

## 第 5 章

## 故障診断の自動化

本章では、自己修復機械の故障診断機能を実現する方法を提案する。

まず、従来の故障診断自動化、支援手法の問題点について述べる。次に、因果的に関連した複数の故障、および、機械構造を変化させる故障を扱うために、単一故障発生過程仮説を提案し、この仮説に基づく故障診断手法を述べる。さらに、対象から得られるセンサ値の曖昧さを考慮するために、定性物理にファジィ理論を導入したファジィ定性物理を用いた故障診断手法について述べる。

### 5.1 従来の故障診断手法

本節では、従来の故障診断自動化手法の問題点と本研究で明らかにすべき点について述べる。

第2.1節で述べたように、故障診断作業は人間の高度な判断を中心とした自動化の困難な作業であるが、第2.2.3節で述べたように、近年の人工知能技術の発達により数多くの自動化の試みがなされている。現在の研究動向としては、保全の特徴である不定性、非繰り返し性、多様性に対応可能性があるモデルベースト推論を用いた故障診断手法が多く見られ、本研究もこの範疇に入るものである。しかし、これらの手法の多くは、計算量を少なくするために単一故障仮説 (Single Fault Assumption) を用いており、故障診断の対象範囲が限定されている。

これに対し、de Kleerら [de Kleer87] は、General Diagnostic Engine(GDE)において、複数故障仮説 (Multiple Faults Assumption) に基づく故障診断を次のように実現している。一般に、複数故障仮説を用いると故障候補が膨大になり、組合せ的爆発を起こす。GDEでは、この膨大な探索空間を極小集合で代表させて管理することにより容易に扱っている。すなわち、この手法では診断対象の部品の正常状態の知識のみを利用し、各部品の正常以外の状態を故障とし、論理的な「否定」として表し、ATMS[de Kleer86]により管理している。GDEのアルゴリズムの概略を以下に示す。

1. 故障の症状から、故障の症状と矛盾する正常部品の極小の組を論理的に導出し、この「矛盾」と矛盾する故障候補を故障候補空間から削除する。例えば図5.1に示すような簡単な機械において、故障の症状が「ファンF1は回転するが、ファンF2は回転しない」であった場合、「ファンF2が回転しない」ことより矛盾の組(F2,G1,G3,M)(すなわち、これら全ての部品が正常な場合はF2が回転しないことに矛盾する)を導出し、これから故障の候補[F1]、[G2]、[F1,G2]<sup>1</sup>が削除される。同様にこのときファンF1が回転することから、矛盾の組(F1,F2,G2,G3)が生成され、故障候補[G1]、[M]、[G1,M]が削除される。
2. 1.により決まる故障候補の空間をその極小集合の組で代表させ、それを故障候補とする。上記の例では、故障候補は[F2]、[G3]、[F1,G1]、[F1,M]、[G1,G2]、[G2,M]となる。

<sup>1</sup>[F1,G2]は、F1とG2のみが同時に故障したことを示す。



3. 得られた結果から、診断のコストが最小になるように点検箇所を選択し点検を行ない、故障候補を絞り込む。これを繰り返すことにより、故障を十分に絞り込むことができる。

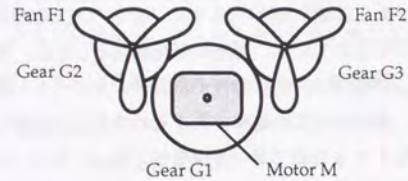


図 5.1: 簡単な機械の例

以上、第 2.2.3 節と合わせて、モデルベース推論を用いた手法を中心に、故障診断自動化手法の特徴を述べたが、これらの特徴と問題点は以下のように整理できる。

#### 1. 経験的知識に基づく手法に対するモデルベース推論の優位性

経験的知識に基づく故障診断手法は未経験の故障に対応できない、知識獲得が困難等の問題点があり、この点に関してモデルベース故障診断手法が優れている。このとき、モデルベース推論を実現する手法として定性物理 [Weld89] が重要な役割を果たす。

