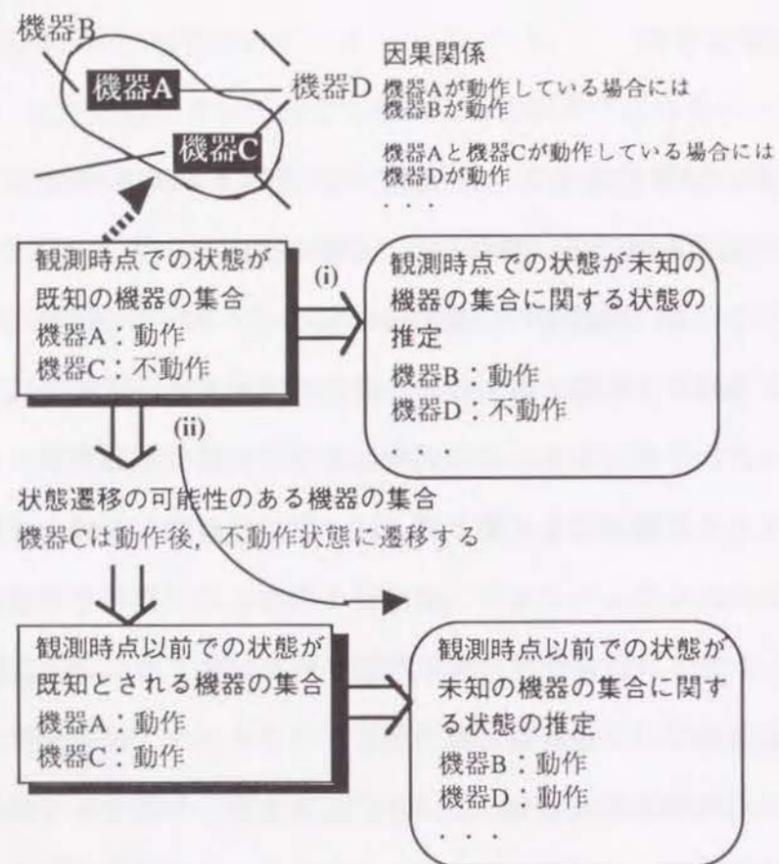


図6.1 観測兆候から未観測部分の振る舞いの推定処理

図6.1に示されるように(i)は、ある時点での対象の状態をスナップショット的に切り出したとみなすことができ、(ii)は状態遷移によって時間軸上で可能性のあるこのスナップショットの状態を導出する処理だとみなすことができる。前述の電力保護リレーの例では、(i)の処理によって観測時点の対象の状態、すなわち「遮断器が開、主リレーや補助リレーは動作していない」状態が導かれる。(ii)の処理によって、先ず「復帰する可能性のある主リレーは、少し前の時点においてトリップ信号を出力する動作状態にあった」ということが推定され、そのことからその他の機器の動作が導かれる。このように(i)、(ii)を組み合わせると観測時点ならびにそれ以前の時点の対象の状態を導いた後に、故障していると考えられる機器を同定することによって、故障発生後の対象の状態変化をとら

えた故障木を作成する。

### 6.2.2 対象のモデルおよび推論処理

図6.2は、本章で提案するモデルの生成から故障木を作成するまでの手法の概要を示している。図6.2中に示される流れにそって、以下の三点を説明する。

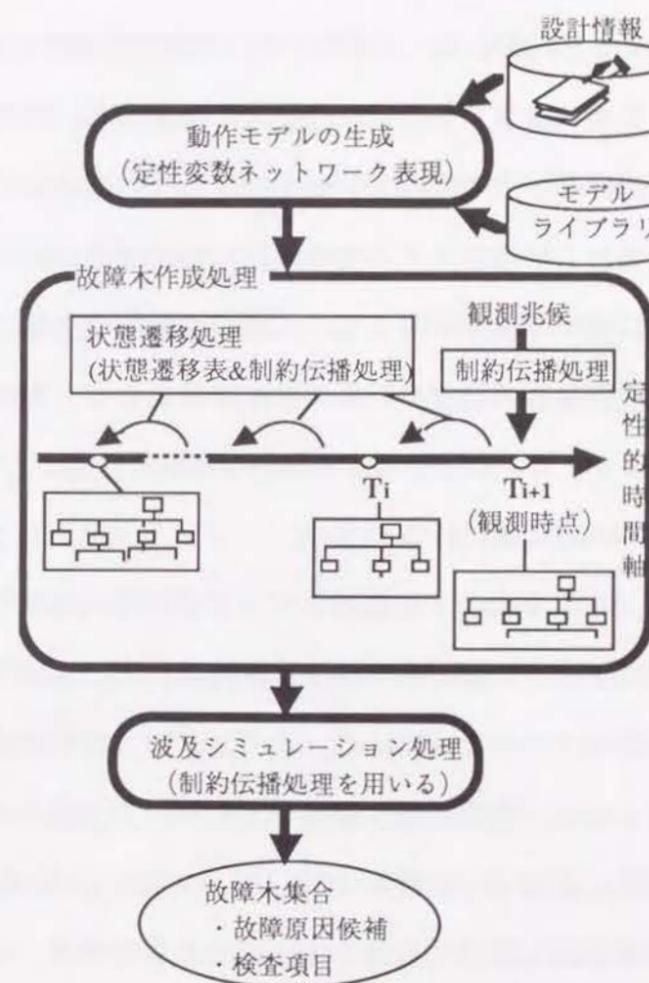


図6.2 故障木作成手法の概要

- (1) 設計情報に基づくモデルの表現およびその生成
- (2) 観測兆候からの故障原因候補の導出
- (3) 故障の波及シミュレーションによる故障原因候補の絞り込み

第一の“モデルの表現およびその生成”に関しては、あらかじめ整理した機器モデルとそれらの接続情報から生成する方式を用いる。具体的には、機器単位の動作状態を、機器モデルに準備した状態変数の値の組み合わせによって表現する。この結果、機器モデル内部の変数間に成立する関係は、対象に関する因果関係の一部ととらえられる。一方、機器間の因果関係に関しては、接続情報をもとに動作モデルを生成する際に、機器間の値の伝播にかかわる変数の属性情報（電圧や温度など対象領域により異なる）を機器モデルに記述しておき、同じ属性を持つ変数間に成立する関係は自動的に作成する。異なる属性を持つ変数間に成立する関係は、その組み合わせを指定して作成する必要がある。これら機器モデル内部および機器モデル間の変数に指定される因果関係を、以後制約と呼ぶ。対象に関する機能に関しては、兆候が機能の欠落などで観測される場合に具体的な機器集合の状態へ対応づけて診断を行なうために必要であるが、ここでは兆候は少なくとも機器単位で与えられると仮定して取り扱うので、モデルには機能レベルの記述は行なわない。

以上のことから、対象を構成する機器モデルとそれらの接続情報をもとに生成した全体モデルによって対象の動作状態は表現される。具体的には状態変数と制約によるネットワークとなっている。このときに、状態変数として実際の値を示した連続量よりは、機器の動作閾値などを用いて、例えば+は閾値以上のように状態の特徴を表現する定性量を用いる。このことは、故障木作成を対象のこのような状態と因果関係に着目して行なうからであり、ネットワークのノードとアークを定性変数と定性的制約から構成する。

第二の“観測兆候からの故障原因候補の導出”に関しては、定性変数と定性的制約によるネットワーク上の一部の変数に値を割り当てた後に、未定の変数の値を制約を用いて求めることに対応している。このことは、6.2.1に述べた(i)の処理に対応する。以後、これを制約伝播処理と呼ぶ。また、6.2.1に述べた(ii)の処理は、状態遷移の可能性のある機器に対応するネットワーク上の定性変数

に関して状態遷移表を準備し、現時点の値からある時間前の遷移可能な値を全て生成した後に、制約を使って一貫性のある値の組み合わせを導出するという処理で実現する。以後、これを状態遷移処理と呼ぶ。このようにして推定した機器の状態のAND/OR表現で故障木が作成される。故障木のルート・ノードは兆候が与えられた機器に、リーフはその時点での故障原因候補にそれぞれ対応し、途中のノードはそれらの間に関与している機器とその状態を示している。(ii)の処理によって導かれた故障木中の機器において、いずれかの状態が状態遷移処理前の状態と異なる場合には、図6.2中に示されるようにある時間前の時点の故障木として時間軸上に付け加えられる。新たな故障木が生成されなくなるまで、このような状態遷移処理を実行する。

このような定性変数と定性的制約に基づく制約伝播処理と状態遷移処理は、例えば図6.3に示すような流れで処理され、定性推論<sup>[30]</sup>ではこれらの効率よい処理方法が示されており、本手法の実装に用いている。

第三に“故障の波及シミュレーションによる故障原因候補の絞り込み”に関しては、故障木のそれぞれのリーフに記述された機器の状態から波及シミュレーションを実行することにより、観測された兆候以外に発生するはずの兆候を求めて、検査用項目として管理する。すなわち、リーフから故障木をルート・ノードへとたどる過程で、発生するはずの兆候が発生していないという矛盾が発生すると、この矛盾を引き起こしたリーフを故障木から削除する。このような新たに発生した矛盾箇所を故障候補に加える考え方もあるが、図6.2の処理の中では観測兆候は正しいと考える。この結果、故障原因候補の絞り込みが行なわれる。処理としては、定性変数と定性的制約によるネットワーク上での制約伝播処理を用いることができる。

以上の処理によって求められた図6.2中に示される故障木の集合によって、観測兆候は説明づけられる。

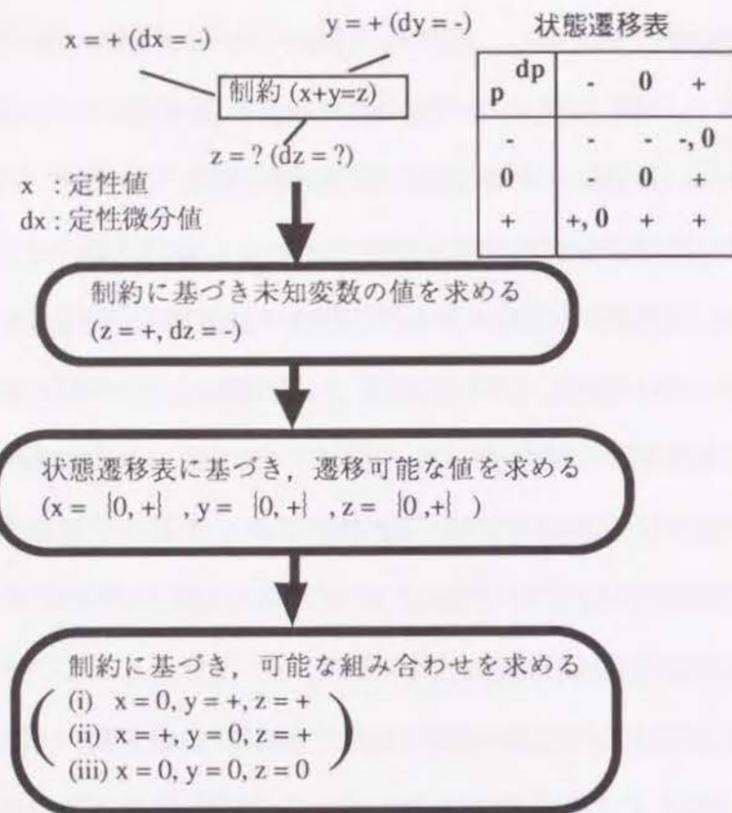


図6.3 制約伝播処理と状態遷移処理の例

### 6.2.3 モデルにおける故障の定義およびその処理

6.2.2で述べた故障木作成処理の中で、故障モデルとして導入した点について説明する。制約伝播処理によって推論された機器モデルの状態の中には、正常動作モードとも故障動作モードとも解釈可能なものがある。例えば、電磁リレーなどでリレー接点が閉じた状態は、リレーコイルが励磁して正常動作した場合と何らかの要因で短絡した場合が考えられる。正常動作した場合にはリレーコイルを励磁する原因を推論する必要があるが、故障動作した場合はこの電磁リレーが故障原因候補に挙げられる。すなわち、あらかじめ故障動作モードのわかっている機器に関しては、機器モデルに故障状態として記述し、制約伝播処理によって状態が求められたときに、その中で故障動作モードにあると解釈可能な機器は、その段階で故障原因候補に加える処理を実行する。