#### 2. 単一故障仮説と複数故障仮説

単一故障仮説に基づいたモデルベース故障診断は、その適用範囲に限界があり、この点に関して複数故障仮説に基づく de Kleer らの GDE が優れている。

#### 3. 故障を論理的に扱い、物理的意味付けがなされていない

GDE を含め多くの故障診断手法では、故障を正常状態でない状態と表しており、故障により部品の振舞いに変化することを考慮に入っていない。特に Reiter [Reiter87] は、このような考え方で故障診断を形式化しており、ここでは、アブノーマル述語  $ab$  を導入し、部品 A の正常状態を  $\neg ab(A)$ 、故障状態を  $ab(A)$  と表現している。Struss ら [Struss89] が指摘するように GDE も同様の問題を抱えている。例えば、図 5.1 において故障の症状「ファン F1 は回転するが、ファン F2 は回転しない」に対し故障候補  $[G2, M]$  を導出しているが、この

候補を物理的に考えると「モーターが故障して動力を発生しておらず、さらに、ギア G2 が故障して動力を受けていないにも関わらず回転している」となり、物理的には意味をなしていない。この問題は、これらの手法で故障時における物理現象や部品の挙動に関する知識を利用していないことに起因している。また、故障時の対象の挙動や状態を知らなければ有効な修復計画を立てることができないと考えられるので、本研究の目的である故障診断、および、修復作業の自動化のためには、物理的に正しい故障診断を行なうことは重要な課題である。

#### 4. 構造変更を扱えない

GDE ではデバイス主義の定性物理に基づき、部品間の入出力関係（デバイス・トポロジ）を用いて故障診断を行なうが、デバイス・トポロジが変化するような故障状態は扱えない。この点に関しては、第 2.2.3 節で述べた全ての故障診断手法に共通している。そのため図 5.1 において、例えば、ファン F1 と F2 が干渉し系全体が回転しなくなるという故障状態、導線間のショート、パイプの液洩れ等の対象構造が変化する状況は扱えない。この問題は、保全の不定性への対応に限定をつけていることになる。

本章では、1. の理由により定性物理を用いたモデルベース推論に基づく故障診断手法を提案するが、3.、4. の問題は機械系を対象にした自己修復システムを構築するために解決しなければならない問題であると考え、故障時の状態を推論可能で、かつ、構造変更を伴う故障を扱える診断手法を提案することを目的とする。この場合、2. に関して、複数故障仮説を用いると探索空間が膨大になり診断が困難になると考えられるので、次節で述べる単一故障発生過程仮説を導入する。



## 5.2 単一故障発生過程仮説に基づく故障診断手法

### 5.2.1 単一故障発生過程仮説

本研究では、次のように言葉の定義を行なう。まず、対象系に規定されている全ての機能が発現されている時、その対象系は「正常」であるといい、そうでない時は、対象系は「異常」であり、対象系内に少なくとも一つの「故障(fault)」が存在することとする。「故障」は診断対象に関するFBSダイアグラムにおける「状態」の変化、すなわち、構成要素の属性値の変化、構成要素間の関係の変化、構成要素の生成、消滅のうち、故障の症状を引き起こす原因となっているものである。また、「故障の症状(fault symptom)」は、故障発生時に観察される対象系の機能や挙動の変化である。このとき、第2.1.4項で述べたように、「故障」や「故障の症状」は機械の機能に依存して規定されるため、設計者や使用者により間接的に定められることになる。

さらにここでは、故障を引き起こす物理現象に注目しこれを「故障原因」と呼ぶことにする。故障原因はある条件で引き起こされ、対象系に対して属性値の変化、構造変化等を引き起こす物理的な現象である。さらに、ある故障原因が発生するとそれが別の故障原因を引き起こすというように連鎖的に故障原因が発生する場合がある。本研究ではこのような故障原因の因果的連鎖を「故障発生過程」と呼ぶことにする。このとき、故障発生過程を引き起こす原因となっている一つ、もしくは、複数の故障原因を「根本的故障原因」と呼ぶ。根本的故障原因が発生する条件は部品の劣化、外乱等が考えられる。ここでは、これら故障原因の発生機構を追求することは本研究の目的とはせず、故障物理学、負荷論の成果として故障原因となり得る物理現象がモデル化されていることを前提とする。本研究では、従来の故障診断作業を拡張し、いわゆる故障解析作業[Shiomi91]を含め、故障の症状からその原因となっている「故障発生過程」を導出する作業を故障診断と呼び、自動化を試みることにする。故障発生過程を導出することで、改良設計や修復を行なうためにより有効な情報を得ることが可能になり、また、故障した対象系の状態をより正確に推定することが可能になると考えられる。ここでは探索空間を狭め故障診断を容易にするために、「一つの対象機械において、同時に最大一つの故障発生過程しか起こらない」と仮定し、この仮定を「単一故障発生過程仮説」と呼ぶことにする。すなわち、因果的に全く関係の無い故障が同時に発生することは確率的に低いとしないことにする。これにより、一つの対

象機械において複数の場所で故障が発生した場合も因果的に互いに関連していると考えられ、因果的な関係に基づき故障発生過程を辿ることにより容易に故障診断を行なうことが可能になる。ここで言う単一故障発生過程仮説には、単一の故障発生過程がその根本的原因として複数の故障原因の発生を前提としているものも含まれる。しかし、同一箇所に発生した故障原因であっても、それらが互いには因果的に独立であるならば、この仮説の範囲外である。

例えば、図5.2に示す電気回路において、導線の劣化により被覆が剥がれ、導線がショートし、過電流が流れ、結果として、ヒューズが溶断し、電球が点灯しないという故障の発生過程を考える。このとき、根本的故障原因は「被覆の劣化」現象、故障は導線A、Bの破覆、接触、および、ヒューズの断線であり、故障の症状は電球の不点灯である。この故障発生過程を図5.3に示す。

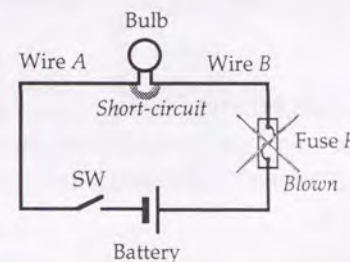


図5.2: ショートした電気回路

### 5.2.2 知識表現

次に本節で用いる知識表現の方法について述べる。第4.6節で述べたように、基本的にはデバイス主義の定性物理に基づくFBSダイアグラムにより対象を表現するが、前項で述べたような故障発生過程を推論するため、故障原因等の物理現象を直接表現可能なように拡張を加える。

本故障診断手法では、知識を物理法則等の原理原則的な知識と、対象の構造を記述する対象モデルの二種類に分ける。これにより、モデルベース推論を利用すること



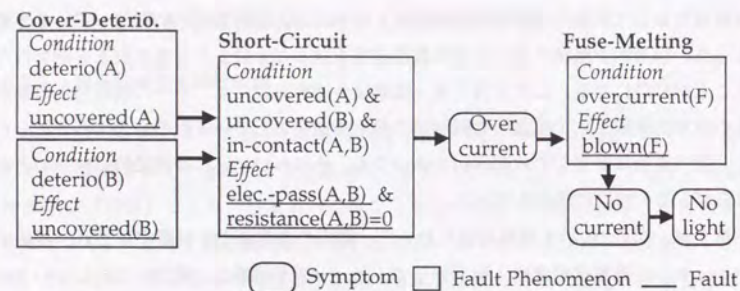


図 5.3: 故障発生過程の例

によって得られる特徴に加えて、経験的知識に基づく故障診断手法には無い以下のような特徴を得ることができる。

1. 原理原則的知識は比較的整理されているので、ヒューリスティックスよりも知識獲得が容易である。また、対象モデルは設計情報を利用することにより容易に構築できる。このため一般に、経験的知識に基づく手法よりも知識獲得が容易になると考えられる。
2. 経験的知識に基づく故障診断システムは、対象系が変更された場合、知識の多くをそのまま利用することが不可能になるが、本研究の手法では、対象モデルを入れ換えることにより、原理原則的知識はそのまま利用して様々な対象系に対して故障診断が可能になる。