### 6.2.4 ソフトウェア構成

図6.4は、6.2.2で述べた処理を実行するシステムのソフトウェア構成である。本システムはLispとFlavorsを用いて実装しており、それぞれのモジュールは次のようになっている。

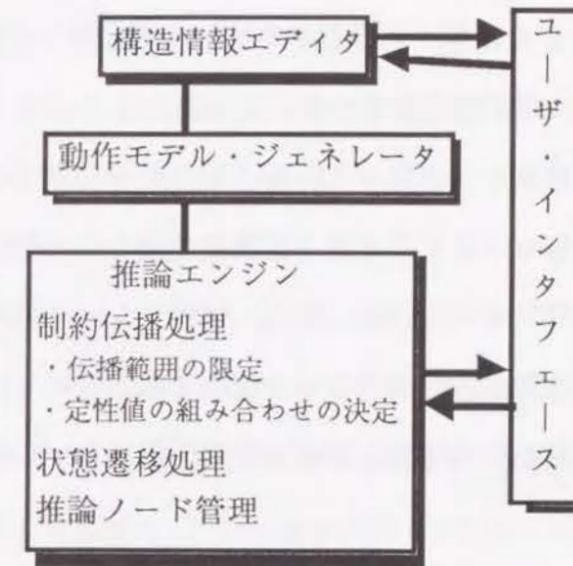


図6.4 ソフトウェア構成

構造情報エディタは、設計情報に基づいてユーザーが画面上で構成機器の作成、移動、接続、削除などを実行するためのものである。動作モデル・ジェネレータは、対象を構成している機器のモデル・ライブラリと構造情報エディタによる機器の接続状態から、推論エンジンのための実行形式のデータを作成する。推論エンジンは、制約伝播処理と状態遷移処理を用いて兆候に対応する故障原因候補を求め、その推論結果をもとに故障木のノードとして管理する。また、制約伝播処理を用いた故障の波及シミュレーションからそれぞれの故障原因候補に関する検査用項目も求める。ユーザー・インタフェースは、推論エンジンの実行を制御するモード設定や実行状況/結果を表示する。なお、制約伝播処理においては、兆候の与えられた機器に因果的に関係する機器に関してのみ処

理を実行すればよく、図6.4に示すように伝播範囲の限定という処理を組み込んでいる。

### 6.3 電力保護リレーへの適用

図6.5に示されるような変電所に設置されている保護リレーは、系統の異常を検知した情報をもとに停電範囲を最小限にいくとめるために、どの遮断器をトリップさせるかを定めるようになっている。したがって、この装置の誤動作、不動作はあってはならないことであり、信頼度の向上のために常時監視方式や自動点検回路などが開発されている。また、保護リレーの誤動作/不動作がひとたび発生すれば、故障木解析を行ない、故障を起こしにくくするような対策がとられる。以下、本手法の故障木解析を母線保護リレーに適用する。

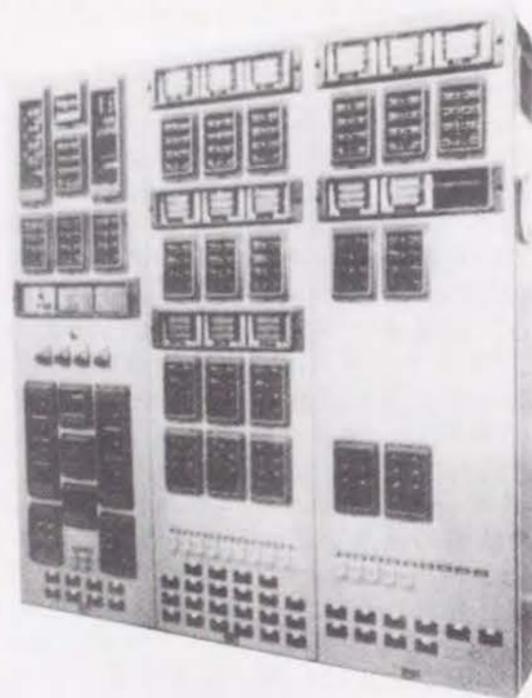


図6.5 母線保護リレーの盤面の例

### 6.3.1 保護リレーの設計情報およびモデルの生成

母線保護リレーはいくつかの回路で構成されており、対応する展開接続図が設計情報として利用できる。図6.6に示すのは、ある母線保護装置（アナログ・リレー）の展開接続図の構成ならびに展開接続図の一例である。この展開接続図上に示される機器を整理して、モデル・ライブラリとして記述した一部が表6.1であり、機器モデルには状態変数名とその定性変数軸とその物理属性、正常状態、定義可能な故障状態、入出力ポートが記述してある。なお、表中、端子の場合0は電流値無し、+は電流値ありを意味し、リレー接点の場合0はbreak状態、+はmake状態を意味している。定性値の組み合わせとして記述してある正常状態は、機器内部の変数間に成立する定性的制約ともなる。例えば、シーケンス回路中のリレーの場合には出力側端子の値が+の場合には、制約伝播処理によって入力側端子の値は+、リレー接点の値も+であると求まる。

表6.1 モデルライブラリーの例

	変数名	変数軸	物理属性	正常状態	故障状態	入/出力ポート
リレー 	入力側端子	(0+)	電流	(000)	(*0*)	入力側端子 出力側端子
	リレー接点	(0+)	位置	(0+0)		
	出力側端子	(0+)	電流	(+++) (+00)		
抵抗 	入力側端子	(0+)	電流	(00)	(*0)	入力側端子 出力側端子
	出力側端子	(0+)	電流	(++)		

表中の正常状態ならびに故障状態として、リレーの場合は（入力側端子 リレー接点 出力側端子）、抵抗の場合は（入力側端子 出力側端子）で表現している。\*はあらゆる定性値。また、故障状態として、リレーの場合には接点の固着及び短絡を、抵抗の場合には断線を定義する。

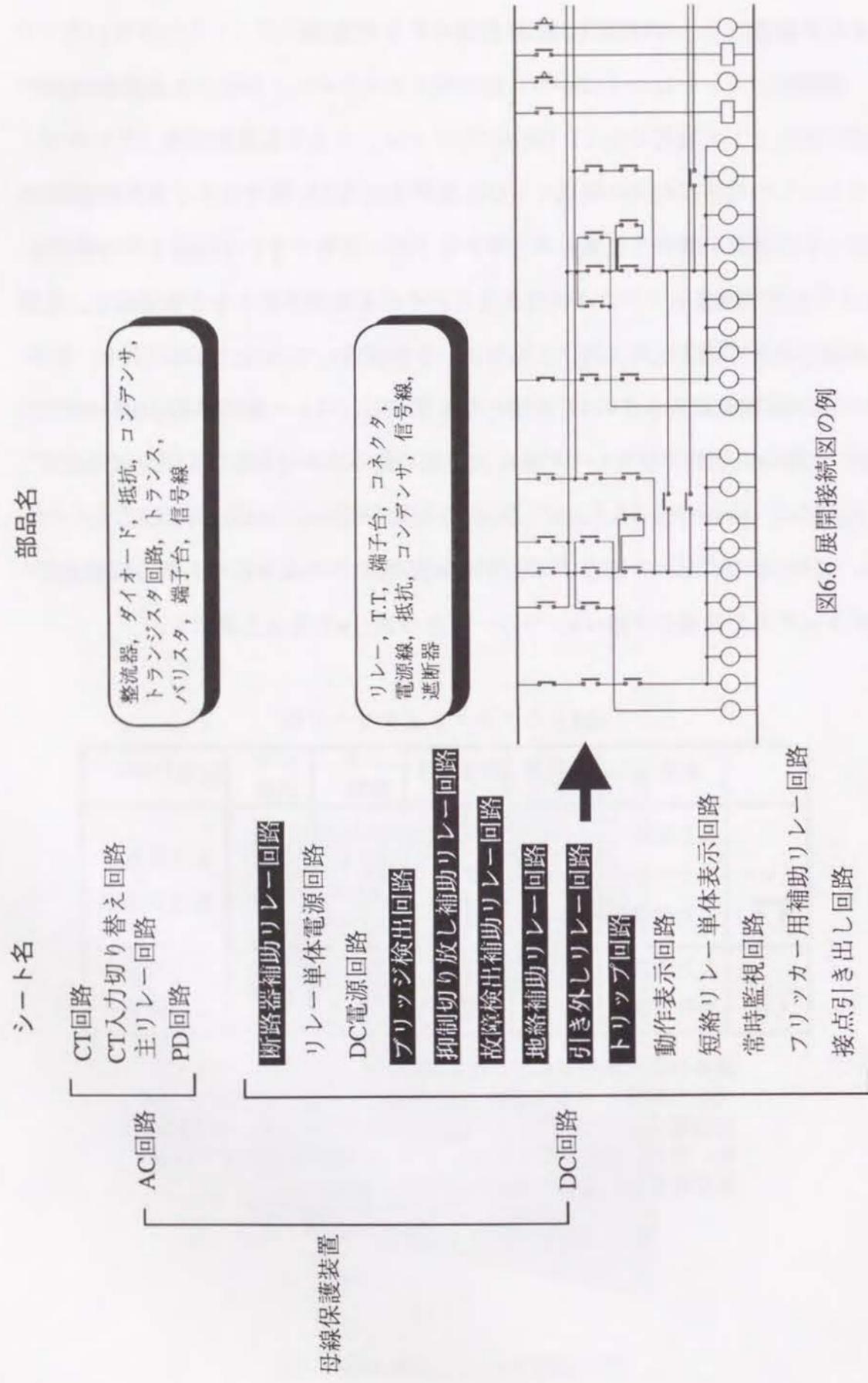


図6.6 展開接続図の例

また、展開接続図そのものの表現のためには、図面の一枚に相当するシートならびにシート上の部品に対応するデバイスという構造型データを準備し、モデル生成時にはこのほかに定性変数と定性的制約を表現する構造型データともあわせて、推論用に実行形式のシートオブジェクト、機器オブジェクト、変数オブジェクト、制約オブジェクトを生成する。さらに、保護リレーの展開接続図には、シート上で直接の接続関係がないにもかかわらず制御シーケンスを表現した部分があるために、対象に特有な部分として以下の処理を付加している。

- ・展開接続図上では制御シーケンスを正確に表現するために、一つの機器が図面上では複数記述されている場合があり、それらの機器が同一である、すなわち変数オブジェクトの定性値は常に等しいという制約オブジェクトを作成する。
- ・動作表現上、リレーとそのリレーコイルが分離して記述されているので、展開接続図上では直接の接続がないにもかかわらず制約オブジェクトを作成する。

図6.7は、以上のモデル・ライブラリと機器の接続情報から動作モデルを生成する流れをまとめたものである。

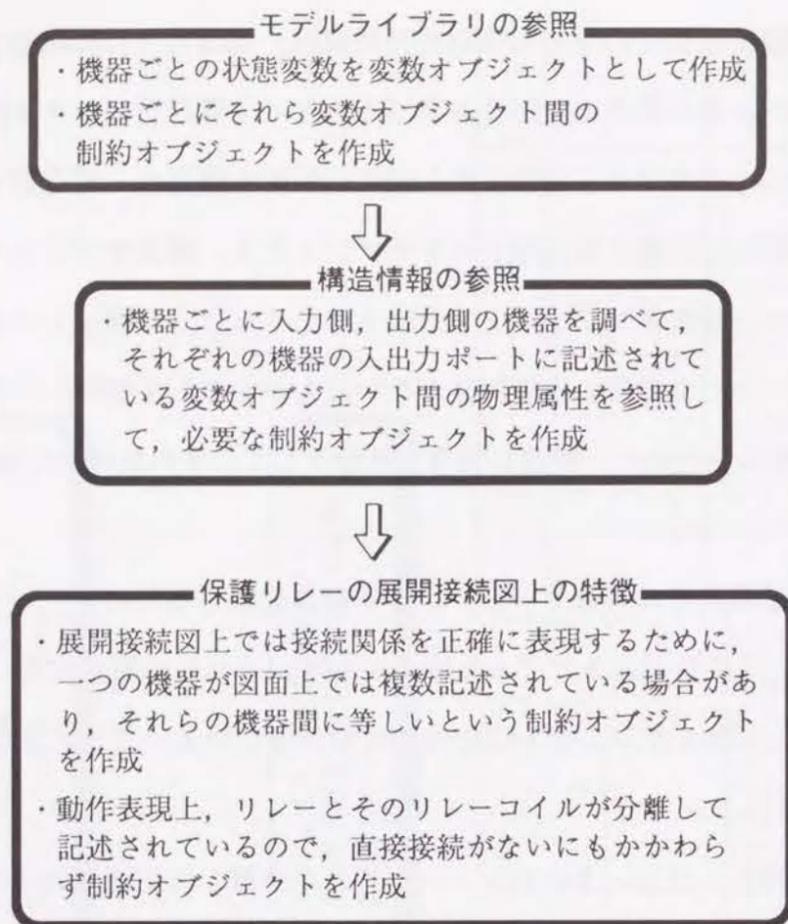


図6.7 モデル生成の流れ

### 6.3.2 制約伝播処理ならびに状態遷移処理

制約伝播処理の中で行なう伝播範囲の限定については，兆候の与えられた機器への信号経路を検出し，それに基づいて伝播処理に必要な定性変数ネットワークの対応する部分を求めている。具体的には，以下に示す手順で制約伝播処理から故障原因候補の導出を行なう。なお，故障原因候補の波及シミュレーションに対する制約伝播処理も，基本的な部分は同じであるので（実際ソフトウェア・モジュールは共通）， [] 内にその場合を示す。

- (1) 兆候の与えられた機器へ [から] のシート上での信号経路の検出。
- (2) 求めた信号経路の組合せから制約伝播を実行すべき機器を抽出し，それら

の機器に關与する変数オブジェクト，制約オブジェクトをもとに処理を行ない，値を割り当てる。

- (3) 信号経路のそれぞれの組合せに対して，機器ごとに記述されている正常状態及び故障状態をもとに故障候補 [動作チェック候補] を検出後，全ての機器が正常動作したという仮定のもとにさらなる原因 [影響] を調べるべき機器を選択して，(1)に戻る。このときに，調べるべき機器は基本的に別のシートに記述されているので，直接的因果関係を示す制約はあらかじめモデル生成の段階で作成されてなく，保護リレーの構成に関する知識から次のような制約に代わるルールを記述してある。すなわち，故障原因候補の導出の際には，リレーの動作状態が求まるとその原因となるリレーコイルを調べるべき機器として選択し，リレー接点がmake状態ならばリレーコイルは励磁しており，リレー接点がbreak状態ならばリレーコイルは励磁してないという具合に状態も決める。逆に，故障の波及シミュレーションの際には，リレーコイルの状態からリレーの動作状態を決める。

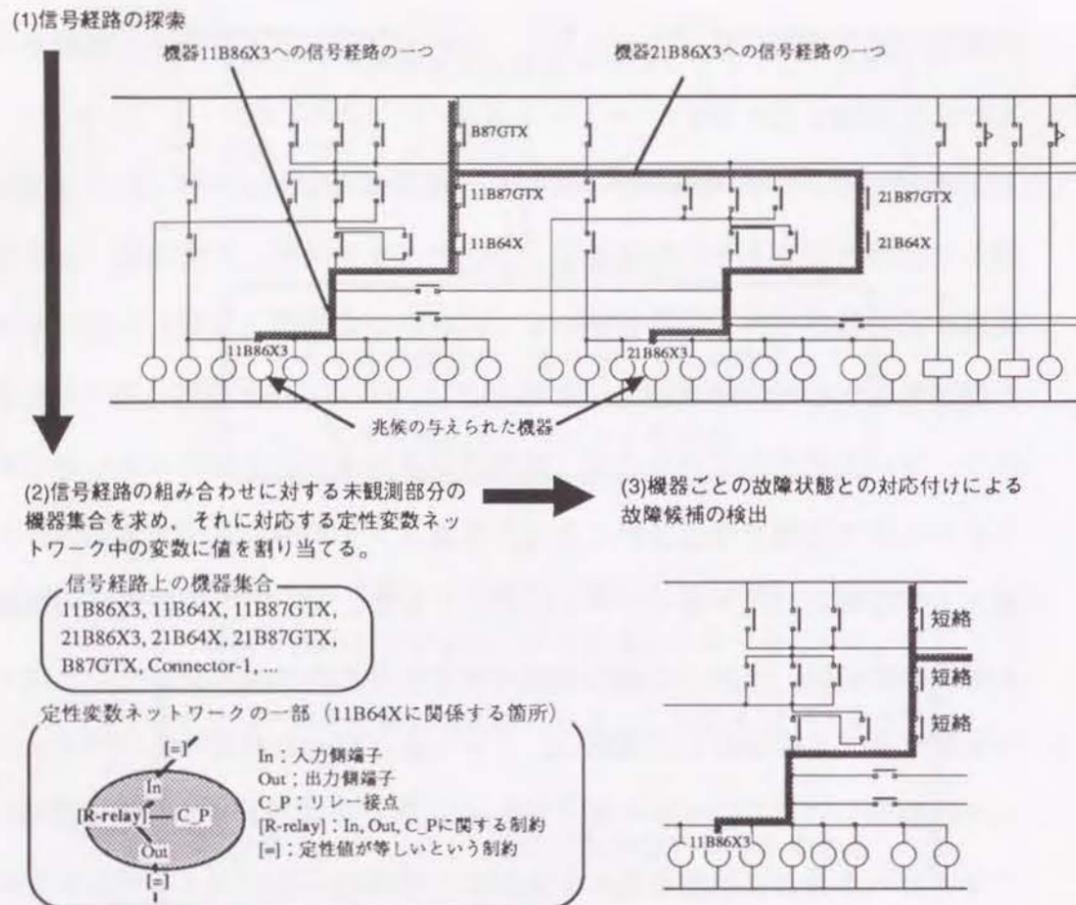


図6.8 制約伝播処理による故障候補の導出例

図6.8は、これらの処理の様子を展開接続図を用いて示したものであり、図中に示されるようにシートごとに故障原因候補が求まる。保護リレーに関する状態遷移処理としては、次の部分に考える必要がある。保護リレーには、一度トリップ信号を出力する条件が成立すると確実に信号を出すための自己保持回路と、逆にある一定時間後には遮断器を手動で復帰させるために今まで保持されていたトリップ信号の出力をとめる復帰回路が組み込まれている（タイマーが用いられている）。すなわち、この回路に参与するリレーの機器オブジェクトには、時点  $T_i$  でbreak状態（リレー接点の値が0）の時には時点  $T_{i+1}$  でmake状態（リレー接点の値が+）という状態遷移表を記述する。

### 6.3.3 動作結果

本システムに、ある遮断器がトリップしたという兆候（具体的には、遮断器は開状態、トリップ回路の主リレーのリレー接点はbreak状態）を与えて、故障木作成を試みた。モデルに含まれるシート数は21（図6.6中の白ヌキの回路部分）、機器の数は2292であり、この動作結果を以下に説明する。

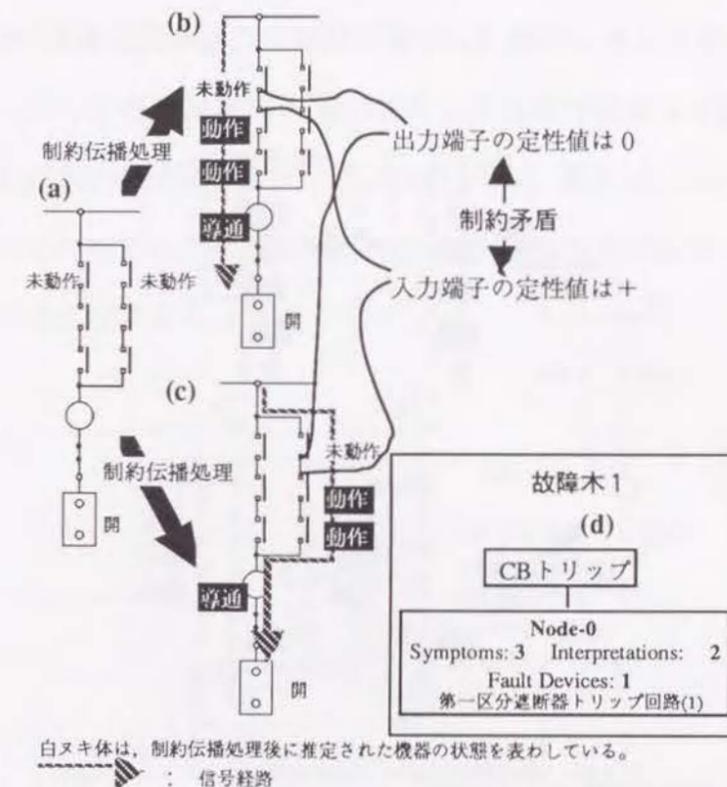


図6.9 観測時点での故障木

故障木1：観測時点においては、図6.9に示すように展開接続図上のそれぞれの信号経路に現われる機器の状態を制約伝播処理によって求める過程で、矛盾が発生して終了する。このことは、本来遮断器のトリップはトリップ回路のリレーが動作して信号経路を構成するはずなのに、現在観測兆候の与えられたリレーの状態から、そのような信号経路は構成されないことを示している。この

場合には、(観測された兆候と遮断器の故障動作モードから) 遮断器の故障のみが、故障木のリーフとして現われる。

ただし、故障木を機器間のAND/ORによる因果関係のままに表示すると複雑になり過ぎるので、図6.9-(d)のようにシートごとにまとめて故障木のノードとして表示する。故障木のノードの示す内容は、ノード番号、与えられた兆候の数、信号経路の組み合わせに基づく解釈の数、故障原因候補の数、シート名である。ノードに蓄えられた情報(どの信号経路上に故障原因候補があるかなど)は、展開接続図上で表示する。

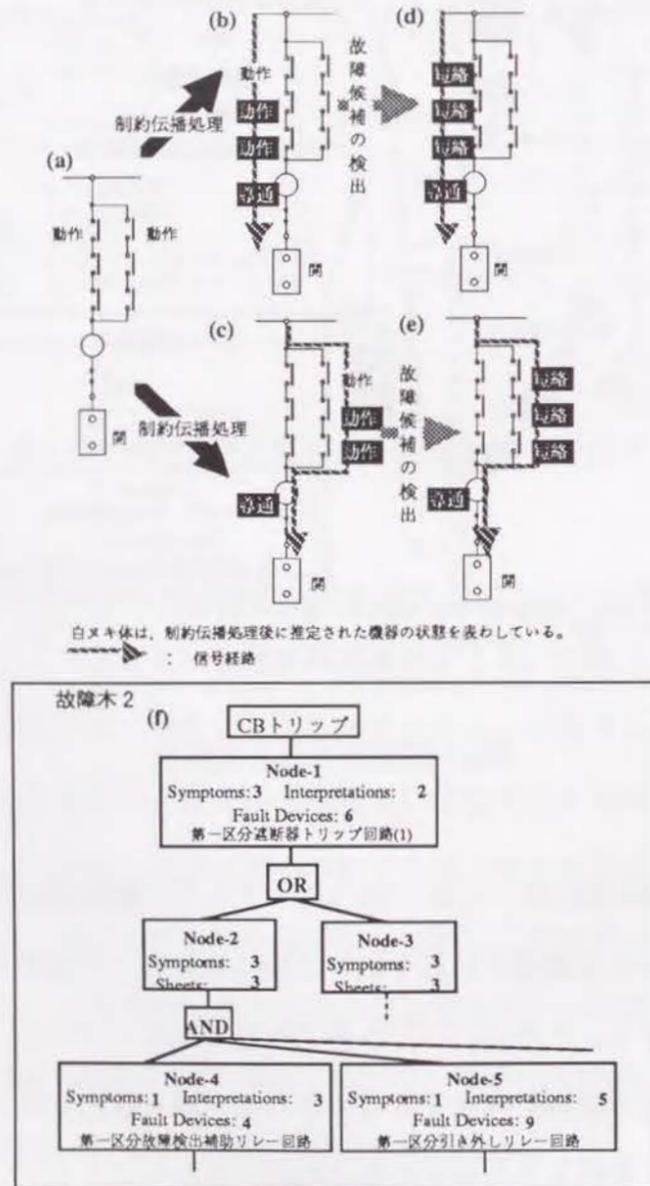


図6.10 観測時点以前の故障木

故障木2: 観測兆候の与えられた主リレーは、状態遷移表を持つことから状態遷移処理を実行する。この結果、制約伝播処理によって図6.10-(b), 図6.10-(c)に示される状態が求まり、さらに図6.10-(d), 図6.10-(e)のような故障としての解釈もなされる。この場合は、トリップ回路のリレーが動作あるいは短絡して信号経路を構成したために、遮断器がトリップしたことを示している。この結果をもとに、図6.10-(f)が示すようにさらに故障原因候補を求めることができた。