#### (1) 原理原則的知識

原理原則的知識は、診断対象に関わる一般的物理法則を対象に依存しない形で記述した知識である。本研究では原理原則的知識として以下の二種類を用意する。

1. 部品、部品間の関係に関する知識
2. 物理現象に関する知識

#### 1. 部品、部品間の関係に関する知識

部品、部品間の関係に関する知識は、対象モデルを記述するために必要な部品や部品間の関係を一般化し、クラスとして階層的に記述したものである。図 5.4 に、部品クラス階層、関係クラス階層の一部を示す。各階層において、下位階層のクラスは上位クラスの性質を引き継いでいる。

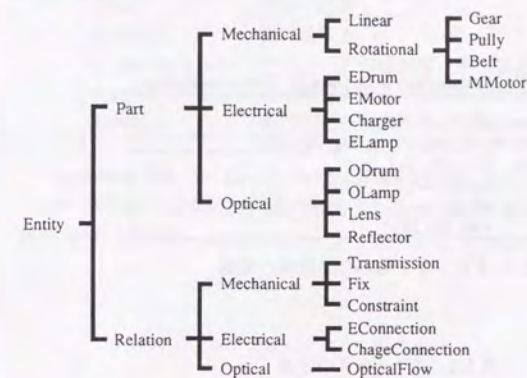


図 5.4: 部品、関係クラスの階層

各部品クラス、関係クラスは同じ枠組で記述されている。例として、ギアの定義を図 5.5 に、ギア同士の噛み合いを表す関係の定義を図 5.6 に示す。

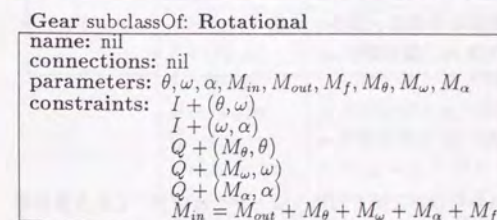


図 5.5: ギアの定義

図に示されている通り、各部品、関係は4種類の項目により記述される。name



は、各部品、関係の名前を記述し、connectionsには、その部品、関係が対象モデル内で接続している関係、部品を記述する。各部品、関係の性質は、parametersとconstraintsにより記述される。parametersにはその部品が持つ属性を表すパラメータが定義され、constraintsにはそのパラメータ間の関係が定性微分方程式により記述される。表5.1にパラメータの定義を、表5.2、表5.3にパラメータ間の関係の定義と種類をそれぞれ示す。

```
GearGearTransmission subclassOf: Transmission
name: nil
connections: nil
parameters:  $\theta_1, \theta_2, \omega_1, \omega_2, \alpha_1, \alpha_2, M_1, M_2$ 
constraints:  $Q + (\theta_1, \theta_2)$ 
               $eql(\omega_1, \omega_2)$ 
               $eql(\alpha_1, \alpha_2)$ 
               $eql(M_1, M_2)$ 
```

図 5.6: ギア・ギア噛み合い関係の定義

表 5.1: パラメータの定義

Parameter	
name	パラメータの名前
value	パラメータの値 (一次元の定性値)
derive	パラメータの微分値、すなわち +、0、-
qSpace	パラメータの量空間
support	自分の存在の前提になる部品、関係、現象
constant	true ならば定数であり、微分値が0。

## 2. 物理現象に関する知識

以上で、部品、部品間の関係の知識の記述を行なうことができるが、これら部品等の性質は、故障により変化する。また、導線のショートにより新たな導通が生成するように、故障により対象系の構造が変化する可能性もある。このような現象を1.の知識で記述することは難しい。そこで、部品等の性質変化、機械の構造変更を引き起こす「現象」を明示的に知識として記述し、現象知識ベースを構築することにする。ここでは、現象を発生条件とその影響により表現する。このため本研究では、表5.4に

表 5.2: パラメータ間の関係の定義

Constraint	
name	関係の名前
equation	拘束条件式
qSpaceRelations	パラメータの量空間の間の関係 <sup>1</sup>
support	この関係が成立する前提となる部品、関係、現象
direction	条件式内の因果的方向性 <sup>2</sup>