次に、ノードごとに故障原因候補の波及シミュレーションを実行すると、観測されるべき兆候が検査用項目として求められる。例えば、図6.11に示されるように、故障原因候補以外は正常として故障の影響を信号経路に基づいて調べると、検査用項目が求まる。

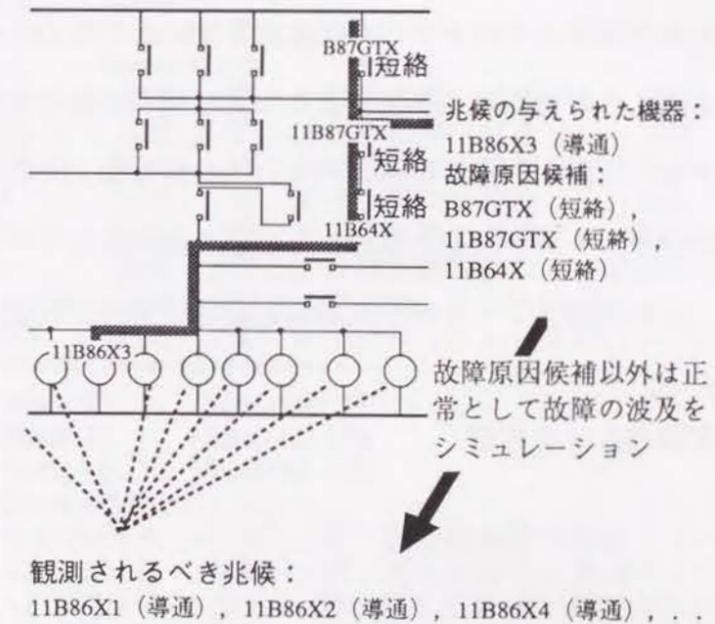


図6.11 故障の波及シミュレーションによる故障原因候補の絞り込み

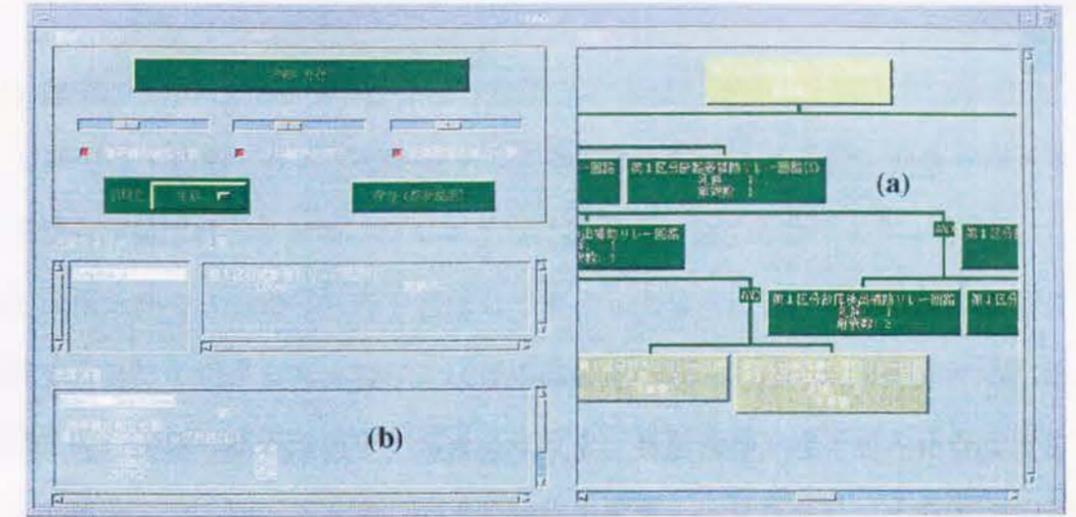
これらの観測されるべき兆候が発生していないことがテスト作業などで確認できれば、この故障原因候補は故障木から除くことができ、このような専門家への計算機援用の形態として、本手法を位置づけることができる。

以上の処理により、初期の観測兆候に対する故障原因候補として、故障木1および故障木2から109の機器を求めることができた。以上述べた故障木作成手法によって、規模の大きな対象に対する故障原因候補の導出ならびにテスト作業の際の支援の可能性が示された。

さらに、FlavorsとLispによるプロトタイプ・システムをもとにCとMotifを用いた故障木解析の支援システムを構築した。ここでは、図6.12-(a)のような故障木上でノードをマウスクリックすることにより、図6.12-(b)上に示されるようにそれぞれのノードに含まれる故障集合の情報を表示する機能に加えて、推論エンジンのFMEA実行時の機能として次の項目を付加した。

- ・故障木のリーフ間のAND/OR関係から故障原因集合を先ず作成し、この集合とともにテスト作業などで観測された事項を入力することにより、最終的な故障原因候補を出力する。

ノードに含まれる故障集合それぞれのAND/OR関係は木が深くなるにつれてわかりにくくなるが、この機能により次のような診断情報が得られる。ここでは故障検出補助リレー回路の一部が短絡したという条件が最初の故障木のノードには含まれていたが、テスト作業の結果から短絡も動作もしていないことが判明した場合に、この補助リレーを含む故障原因候補の集合がFMEAの結果から除かれた例である。



```

--- FTAに与えた兆候 ---
第1区分遮断器トリップ回路(1)      CB046      誤動作
--- FMEAに与えた兆候 ---
第1区分故障検出補助リレー回路      21B64      未動作
--- 診断結果 ---

```

```

*****
* Node-27 (State-158: 1/1)
* Node-26 (State-50: 1/1)
* Node-14 (State-39: 2/2)      故障集合
* Node-9 (State-29: 1/1)
* Node-16 (State-41: 1/2)
* Node-1 (State-3: 2/2)
>11B27H-A : 短絡 ...(第1区分故障検出補助リレー回路)
>11B27H-B : 短絡 ...(第1区分故障検出補助リレー回路)
>11B27H-C : 短絡 ...(第1区分故障検出補助リレー回路)
>11B64 : 短絡 ...(第1区分故障検出補助リレー回路)
>11B27L-A : 固着 ...(第1区分故障検出補助リレー回路)
>LS461 : 短絡 ...(第1区分断路器補助リレー回路(1))
>11B87GTX : 短絡 ...(第1区分引外しリレー回路)
>B87GTX : 短絡 ...(第1区分引外しリレー回路)
*****
Node-23 (State-156: 1/1)
Node-22 (State-46: 1/1)      除外された故障集合
Node-10 (State-30: 1/2)
Node-6(State-20: 1/1)
>21B27H-A : 短絡 ...(第1区分故障検出補助リレー回路)
>21B27H-B : 短絡 ...(第1区分故障検出補助リレー回路)
>21B27H-C : 短絡 ...(第1区分故障検出補助リレー回路)
>21B27L-C : 固着 ...(第1区分故障検出補助リレー回路)
>LS462 : 短絡 ...(第1区分断路器補助リレー回路(1))
*****

```

図6.12 解析用画面例と出力結果

#### 6.4 結言

本章では、観測された兆候に対する故障診断を行なうために、対象を構成する機器の接続情報とモデル・ライブラリに基づいて定性的動作モデルを作成し、制約伝播処理や状態遷移処理を利用して導出した故障原因候補を故障木の形で管理する方法を提案した。特に、従来の故障木作成で考慮されていなかった故障発生後の対象の状態変化をとらえることを、状態遷移処理を組み込むことによって解消した。適用例を通してその動作を確認し、結果としてえられた故障木によってテスト入力などの確認試験の作業における支援の可能性を示した。今後は、他の対象に関するモデル・ライブラリなどを整理して、適用範囲を広げていきたいと考える。

## 7 結論

本論文は、動的システムの因果理解という側面から、人間の思考に整合する機能を備えた計算機と人間とのインタフェース構築に取り組んだものである。要素技術として、定性推論を位置づけるとともに、認知科学や実問題への適用の両側面をふまえて構築手法を議論した。以下に、各章で得られた研究成果をまとめる。

第2章では、定性推論の基本的な表現を述べた後に、動的システムの因果性の取り扱いに関して、従来の定性推論における代表的な手法として、「架空の因果性を用いた挙動解析」と「構造方程式を用いた因果解析」について問題点を含めて説明した。「架空の因果性を用いた挙動解析」では、局所的な因果伝播という考え方がわれわれの因果理解に近いものでありながら、用いられるヒューリスティクスの十分性が検証できない。このことから、実問題への適用に際して、より頑健な因果伝播の手法が望まれる点を明らかにした。「構造方程式を用いた因果解析」は、方程式に含まれる変数の値が求められる過程をもとに、変数間の依存構造を決定するシステムティックな手法である。ただし、連立方程式の解は同時的に求められることから、同時的制約関係にある変数群の内部に依存構造をみいだすことはできない。したがって、常に依存構造からわれわれの因果理解に近い因果的説明を求めることはできない点を明らかにした。

第3章では、定量的解析が実行可能な場合において定性的解析を実行する意義として、現象を因果的に説明することに焦点をあて、計算機上での具体的な実現に向けて定性モデルの構築、定性シミュレーション手法や因果的説明の導出を定量的解析と関連づける手法を提案した。提案した手法の特徴は、以下のとおりである。

- (1) 定性モデルの構築に際しては、定量モデルから導かれた依存構造を利用することにより、モデル作成者の視点を反映する変数群に関して定性関係を定義すべき変数の対が求まる。
- (2) 定量データに対する定性的視点をもとに導いた定性的制約データと「解釈のための時間」を用いて、影響の伝播による定性シミュレーションに発生する曖昧さの解消を行なう。
- (3) 定量データとの対応づけを組み込んだ定性シミュレーションを、影響の伝播に基づいて実行することによって、現象に内在する変数間の相互の影響を明らかにし、そのシミュレーション過程に基づく因果的説明を導出する。

第4章では、第3章の手法を発電プラントへ適用し、定量モデルとしての方程式から依存構造を求め、次に運転員が着目する25個の変数群に関して縮約した依存構造を導いた。この結果作成された定性モデルから、定量データとの対応づけにより定性シミュレーションだけでは解消できない曖昧さの発生を解消し、因果的な説明を導出することができた。定性モデルの構築に際しては、求めた変数の対が直接的影響を表していると考えられない場合もあり、その際にはモデル作成者が、計算機から導出された依存構造をもとに、さらに依存リンクを除く必要もある。

第5章では、因果理解という認知プロセスをメンタルモデルの形成という立場から考察し、因果理解支援方式を提案した。具体的には、階層型定性モデルとそれを用いた定性シミュレーションをもとにして、学習者とのインタラクションを構成する。階層型定性モデルは、学習者のメンタルモデルに基づく対象システムのとらえ方を階層的な機器をもとに構成したものであり、定性シミュレーションは局所的な影響伝播をもとに実行される。定性シミュレーションによって現象に対する因果的な説明を導出する過程で、相反する影響の伝播などの因

果的に解釈が複雑な時点を学習者に認識させるとともに、納得のいくまで「表象-吟味のループ」を利用するツールとなっている。

第6章では、事象間の因果理解が前提となっている故障木解析について議論した。故障時点と観測時点が異なり、その間に故障状態が変化する対象では、専門家といえどもそのような故障木解析は困難である。この問題に対して、対象を構成する機器の接続情報とモデル・ライブラリに基づいて定性的動作モデルを作成し、観測された兆候から制約伝播処理や状態遷移処理を利用して故障原因候補を導出する手法を提案した。適用例を通してその動作を確認し、結果としてえられた故障木によってテスト入力などの確認試験の作業における支援の可能性を示した。導出した故障原因候補を故障木の形で管理することによって、テストなどの検証作業に関する専門家への支援についても述べた。

今後の研究課題としては、定量的解析との対応操作の中で定性モデルの修正をより強力に支援する機能の実現を通して、より頑健な定性シミュレーション手法の開発が重要と考える。また、説明の一般的な手段として「因果的説明」と「目的論的説明」が議論されるが<sup>[7]</sup>、これら二つの関連づけを考察し、教育・訓練システムの要素技術としてツール化することや、故障木解析としての適用範囲を広げることも重要な課題である。

## 謝辞

本研究は、筆者が三菱電機（株）中央研究所で行なった研究を、京都大学工学研究科応用システム科学専攻布川昊教授のご指導のもとにまとめたものである。終始ご懇切なご指導とご助言を賜りました布川昊教授に深謝いたします。また、本論文の内容につきご指導を賜った、同大学工学研究科応用システム科学専攻足立紀彦教授、同大学工学研究科精密工学専攻片井修教授にお礼を申し上げます。

三菱電機（株）中央研究所システム基礎研究部元部長武田捷一博士には、良好なる研究環境を与えていただくとともに、本研究を遂行する上で多くのご理解をいただいた。同研究部元第4グループマネージャ西田正吾博士（現大阪大学基礎工学部教授）には、入社以来研究のご指導をいただくとともに、日頃から多くのご助言とご激励をいただいた。また、三菱電機（株）先端技術総合研究所システム情報研究部部長渡邊治氏には、本研究をまとめるにあたり多くのご理解をいただいた。同研究部第3プロジェクトチームリーダー瀬尾和男博士には、本論文への細やかなご助言をいただいた。同研究部の先輩、同僚には、本研究を遂行する上でご協力をいただいた。また、同社制御製作所電力系統部次長合田忠弘氏、電力系統部電力系統品管課塩谷全啓氏には、電力保護リレーに関して有益なご教示をいただいた。

以上の方々に、衷心より感謝の意を表します。

最後に、学びの道において幼少の頃より常に暖かく励ましてくれた両親敏夫、ミサエ、妻として研究生生活を心身両面から支えてくれた恵に心から感謝いたします。

## 参考文献

- [1]Addanki, S., Cremonini, R. and Penberthy, J. S., "Graphs of Models", *Artificial Intelligence*, Vol.51, No.1-3, pp.145-177 (1991)
- [2]Akiyoshi, M. and Nishida, S., "A Computer-based Supporting Environment for Causal Understanding of Large Scale Systems by Hierarchical Qualitative Simulation", *Proc. of 1990 IEEE Conference on Systems, Man and Cybernetics*, pp.823-825 (1990)
- [3]Akiyoshi, M. and Nishida, S., "A Qualitative Simulation-based Learning Environment: How to Enhance Causal Understanding of Complex Phenomena in Large-scale Plants", *Proc. of IEEE Phoenix Conference on Computer and Communications*, pp.531-537 (1993)
- [4]Akiyoshi, M. and Nishida, S., "Towards Generating Causal Explanation: Qualitative Simulation with Association Mechanism to Quantitative Information", *IEEE Trans. on Industrial Electronics* (to appear in April, 1996)
- [5]Amador, F. G, Finkelstein, A. and Weld, D. S., "Real-time Self-explanatory Simulation", *Proc. of the 12th National Conference on Artificial Intelligence*, pp.562-567 (1993)
- [6]Crawford, J., Farquhar, A. and Kuipers, B. J., "QPC: A Compiler from Physical Models into Qualitative Differential Equations", *Proc. of the 9th National Conference on Artificial Intelligence*, pp.365-372 (1990)
- [7]de Kleer, J., "Multiple Representations of Knowledge in a Mechanics Problem Solver", *Proc. of the 5th International Joint Conference on Artificial Intelligence*, pp.299-304 (1979)
- [8]de Kleer, J., "The Origin and Resolution of Ambiguities in Causal Arguments", *Proc. of the 6th International Joint Conference on Artificial Intelligence*, pp.197-203 (1979)
- [9]de Kleer, J., "How Circuits Work", *Artificial Intelligence*, Vol.24, No.1-3, pp.205-280 (1984)
- [10]de Kleer, J., "Diagnosing Multiple Faults", *Artificial Intelligence*, Vol.32, No.11, pp.97-130 (1987)
- [11]de Kleer, J. and Bobrow, D. G., "Qualitative Reasoning with Higher-Order Derivatives", *Proc. of the third National Conference on Artificial Intelligence*, pp.86-91 (1984)
- [12]de Kleer, J. and Brown, J. S., "A Qualitative Physics based on Confluences", *Artificial Intelligence*, Vol.24, No.1-3, pp.7-83 (1984)
- [13]de Kleer, J. and Williams, B. C., "Diagnosis with Behavioral Modes", *Proc. of the 11th International Joint Conference on Artificial Intelligence*, pp.1324-1330 (1989)
- [14]DeCoste, D., "Dynamic Across-time Measurement Interpretation", *Proc. of the 9th National Conference on Artificial Intelligence*, pp.373-379 (1990)
- [15]Falkenhainer, B. and Forbus, K. D., "Setting up Large-scale Qualitative Models", *Proc. of the 7th National Conference on Artificial Intelligence*, pp.301-306 (1988)
- [16]Falkenhainer, B and Forbus, K. D., "Compositional Modeling :Finding the Right Model for the Job", *Artificial Intelligence*, Vol.51, No.1-3, pp.95-143 (1991)
- [17]Forbus, K. D., "Qualitative Process Theory", *Artificial Intelligence*, Vol.24, No.1-3, pp.85-168 (1984)
- [18]Forbus, K. D., "Interpreting Measurements of Physical Systems", *Proc. of the 5th National Conference on Artificial Intelligence*, pp.113-117 (1986)
- [19]Forbus, K. D. and Falkenhainer, B, "Self-explanatory Simulations : An Integration of Qualitative and Quantitative Knowledge", *Proc. of the 9th National*

- Conference on Artificial Intelligence*, pp.380-387 (1990)
- [20]Forbus, K. D. and Falkenhainer, B., "Self-explanatory Simulations: Scaling up to Large Models", *Proc. of the 11th National Conference on Artificial Intelligence*, pp.685-690 (1992)
- [21]Gautier, P. O. and Gruber, T. R., "Generating Explanations of Device Behavior using Compositional Modeling and Causal Ordering", *Proc. of the 12th National Conference on Artificial Intelligence*, pp.264-270 (1993)
- [22]Gentner, D. and Stevens, A. L. (eds.), *Mental Models*, Lawrence Erlbaum Associates, New Jersey (1983)
- [23]Hollan, J. D., Hutchins, E. L. and Weitzman, L., "STEAMER :An Interactive Inspectable Simulation-Based Training System", *The AI Magazine*, Summer 1984, pp.15-27 (1984)
- [24]Ishida, Y., Adachi, N. and Tokumaru H., "Some Results on the Qualitative Theory of Matrix", *Trans. of the Society of Instrument and Control Engineers*, Vol.17, No.1, pp.49-55 (1981)
- [25]Iwasaki, Y., *Model Based Reasoning of Device Behavior with Causal Ordering*, PhD thesis, Carnegie Mellon University (1988)
- [26]Iwasaki, Y., "Causal Ordering in a Mixed Structure", *Proc. of the 7th National Conference on Artificial Intelligence*, pp.313-318 (1988)
- [27]Iwasaki, Y., "On the Relationship between Model Abstraction and Causality: Variance of Causal Ordering under Abstraction Operations", *Proc. of PRICAI-90*, pp.396-401 (1990)
- [28]Iwasaki, Y. and Simon, H. A., "Causality in Device Behavior", *Artificial Intelligence*, Vol.29, No.1, pp.3-32 (1986)
- [29]Kuipers, B. J., "Commonsense Reasoning about Causality : Deriving Behavior from Structure", *Artificial Intelligence*, Vol.24, No.1-3, pp.169-204 (1984)
- [30]Kuipers, B. J., "Qualitative Simulation", *Artificial Intelligence*, Vol.29, No.3, pp.289-338 (1986)
- [31]Kuipers, B. J., "Abstraction by Time-scale in Qualitative Simulation", *Proc. of the 6th National Conference on Artificial Intelligence*, pp.621-625 (1987)
- [32]Kuipers, B. J., "Qualitative Simulation as Causal Explanation", *IEEE Trans. on Systems, Man, and Cybernetics*, Vol.17, No.3, pp.257-266 (1987)
- [33]Kuipers, B. J. and Berleant, D., "Using Incomplete Quantitative Knowledge in Qualitative Reasoning", *Proc. of the 7th National Conference on Artificial Intelligence*, pp.324-329 (1988)
- [34]Kuipers, B. J. and Chiu, C., "Taming Intractable Branching in Qualitative Simulation", *Proc. of the 6th International Joint Conference on Artificial Intelligence*, pp.1079-1085 (1987)
- [35]Lapp, S. A. and Powers, G. J., "Computer-Aided Synthesis of Fault-Trees", *IEEE Trans. on Reliability*, Vol.R-26, April, pp.2-13 (1977)
- [36]Lee, W. W. and Kuipers, B. J., "Non-Intersection of Trajectories in Qualitative Phase Space: A Global Constraint for Qualitative Simulation", *Proc. of the 7th National Conference on Artificial Intelligence*, pp.286-290 (1988)
- [37]Nishida, T. and Doshita, S., "A Geometric Approach to Total Envisioning", *Proc. of the 12th International Joint Conference on Artificial Intelligence*, pp.1150-1155 (1991)
- [38]Nishida, T., Mizutani, K., Kubota, A. and Doshita, S., "Automated Phase Portrait Analysis by Integrating Qualitative and Quantitative Analysis", *Proc. of the 10th National Conference on Artificial Intelligence*, pp.811-816 (1991)
- [39]Norman, D. A., "Some Observations on Mental Models", in Gentner, D. and Stevens, A. L. (eds.), *Mental Models*, pp. 7-14, Lawrence Erlbaum Associates, New Jersey (1983)