<sup>1</sup>二項関係ならば、 $(p_1, l_1) = (p_2, l_2)$  と、三項関係ならば、 $(p_1, l_1) = ((p_2, l_2), (p_3, l_3))$  と記述。

<sup>2</sup>例えば電球の入力電圧と発光量の関係のように一方の変化は他方に影響するが、もう一方の変化は他方に影響しない関係。

表 5.3: 拘束条件式の種類

種類	内容
$eql(p_1, p_2)$	$p_1$ と $p_2$ の符号、微分値が等しい
$I + (p_1, p_2)$	$p_1$ の微分値と $p_2$ の符号が等しい
$I - (p_1, p_2)$	$p_1$ の微分値と $p_2$ の符号が逆
$Q + (p_1, p_2)$	$p_1$ の微分値と $p_2$ の微分値が等しい
$Q - (p_1, p_2)$	$p_1$ の微分値と $p_2$ の微分値が逆
$add(p_1, p_2, p_3)$	$p_1 = p_2 + p_3$ を表す
$sub(p_1, p_2, p_3)$	$p_1 = p_2 - p_3$ を表す
$mul(p_1, p_2, p_3)$	$p_1 = p_2 \times p_3$ を表す
$div(p_1, p_2, p_3)$	$p_1 = p_2 \div p_3$ を表す



示す述語の論理和、論理積を用いて現象の発生条件を記述する。同様に、現象の影響は表5.5に示す述語の論理積により表現する。例えば、「ギア割れ」現象は、図5.7のように記述する。

すなわち、本研究では、正常時の各部品等の性質は1.で記述し、正常時以外に対象系上で生起する可能性のある部品の性質の変化等は2.の知識として記述する。

表5.4: 現象の発生条件の記述

述語	意味
<i>exist(name, Entity)</i>	クラス <i>Entity</i> かそのサブクラスの实体 <sup>1</sup> が存在する。この現象内では <i>name</i> でこの実体を参照する
<i>connect(entity<sub>1</sub>, entity<sub>2</sub>)</i>	<i>entity<sub>1</sub></i> と <i>entity<sub>2</sub></i> が接続している
<i>assignP(name, entity, pname)</i>	<i>entity</i> 内に <i>pname</i> で参照されるパラメータが存在する。この現象内では <i>name</i> でそのパラメータを参照する
<i>assignV(name, pname)</i>	パラメータ <i>pname</i> の値を一時変数 <i>name</i> に代入
<i>= (pname, value)</i>	<i>pname</i> の値が <i>value</i> に等しい
<i>&gt; (pname, value)</i>	<i>pname</i> の値が <i>value</i> より大きい
<i>&gt;= (pname, value)</i>	<i>pname</i> の値が <i>value</i> より小さくない
<i>&lt; (pname, value)</i>	<i>pname</i> の値が <i>value</i> より小さい
<i>=&lt; (pname, value)</i>	<i>pname</i> の値が <i>value</i> より大きくない
<i>not( 述語 )</i>	述語の否定

<sup>1</sup>ここで实体と呼んでいるのは、部品と部品間の関係の総称である。

## (2) 対象モデルの表現

対象モデルは、故障診断の対象となる特定の機械の構造、性質を記述した知識である。図5.8に、簡単な対象モデルの例を示す。図5.8が示すように、対象モデルは機能モデル、実体モデル (Entity Model)、パラメータモデルの三つの異なる記述レベルのモデルから構成される。