- [40]Norman, D. A. and Draper, S. W. (eds.), User Centered System Design - New Perspectives on Human-Computer Interaction", Lawrence Erlbaum Associates, New Jersey (1986)
- [41]Raiman, O., "Order of Magnitude Reasoning", *Proc. of the 5th National Conference on Artificial Intelligence*, pp.100-104 (1986)
- [42]Rasmussen, J., "Skills, Rules, and Knowledges: Signals, Signs, and Symbols, and Other Distinctions in Human Performance Models", *IEEE Trans. on Systems, Man, and Cybernetics*, Vol.SMC-13, No.3, pp.257-266 (1983)
- [43]Rieger, C. and Grinberg, M., "The Declarative Representation and Procedural Simulation of Causality in Physical Mechanisms", *Proc. of the 5th International Joint Conference on Artificial Intelligence*, pp.250-256 (1977)
- [44]Sacks, E., "A Dynamic Systems Perspective on Qualitative Simulation", *Artificial Intelligence*, Vol.42, No.2-3, pp.349-362 (1990)
- [45]Salem, S. L., Apostolakis, G. E. and Okrent, D., "A New Methodology for the Computer-Aided Construction of Fault Tree", *Annals of Nuclear Energy*, Vol.4, No. , pp.417-433 (1977)
- [46]Struss, P., "Global Filters for Qualitative Behaviors", *Proc. of the 7th National Conference on Artificial Intelligence*, pp.275-279(1988)
- [47]Struss, P. and Dressler, O., "Physical Negation - Integrating Fault Models into the General Diagnostic Engine", *Proc. of the 11th International Joint Conference on Artificial Intelligence*, 1318-1323 (1989)
- [48]Tayler, J. R., "An Algorithm for Fault-Tree Construction", *IEEE Trans. on Reliability*, Vol.R-31, No. 2, pp.137-146 (1982)
- [49]Wenger, E., *Artificial Intelligence and Tutoring Systems*, Morgan Kaufmann Publishers, California (1987)
- [50]White, B. Y. and Frederiksen J. R., "Intelligent Tutoring Systems Based upon Qualitative Model Evolutions", *Proc. of the 5th National Conference on Artificial Intelligence*, pp.313-319 (1986)
- [51]White, B. Y. and Frederiksen J. R., "Causal Model Progressions as a Foundation for Intelligent Learning Environments", *Artificial Intelligence*, Vol.42, No.1, pp.99-157 (1990)
- [52]Williams, B. C., "Qualitative Analysis of MOS Circuits", *Artificial Intelligence*, Vol.24, No.1-3, pp.281-346 (1984)
- [53]Yip, K. M.-K., "Understanding Complex Dynamics by Visual and Symbolic Reasoning", *Artificial Intelligence*, Vol.51, pp.179-221 (1991)
- [54]秋吉, 西田, "大規模システムの挙動理解のための階層型定性シミュレーション", *情報処理学会論文誌*, Vol.32, No.1, pp.32-41 (1991)
- [55]秋吉, 西田, "定量的解析との対応操作による定性シミュレーション", *人工知能学会誌* Vol.7, No.5, pp.837-849 (1992)
- [56]秋吉, 西田, 塩谷, 合田, "定性推論を利用した故障木作成手法とその保護リレーへの適用", *電気学会論文誌B分冊*, Vol.114, No.4, pp.410-418 (1994)
- [57]安西, 石崎, 大津, 波多野, 溝口 (編), *認知科学ハンドブック*, pp.65-77, 共立出版, 東京 (1992)
- [58]石田, "定性的推論を利用したプロセス診断知識獲得一定性的シミュレーションによる診断ルールの生成-", *電子情報通信学会誌*, Vol.J72-D-II, No.3, pp.416-426 (1989)
- [59]石田, 得丸, "ダイナミカルモデルにおける定性的シミュレーションアルゴリズムー因果論的アプローチー", *計測自動制御学会論文誌*, Vol.27, No.1, pp.101-106 (1991)
- [60]伊藤, 本位田, 沢村, 志田, "定性推論と定量推論を導入した待ち行列ネットワークのボトルネック診断と改善法", *人工知能学会誌*, Vol.5, No.1, pp.92-105 (1990)

- [61]岩政, 鈴木, 持地, "深い知識に基づく制御用エキスパートシステム—定性的因果モデルとクラスタリングによる診断機構の開発", 計測自動制御学会第13回知能システムシンポジウム, pp.1-7 (1991)
- [62]大木, 藤井, 古川, "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)
- [63]大和田, 溝口, 北沢, "定性的シミュレーションに基づく診断システムの構築法", 人工知能学会誌, Vol.3, No.5, pp.617-626 (1988)
- [64]佐伯 (編), 認知心理学講座3 推論と理解, 東京大学出版会, 東京 (1982)
- [65]佐伯 (編), 認知科学選書4 理解とは何か, 東京大学出版会, 東京 (1985)
- [66]佐山, 井上, "フォールト・ツリー解析—その基礎と応用", 計測自動制御学会誌, Vol.20, No.2, pp.32-41 (1981)
- [67]塩見, 故障解析と診断, 日科技連出版社, 東京 (1977)
- [68]駒井, 松本, 坂口, "モデルに基づく診断手法と電力系統事故診断への適用", 電気学会論文誌B分冊, Vol.110, No. 4, pp.258-266 (1990)
- [69]西田, "ヒューマン・インタフェースから見た教育・訓練システムの現状と今後の展望", 計測自動制御学会誌, Vol.31, No.12, pp.1211-1217 (1992)
- [70]西田, 川村, 堂下, "動的因果関係解析法による電子回路の定性的解析", 情報処理学会論文誌, Vol.28, No.2, pp.177-188 (1987)
- [71]西田, 佐伯, ヒューマン・コンピュータ交流技術, オーム社, 東京 (1991)
- [72]福井, 川上, "知識と対象の物理モデルに基づく電力系統故障区間の判定法", 電気学会論文誌C分冊, Vol.107, No. 2, pp.181-188 (1987)
- [73]G. H. フォンウリクト (丸山, 木岡 訳), 説明と理解 第2, 3章, 産業図書, 東京(1984)
- [74]溝口, "定性推論のためのモデル化支援システム", 人工知能学会誌, Vol.4, No.5, pp.547-555 (1989)
- [75]溝口, 古川, 安西 (共編), 定性推論 第2章, pp.11-76, 共立出版, 東京 (1989)
- [76]宮崎, 上野, 認知科学選書1 視点, 東京大学出版会, 東京 (1985)
- [77]武藤, 関根, "論理的制約関係を用いた電力設備の事故診断手法", 電気学会論文誌B分冊, Vol.112, No. 10, pp.927-936 (1992)
- [78]山口, 溝口, 角所, "深い知識に基づく故障木自動生成システム", 計測自動制御学会関西支部シンポジウム「あいまい情報処理と知的システム制御」論文集, pp.85-90 (1988)
- [79]山口, 溝口, 田岡, 小高, 野村, 角所, "深い知識に基づく知識コンパイラの基本設計", 人工知能学会誌, Vol.2, No.3, pp.333-340 (1987)
- [80]山口, 溝口, 中村, 小澤, 鳥越, 野村, 角所, "対象モデルと故障モデルに基づく知識コンパイラIIの構築と評価", 人工知能学会誌, Vol.7, No.4, pp.663-674 (1992)
- [81]吉田, 元田, "階層的定性推論のための浅い知識の合成法", 人工知能学会論文誌, Vol.4, No.4, pp.447-455 (1989)
- [82]鷺尾, "物理法則に基づく外的駆動型因果性の導出", 人工知能学会誌, Vol.5, No.4, pp.482-491 (1990)
- [83]渡辺, 渡部, 時間と人間, 中央公論社, 東京 (1979)

## 付録

### A 発電プラントに関する依存構造

プロセス変数の表記は、(変数名)もしくは(変数名 変化)で行なう。

表記は、

「変数群 (から影響) →

変数名 →

変数群 (に影響)」とする。

((総反応度)(炉心出力)(先行中性子濃度)) →

((炉心出力 変化)) →

((炉心出力))

((炉心出力)(先行中性子濃度)) →

((先行中性子濃度 変化)) →

((先行中性子濃度))

((炉心出力偏差)(遅発炉心出力偏差)) →

((遅発炉心出力偏差 変化)) →

((遅発炉心出力偏差))

((炉心内冷却水温度偏差測定値)(炉心内冷却水温度偏差)) →

((炉心内冷却水温度偏差測定値 変化)) →

((炉心内冷却水温度偏差測定値))

((燃料2温度偏差測定値)(燃料2温度偏差)) →

((燃料2温度偏差測定値 変化)) →

((燃料2温度偏差測定値))

((総炉心出力偏差)(燃料1温度偏差)(燃料2温度偏差)) →

((燃料1温度偏差 変化)) →

((燃料1温度偏差))

((総炉心出力偏差)(燃料1温度偏差)(燃料2温度偏差)(炉心内冷却水温度偏差)) →

((燃料2温度偏差 変化)) →

((燃料2温度偏差))

((総炉心出力偏差)(炉心内出口冷却水温度偏差)(燃料2温度偏差)(炉心内冷却水温度偏差)(炉心内入口冷却水温度偏差)) →

((炉心内冷却水温度偏差 変化)) →

((炉心内冷却水温度偏差))

((SG内出口水温度偏差)(SG内水温度偏差)(SG内入口水温度偏差)(二次冷却水温度偏差)) →

((SG内水温度偏差 変化)) →

((SG内水温度偏差))

((炉心内冷却水温度偏差)(冷却水1温度偏差)) →

((冷却水1温度偏差 変化)) →

((冷却水1温度偏差))

((冷却水2温度偏差)(炉心内入口冷却水温度偏差)) →

((冷却水2温度偏差 変化)) →

((冷却水2温度偏差))

((冷却水3温度偏差)(SG内入口水温度偏差)) →

((冷却水3温度偏差 変化)) →

((冷却水3温度偏差))

((SG内水温度偏差)(冷却水4温度偏差)) →

((冷却水4温度偏差 変化)) →

((冷却水4温度偏差))

((SG内入口水温度偏差)(容器出口温度偏差)) →

((SG内入口水温度偏差 変化)) →

((SG内入口水温度偏差))

((SG内出口水温度偏差)(容器入口温度偏差)) →

((容器入口温度偏差 変化)) →

((容器入口温度偏差))

((容器入口温度偏差)(炉心内入口冷却水温度偏差)) →

((炉心内入口冷却水温度偏差 変化)) →

((炉心内入口冷却水温度偏差))

((炉心内出口冷却水温度偏差)(冷却水2温度偏差)(容器出口温度偏差)) →

((容器出口温度偏差 変化)) →

((容器出口温度偏差))

((フィードウォーター流量偏差)(二次側流量偏差)) →

((二次冷却水質量偏差 変化)) →

((二次冷却水質量偏差))

((フィードウォーター流量偏差)(二次側流量偏差)(SG内水温度偏差)(二次冷却水温度偏差)) →

((二次側内部エネルギー偏差 変化)) →

((二次側内部エネルギー偏差))

((二次冷却水質量偏差)(二次側内部エネルギー偏差)(二次冷却水温度偏差)) →

((二次冷却水温度偏差 変化)) →

((二次冷却水温度偏差))

((比例ダンプ温度誤差信号)(フィルター後の比例ダンプ温度誤差信号)) →

((フィルター後の比例ダンプ温度誤差信号 変化)) →

((フィルター後の比例ダンプ温度誤差信号))

((二次側流量偏差)(二次側流量偏差測定値)) →

((二次側流量偏差測定値 変化)) →

((二次側流量偏差測定値))

((SG蒸気ダンプ弁流量)(蒸気ダンプ弁流量偏差)) →

((蒸気ダンプ弁流量偏差 変化)) →

((蒸気ダンプ弁流量偏差))

((フラッシング質量)(スプレー質量)(逃し弁による質量)) →

((加圧器内蒸気質量偏差 変化)) →

((加圧器内蒸気質量偏差))

((サージ質量 サージエンタルビー)(フラッシング質量)(スプレー質量)) →

((加圧器内冷却水質量偏差 変化)) →

((加圧器内冷却水質量偏差))

((サージ質量 サージエンタルビー)(フラッシング質量)(スプレー質量)(バックアップヒータ出力偏差)

) →

((加圧器内冷却水内部エネルギー偏差 変化)) →

((加圧器内冷却水内部エネルギー偏差))

((サージ質量 サージエンタルビー)(フラッシング質量)(スプレー質量)(加圧器内蒸気質量偏差)(加圧器

内冷却水容量偏差)(バックアップヒータ出力偏差)(逃し弁による質量)) →

((加圧器内蒸気容量偏差 変化)) →

((加圧器内蒸気容量偏差))

((サージ質量 サージエンタルビー)(フラッシング質量)(スプレー質量)(加圧器内蒸気質量偏差)(加圧器

内冷却水容量偏差)(バックアップヒータ出力偏差)(逃し弁による質量)) →

((加圧器圧力偏差 変化)) →

((加圧器圧力偏差))

((サージ質量 サージエンタルビー)(フラッシング質量)(スプレー質量)(バックアップヒータ出力偏差)

) →

((加圧器内冷却水容量偏差 変化)) →

((加圧器内冷却水容量偏差))

((バックアップヒータ出力偏差)(バックアップヒータ出力)) →

((バックアップヒータ出力偏差 変化)) →

((バックアップヒータ出力偏差))

((SG内水温度偏差)(冷却水3温度偏差)(冷却水4温度偏差)(SG内入口水温度偏差)(容器出口温度偏差))

→

((平均温度偏差 変化)) →

((平均温度偏差))

((SG内水温度偏差)(冷却水3温度偏差)(冷却水4温度偏差)(SG内入口水温度偏差)(容器出口温度偏差)(平

均温度偏差)(平均温度偏差補償値)) →

((平均温度偏差補償値 変化)) →

((平均温度偏差補償値))

((サージ質量 サージエンタルピー)(フラッシング質量)(スプレー質量)(加圧器内蒸気質量偏差)(加圧器  
圧力偏差)(加圧器内冷却水容量偏差)(バックアップヒータ出力偏差)

(加圧器圧力偏差補償値)(逃し弁による質量)) →

((加圧器圧力偏差補償値 変化)) →

((加圧器圧力偏差補償値))

((総反応度)(炉心出力)(先行中性子濃度)(炉心出力偏差補償値)) →

((炉心出力偏差補償値 変化)) →

((炉心出力偏差補償値))

(( $\delta$ )(制御棒速度)) →

((制御棒反応度 変化)) →

((制御棒反応度))

((SG内水温度偏差)(冷却水3温度偏差)(冷却水4温度偏差)(SG内入口水温度偏差)(容器出口温度偏差)(ス

テップ外乱SG内水温度偏差補償値)(ステップ外乱SG内水温度偏差)) →

((ステップ外乱SG内水温度偏差補償値 変化)) →

((ステップ外乱SG内水温度偏差補償値))

((SG内水温度偏差)(冷却水3温度偏差)(冷却水4温度偏差)(SG内入口水温度偏差)(容器出口温度偏差)(平  
均温度偏差)(平均温度偏差遅れ値)) →

((平均温度偏差遅れ値 変化)) →

((平均温度偏差遅れ値))

((SG内水温度偏差)(冷却水3温度偏差)(冷却水4温度偏差)(SG内入口水温度偏差)(容器出口温度偏差))

→

((ステップ外乱SG内水温度偏差 変化)) →

((ステップ外乱SG内水温度偏差))

((二次側流量偏差)(二次側圧力)(平均温度偏差)) →

(フィードウォーター流量偏差) →

((二次冷却水質量偏差 変化))((二次側内部エネルギー偏差 変化)))

((圧力流量偏差)(負荷流量偏差)(蒸気ダンプ弁流量偏差)(安全弁流量偏差)) →

(二次側流量偏差) →

((フィードウォーター流量偏差)(スクラム反応度)((二次冷却水質量偏差 変化))((二次側内部エネルギ  
ー偏差 変化))((二次側流量偏差測定値 変化)))

((飽和蒸気流量)) →

(圧力流量偏差) →

((二次側流量偏差))

((飽和蒸気容量)) →

(飽和蒸気流量) →

((圧力流量偏差))

((飽和水容量TO蒸気)(飽和水容量)) →

(飽和蒸気容量) →

((飽和蒸気流量))

((総炉心出力偏差)(フラッシング質量)(炉心内出口冷却水温度偏差)(SG内出口水温度偏差)(加圧器内冷  
却水エンタルピー偏差)(スプレー質量)(燃料2温度偏差)

(炉心内冷却水温度偏差)(SG内水温度偏差)(冷却水2温度偏差)(SG内入口水温度偏差)(容器入口温度偏差)  
(炉心内入口冷却水温度偏差)(容器出口温度偏差)

(加圧器内蒸気質量偏差)(加圧器内冷却水容量偏差)(バックアップヒータ出力偏差)(逃し弁による質量)  
) →

(サージ質量 サージエンタルピー) →

((加圧器内冷却水質量偏差 変化)((加圧器内冷却水内部エネルギー偏差 変化)((加圧器内蒸気容量  
偏差 変化)((加圧器圧力偏差 変化)((加圧器内冷却水容量偏差 変化))

((加圧器圧力偏差補償値 変化)))

((即時炉心出力偏差)(遅発炉心出力偏差)) →

(総炉心出力偏差) →

((サージ質量 サージエンタルピー)((燃料1温度偏差 変化)((燃料2温度偏差 変化)((炉心内冷却水温度  
偏差 変化)))

((二次側圧力) →

(飽和水容量TO蒸気) →

((飽和蒸気容量))

((誤差温度信号) →

(SG蒸気ダンプ弁流量) →

((蒸気ダンプ弁流量偏差 変化)))

((炉心出力偏差) →

(即時炉心出力偏差) →

((総炉心出力偏差))

((実効燃料温度偏差)(制御棒外乱反応度)(炉心内冷却水温度偏差)(加圧器圧力偏差)(制御棒反応度)(ス  
クラム反応度)) →

(総反応度) →

((炉心出力 変化)((炉心出力偏差補償値 変化)))

((二次冷却水温度) →

(飽和水容量) →

((飽和蒸気容量))

((二次冷却水温度) →

(二次側圧力) →

((フィードウォーター流量偏差)(飽和水容量TO蒸気))

((参照温度信号)(フィルター後の比例ダンプ温度誤差信号)) →

(誤差温度信号) →

((SG蒸気ダンプ弁流量))

((加圧器内冷却水エンタルピー偏差)(加圧器圧力偏差)) →

(フラッシング質量) →

((サージ質量 サージエンタルピー)((加圧器内蒸気質量偏差 変化)((加圧器内冷却水質量偏差 変化))

((加圧器内冷却水内部エネルギー偏差 変化))

((加圧器内蒸気容量偏差 変化)((加圧器圧力偏差 変化)((加圧器内冷却水容量偏差 変化)((加圧器圧力  
偏差補償値 変化)))

((参照温度信号)(平均温度偏差補償値)(加圧器圧力偏差補償値)(炉心出力偏差補償値)) →

(制御棒位置による誤差温度信号) →

(( $\delta$ ))

((参照温度信号)(平均温度偏差補償値)(加圧器圧力偏差補償値)(炉心出力偏差補償値)) →

(制御棒速度による誤差温度信号) →

((制御棒速度))

((炉心内出口冷却水温度偏差)(炉心内入口冷却水温度偏差)(加圧器圧力偏差)) →

(炉心内圧力トリップマージン) →

NIL

((炉心出力) →

(炉心出力偏差) →

((即時炉心出力偏差)((遅発炉心出力偏差 変化)))

((炉心内冷却水温度偏差測定値)(燃料2温度偏差測定値)(燃料1温度偏差)) →  
 (実効燃料温度偏差) →  
 ((総反応度))

((冷却水1温度偏差)(冷却水2温度偏差)) →  
 (炉心内出口冷却水温度偏差) →  
 ((サージ質量 サージエンタルピー)(炉心内圧力トリップマージン)((炉心内冷却水温度偏差 変化))(  
 (容器出口温度偏差 変化)))

((冷却水3温度偏差)(冷却水4温度偏差)) →  
 (SG内出口水温度偏差) →  
 ((サージ質量 サージエンタルピー)((SG内水温度偏差 変化))(容器入口温度偏差 変化)))

((二次冷却水温度偏差)) →  
 (二次冷却水温度) →  
 ((飽和水容量)(二次側圧力))

((SG内水温度偏差)(冷却水3温度偏差)(冷却水4温度偏差)(SG内入口水温度偏差)(容器出口温度偏差)(平  
 均温度偏差)) →  
 (比例ダンブ温度誤差信号) →  
 (((フィルター後の比例ダンブ温度誤差信号 変化)))

((二次側流量偏差測定値)) →  
 (参照温度信号) →  
 ((誤差温度信号)(制御棒位置による誤差温度信号)(制御棒速度による誤差温度信号))

((加圧器内冷却水質量偏差)(加圧器内冷却水内部エネルギー偏差)) →  
 (加圧器内冷却水エンタルピー偏差) →  
 ((サージ質量 サージエンタルピー)(フラッシング質量))

((加圧器圧力偏差)) →  
 (スプレー質量) →  
 ((サージ質量 サージエンタルピー)((加圧器内蒸気質量偏差 変化))(加圧器内冷却水質量偏差 変化))

((加圧器内冷却水内部エネルギー偏差 変化))  
 ((加圧器内蒸気容量偏差 変化))((加圧器圧力偏差 変化))((加圧器内冷却水容量偏差 変化))((加圧器圧力  
 偏差補償値 変化))

((加圧器圧力偏差)(ステップ外乱SG内水温度偏差補償値)(平均温度偏差遅れ値)) →  
 (SG内圧力トリップマージン) →  
 ((スクラム反応度))

NIL→  
 (制御棒外乱反応度) →  
 ((総反応度))

NIL→  
 (負荷流量偏差) →  
 ((二次側流量偏差))

((炉心出力 変化)) →  
 (炉心出力) →  
 ((炉心出力偏差)(スクラム反応度)((炉心出力 変化))((先行中性子濃度 変化))((炉心出力偏差補償値  
 変化)))

((先行中性子濃度 変化)) →  
 (先行中性子濃度) →  
 (((炉心出力 変化))((先行中性子濃度 変化))((炉心出力偏差補償値 変化)))

((遅発炉心出力偏差 変化)) →  
 (遅発炉心出力偏差) →  
 ((総炉心出力偏差)((遅発炉心出力偏差 変化)))

((炉心内冷却水温度偏差測定値 変化)) →  
 (炉心内冷却水温度偏差測定値) →  
 ((実効燃料温度偏差)((炉心内冷却水温度偏差測定値 変化)))

((燃料2温度偏差測定値 変化)) →

(燃料2温度偏差測定値) →

((実効燃料温度偏差)((燃料2温度偏差測定値 変化)))

((燃料1温度偏差 変化)) →

(燃料1温度偏差) →

((実効燃料温度偏差)((燃料1温度偏差 変化))((燃料2温度偏差 変化)))

((燃料2温度偏差 変化)) →

(燃料2温度偏差) →

((サージ質量 サージエンタルビー)((燃料2温度偏差測定値 変化))((燃料1温度偏差 変化))((燃料2温度偏差 変化))((炉心内冷却水温度偏差 変化)))

((炉心内冷却水温度偏差 変化)) →

(炉心内冷却水温度偏差) →

((サージ質量 サージエンタルビー)(総反応度)((炉心内冷却水温度偏差測定値 変化))((燃料2温度偏差 変化))((炉心内冷却水温度偏差 変化))((冷却水1温度偏差 変化))

((SG内水温度偏差 変化)) →

(SG内水温度偏差) →

((サージ質量 サージエンタルビー)(比例ダンプ温度誤差信号)((SG内水温度偏差 変化))((冷却水4温度偏差 変化))((二次側内部エネルギー偏差 変化))((平均温度偏差 変化))((平均温度偏差補償値 変化))((ステップ外乱SG内水温度偏差補償値 変化))((平均温度偏差遅れ値 変化))((ステップ外乱SG内水温度偏差 変化))

((冷却水1温度偏差 変化)) →

(冷却水1温度偏差) →

((炉心内出口冷却水温度偏差)((冷却水1温度偏差 変化)))

((冷却水2温度偏差 変化)) →

(冷却水2温度偏差) →

((サージ質量 サージエンタルビー)(炉心内出口冷却水温度偏差)((冷却水2温度偏差 変化))((容器出口温度偏差 変化)))

((冷却水3温度偏差 変化)) →

(冷却水3温度偏差) →

((SG内出口水温度偏差)(比例ダンプ温度誤差信号)((冷却水3温度偏差 変化))((平均温度偏差 変化))((平均温度偏差補償値 変化))((ステップ外乱SG内水温度偏差補償値 変化))((平均温度偏差遅れ値 変化))((ステップ外乱SG内水温度偏差 変化)))

((冷却水4温度偏差 変化)) →

(冷却水4温度偏差) →

((SG内出口水温度偏差)(比例ダンプ温度誤差信号)((冷却水4温度偏差 変化))((平均温度偏差 変化))((平均温度偏差補償値 変化))((ステップ外乱SG内水温度偏差補償値 変化))((平均温度偏差遅れ値 変化))((ステップ外乱SG内水温度偏差 変化)))