機能モデルは、第4章で述べた FBS ダイアグラムで用いる機能階層ネットワークを使用する。実体モデルは、対象系内に存在する部品とその接続構造を表現する部品

表5.5: 現象の影響の記述

述語	意味
<i>addConst(entity, constraint)</i>	<i>entity</i> に拘束条件 <i>constraint</i> を付加する
<i>addEntity(name, Entity)</i>	クラス <i>Entity</i> の実体を名前 <i>name</i> として生成し、対象系に付加する
<i>addP(entity, param, value, derive)</i>	<i>entity</i> にパラメータ <i>param</i> を付加する。初期値を <i>value</i> 、 <i>derive</i> とする
<i>changeConst(entity, cname, t/f)</i>	<i>entity</i> 内の拘束条件式 <i>cname</i> を有効/無効にする
<i>changePConstant(param, t/f)</i>	<i>param</i> を定数化/変数化する
<i>makeP(pname, value, derive)</i>	<i>pname</i> の値を <i>value</i> に、微分値を <i>derive</i> にする
<i>relate(part<sub>1</sub>, part<sub>2</sub>, Relation, name)</i>	関係クラス <i>Relation</i> のインスタンスを名前 <i>name</i> として新たに生成し、部品 <i>part<sub>1</sub></i> と部品 <i>part<sub>2</sub></i> の間を結ぶ
<i>rmP(pname)</i>	<i>pname</i> を対象系から削除する。この時関連する全ての拘束条件式も削除される
<i>rmEntity(entity)</i>	<i>entity</i> を対象系から削除する。この時 <i>entity</i> に依存する関係、パラメータ、拘束条件式も削除される。

## GearBreak subclassOf: Phenomenon

<b>Condition:</b>	<i>exist(gear<sub>1</sub>, Gear) &amp; exist(gear<sub>2</sub>, Gear) &amp; exist(rel, GearGear-Transmission) &amp; connect(gear<sub>1</sub>, rel) &amp; connect(rel, gear<sub>2</sub>) &amp; assignP(<i>M<sub>out</sub></i>, gear<sub>1</sub>, <i>M<sub>out</sub></i>) &amp; assignP(<i>M<sub>in</sub></i>, gear<sub>2</sub>, <i>M<sub>in</sub></i>) &amp; &gt; (<i>M<sub>out</sub></i>, MAX)</i>
<b>Effect:</b>	<i>rmEntity(gear<sub>1</sub>) &amp; makeP(<i>M<sub>in</sub></i>, 0, 0)</i>

図5.7: 「ギア割れ」現象



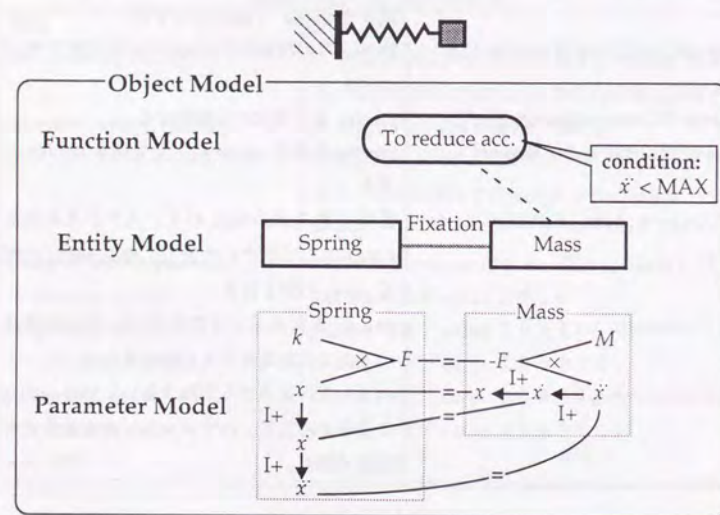


図 5.8: 対象モデルの例

接続グラフであり、(1)で述べた部品クラス、部品間関係クラスのインスタンスにより構成される。実体モデルから、部品、部品間関係に記述されたパラメータ間の関係を抽出すると対象機械全体の性質を表すパラメータ関係のネットワークを構成することができる。これをパラメータモデルと呼ぶ。(1)で述べたように各パラメータは量空間に定義された一次元の定性値を取る。このとき、特定の対象における定性的な正常値をパラメータモデル内の各パラメータの量空間上に記述する。故障診断時に、この正常値と実際の値を比較することにより、各パラメータ値が正常値よりも大きい、小さいという故障診断を行なう上で有効な情報を容易に得ることが可能になる。