((SG内入口水温度偏差 変化)) →

(SG内入口水温度偏差) →

((サージ質量 サージエンタルビー)(比例ダンプ温度誤差信号)((SG内水温度偏差 変化))((冷却水3温度偏差 変化))((SG内入口水温度偏差 変化))((平均温度偏差 変化))((平均温度偏差補償値 変化))((ステップ外乱SG内水温度偏差補償値 変化))((平均温度偏差遅れ値 変化))((ステップ外乱SG内水温度偏差 変化))

((容器入口温度偏差 変化)) →

(容器入口温度偏差) →

((サージ質量 サージエンタルビー)((容器入口温度偏差 変化))((炉心内入口冷却水温度偏差 変化)))

((炉心内入口冷却水温度偏差 変化)) →

(炉心内入口冷却水温度偏差) →

((サージ質量 サージエンタルビー)(炉心内圧力トリップマージン)((炉心内冷却水温度偏差 変化))((冷却水2温度偏差 変化))((炉心内入口冷却水温度偏差 変化)))

((容器出口温度偏差 変化)) →

(容器出口温度偏差) →  
((サージ質量 サージエンタルピー)(比例ダンプ温度誤差信号)(SG内入口水温度偏差 変化))((容器出口温度偏差 変化))((平均温度偏差 変化))  
((平均温度偏差補償値 変化))((ステップ外乱SG内水温度偏差補償値 変化))((平均温度偏差遅れ値 変化))  
((ステップ外乱SG内水温度偏差 変化))

((二次冷却水質量偏差 変化)) →  
(二次冷却水質量偏差) →  
((二次冷却水温度偏差 変化))

((二次側内部エネルギー偏差 変化)) →  
(二次側内部エネルギー偏差) →  
((二次冷却水温度偏差 変化))

((二次冷却水温度偏差 変化)) →  
(二次冷却水温度偏差) →  
((二次冷却水温度)(安全弁流量偏差)(SG内水温度偏差 変化))((二次側内部エネルギー偏差 変化))  
(二次冷却水温度偏差 変化))

((フィルター後の比例ダンプ温度誤差信号 変化)) →  
(フィルター後の比例ダンプ温度誤差信号) →  
((誤差温度信号)(フィルター後の比例ダンプ温度誤差信号 変化))

((二次側流量偏差測定値 変化)) →  
(二次側流量偏差測定値) →  
((参照温度信号)(二次側流量偏差測定値 変化))

((蒸気ダンプ弁流量偏差 変化)) →  
(蒸気ダンプ弁流量偏差) →  
((二次側流量偏差)(蒸気ダンプ弁流量偏差 変化))

((加圧器内蒸気質量偏差 変化)) →  
(加圧器内蒸気質量偏差) →

((サージ質量 サージエンタルピー)(加圧器内蒸気容量偏差 変化)(加圧器圧力偏差 変化)(加圧器圧力偏差補償値 変化))

((加圧器内冷却水質量偏差 変化)) →  
(加圧器内冷却水質量偏差) →  
((加圧器内冷却水エンタルピー偏差))

((加圧器内冷却水内部エネルギー偏差 変化)) →  
(加圧器内冷却水内部エネルギー偏差) →  
((加圧器内冷却水エンタルピー偏差)(スクラム反応度))

((加圧器圧力偏差 変化)) →  
(加圧器圧力偏差) →  
((総反応度)(フラッシング質量)(炉心内圧力トリップマージン)(スプレー質量)(SG内圧力トリップマージン)(逃し弁による質量)(バックアップヒータ出力)(加圧器圧力偏差補償値 変化))

((加圧器内蒸気容量偏差 変化)) →  
(加圧器内蒸気容量偏差) →  
NIL

((加圧器内冷却水容量偏差 変化)) →  
(加圧器内冷却水容量偏差) →  
((サージ質量 サージエンタルピー)(バックアップヒータ出力)(加圧器内蒸気容量偏差 変化)(加圧器圧力偏差 変化)(加圧器圧力偏差補償値 変化))

((バックアップヒータ出力偏差 変化)) →  
(バックアップヒータ出力偏差) →  
((サージ質量 サージエンタルピー)(加圧器内冷却水内部エネルギー偏差 変化)(加圧器内蒸気容量偏差 変化)(加圧器圧力偏差 変化)(加圧器内冷却水容量偏差 変化)(バックアップヒータ出力偏差 変化)(加圧器圧力偏差補償値 変化))

((平均温度偏差 変化)) →  
(平均温度偏差) →

((フィードウォーター流量偏差)(比例ダンブ温度誤差信号)((平均温度偏差補償値 変化))((平均温度偏差遅れ値 変化)))

((平均温度偏差補償値 変化)) →

(平均温度偏差補償値) →

((制御棒位置による誤差温度信号)(制御棒速度による誤差温度信号)((平均温度偏差補償値 変化)))

((加圧器圧力偏差補償値 変化)) →

(加圧器圧力偏差補償値) →

((制御棒位置による誤差温度信号)(制御棒速度による誤差温度信号)((加圧器圧力偏差補償値 変化)))

((炉心出力偏差補償値 変化)) →

(炉心出力偏差補償値) →

((制御棒位置による誤差温度信号)(制御棒速度による誤差温度信号)((炉心出力偏差補償値 変化)))

((制御棒反応度 変化)) →

(制御棒反応度) →

((総反応度))

((ステップ外乱SG内水温度偏差補償値 変化)) →

(ステップ外乱SG内水温度偏差補償値) →

((SG内圧力トリップマージン)((ステップ外乱SG内水温度偏差補償値 変化)))

((平均温度偏差遅れ値 変化)) →

(平均温度偏差遅れ値) →

((SG内圧力トリップマージン)((平均温度偏差遅れ値 変化)))

((ステップ外乱SG内水温度偏差 変化)) →

(ステップ外乱SG内水温度偏差) →

((ステップ外乱SG内水温度偏差補償値 変化)))

((二次側流量偏差)(SG内圧力トリップマージン)(炉心出力)(加圧器内冷却水内部エネルギー偏差)) →

(スクラム反応度) →

((総反応度))

((二次冷却水温度偏差)) →

(安全弁流量偏差) →

((二次側流量偏差))

((加圧器圧力偏差)) →

(逃し弁による質量) →

((サージ質量 サージエンタルピー)((加圧器内蒸気質量偏差 変化))((加圧器内蒸気容量偏差 変化))

(加圧器圧力偏差 変化))((加圧器圧力偏差補償値 変化)))

((加圧器圧力偏差)(加圧器内冷却水容量偏差)) →

(バックアップヒータ出力) →

((バックアップヒータ出力偏差 変化)))

((制御棒位置による誤差温度信号)) →

( $\delta$ ) →

((制御棒速度)((制御棒反応度 変化)))

((制御棒速度による誤差温度信号)( $\delta$ )) →

(制御棒速度) →

((制御棒反応度 変化)))

## 研究発表一覧

### 寄稿論文

- [1]秋吉, 西田, "大規模システムの挙動理解のための階層型定性シミュレーション", 情報処理学会論文誌, Vol.32, No.1, pp.32-41 (1991)
- [2]秋吉, 西田, "定量的解析との対応操作による定性シミュレーション", 人工知能学会誌, Vol.7, No.5, pp.837-849 (1992)
- [3]秋吉, 西田, 塩谷, 合田, "定性推論を利用した故障木作成手法とその保護リレーへの適用", 電気学会論文誌B分冊Vol.114, pp.410-418, No.4 (1994)
- [4]Akiyoshi, M. and Nishida, S., "Towards Generating Causal Explanation: Qualitative Simulation with Association Mechanism to Quantitative Information", *IEEE Trans. on Industrial Electronics* (to appear in April, 1996)

### 国際会議

- [5]Akiyoshi, M. and Nishida, S., "A Computer-based Supporting Environment for Causal Understanding of Large Scale Systems by Hierarchical Qualitative Simulation", *Proc. of 1990 IEEE Conference on Systems, Man and Cybernetics*, pp.823-825 (1990)
- [6]Akiyoshi, M. and Nishida, S., "A Qualitative Simulation-based Learning Environment: How to Enhance Causal Understanding of Complex Phenomena in Large-scale Plants", *Proc. of IEEE Phoenix Conference on Computer and Communications*, pp.531-537 (1993)
- [7]Akiyoshi, M. and Nishida, S., "Towards Generating Causal Explanation: Qualitative Simulation with Association Mechanism to Quantitative Information", *Proc. of 1994 Seiken/IEEE Symposium on Emerging Technologies & Factory Automation*, pp.116-123 (1994)

### 研究会

- [8]秋吉, 西田, "階層型定性シミュレーションによるシステムダイナミクスの理解支援", 情報処理学会知識工学と人工知能研究会資料, 65-5, pp.61-69 (1989)
- [9]秋吉, 西田, "システムダイナミクスの因果理解における人間の思考を重視した支援システム-BECAUSE-", 人工知能学会研究会資料, SIG-HICG-8904-2, pp.11-20 (1990)
- [10]秋吉, 西田, "定量データから生成した時間概念を用いた定性シミュレーション手法", 情報処理学会人工知能研究会資料, 74-9, pp.79-86 (1991)
- [11]秋吉, 西田, "定性推論を利用した故障木作成方式", 情報処理学会人工知能研究会資料, 86-9, pp.65-72 (1993)

### シンポジウム

- [12]秋吉, 西田, "階層型定性シミュレータによるシステムダイナミクスの因果理解支援", 計測自動制御学会第12回知能システムシンポジウム論文集, pp.7-12 (1990)
- [13]秋吉, 西田, "定性シミュレータによる因果理解支援システム", *Proc. of International Symposium "Computer World '91"*, pp.149-156 (1991)
- [14]秋吉, 西田, "定量モデルに内在する変数間の依存構造を用いた定性モデル構築手法", 計測自動制御学会合同シンポジウム (第17回システムシンポジウム, 第14回知能システムシンポジウム, 第1回ニューラルネットワークシンポジウム), pp.311-315 (1991)
- [15]秋吉, 西田, "定性シミュレーションに基づく因果理解支援システム", 日本機会学会第2回設計工学システム部門講演会論文集, pp.339-344 (1992)
- [16]秋吉, 西田, "定性推論による対象の状態遷移を考慮した故障木作成手法",

計測自動制御学会第19回知能システムシンポジウム論文集, pp.79-83

(1994)

#### 全国大会

[17]秋吉, 西田, "階層型定性シミュレーションによるシステム挙動の理解支援", 情報処理学会第37回全国大会論文集2K-2, pp.699-700 (1988)

[18]秋吉, 西田, "理解支援のための階層型定性シミュレーション(1)ーモデルの表現ー", 情報処理学会第38回全国大会論文集4F-2, pp.456-457 (1989)

[19]秋吉, 西田, "理解支援のための階層型定性シミュレーション(2)ーユーザー・インターフェースー", 情報処理学会第38回全国大会論文集4F-3, pp.458-459 (1989)

[20]秋吉, 西田, "階層型定性シミュレータによるシステム・ダイナミクスの原因理解支援システムーBECAUSEー", 情報処理学会第40回全国大会論文集5D-7, pp.290-291 (1990)

[21]秋吉, 西田, "定量データを利用した定性シミュレーション手法", 情報処理学会第42回全国大会論文集2F-6, pp.203-204 (1991)

[22]秋吉, 西田, "定性推論に基づく故障木作成方式ー構造情報と部品動作記述から導いた定性モデルの利用ー", 情報処理学会第45回全国大会論文集2H-7, pp.111-112 (1992)

#### その他

[23]秋吉, 西田, "メンタル・モデルを用いた理解支援方式に関する考察", 日本認知科学会第7回大会論文集, pp.92-93 (1990)

[24]秋吉, 西田, "物理システムの機能概念と挙動の因果木を用いた説明についての考察", 日本認知科学会第8回大会論文集, pp.96-97 (1991)