このように対象モデルの挙動、状態レベルの記述を実体モデルとパラメータモデルという二つのレベルに分けることにより、機械系、電気系等様々な視点にまたがったシステムをパラメータモデルとして統一的に表現できる。また、後述するように本研究の故障診断手法はパラメータモデル上の探索により実現されるので、視点や対象系に依存しない汎用的なものとすることができる。一方で、パラメータモデルの示す状態の物理的意味を表現するために、特に、対象系上で発生する現象を知るためには実体モデルと対応付けることが有効になる。

### 5.2.3 挙動シミュレーションの方法

第5.2.4項で述べるように、本研究の故障診断では故障発生時の対象系の挙動をシミュレーションする。これを本研究では「故障シミュレーション」と呼ぶ。また、第6章で述べるように、修復計画時においても修復作業実行時の対象系の挙動をシミュレーションする。これらのシミュレーションを実現するために、本研究では、前項で述べた対象表現に基づく汎用の挙動シミュレータを開発した。ここでは、この挙動シミュレーションの方法について述べる。

本研究でいう挙動シミュレーションは、与えられた対象モデル  $m_0$  と対象モデル内の各パラメータの初期値(値と微分値の組)の集合  $V_0$  から対象系の時間的な状態遷移列を求めることであり、以下のように書ける。

$$(m_0, V_0) \rightarrow (m_1, V_1) \rightarrow (m_2, V_2) \rightarrow \dots \quad (5.1)$$

本研究では、第4.2節で述べた QSIM(Qualitative SIMulation) アルゴリズム [Kuipers86] に基づいて挙動シミュレーションを行なうことにする。QSIM アルゴリズムは与えられた連立定性微分方程式、これは本研究の対象モデル  $m_0$ 、特に、パラメータモデル



に対応するが、と初期値の組  $V_0$  からパラメータ値の時間的な状態遷移列を求めるものである (式 5.2 参照)。

$$(m_0, V_0) \rightarrow (m_0, V_1) \rightarrow (m_0, V_2) \rightarrow \dots \quad (5.2)$$

また、QSIM アルゴリズムにおける時間の概念は長さの無い時点 (time point) と長さのある時間 (time interval) の繰り返しにより、以下のように表される。

$$\dots t_i, (t_i, t_{i+1}), t_{i+1}, (t_{i+1}, t_{i+2}), \dots \quad (5.3)$$

本研究ではこの QSIM アルゴリズムを以下の二点に関して変更して用いることにする。

1. QSIM アルゴリズムは本研究の「現象」のように方程式系を動的に変更したり、値を非連続的に変更するような枠組には対応していないので、これらの現象による変更を推論可能なように拡張を行なう。
2. QSIM アルゴリズムでは、変数値が区間値であり、かつ、時間微分値が 0 であることを許さないという仮定を置いており、変数値が区間値で微分値が 0 になった場合、動的にランドマークを生成する。これにより、例えば、単振動、減衰振動、拡大振動の区別をつけることができる [Kuipers86] が、対象系が複雑になるに従い意味不明のランドマークが多数生成されてしまう可能性がある [Kurumatani89]。そこで本研究では、ランドマークの動的生成を行わない。これを実現するために、車谷 [Kurumatani89] により提案された改良型の推論則を用いることにする。

本研究の挙動シミュレーションは、以下に示す方法により実現する。

#### 1. 状態解析

- (a) 与えられた対象モデル  $m_i$  上で、条件部のマッチングする現象を第 5.2.2 項で述べた現象に関する知識から探索する。すなわち、本段階は、現象に関する知識を  $K_p$ 、条件部がマッチングした現象の集合を  $P_i$  とすると以下のように書ける。マッチングした現象が無ければ、2. 以下を実行する。

$$m_i, V_i, K_p \mapsto P_i \quad (5.4)$$

- (b) マッチングした物理現象の影響を対象モデルに付加する。すなわち、前項で求められた  $P_i$  の各現象の影響部の集合を  $P_e$  としたとき、以下の操作を実行する。

$$m_i, V_i, P_e \mapsto m'_i, V'_i \quad (5.5)$$

例えば、図 5.7 の「ギア割れ」現象が生じた場合、該当するギア、そのギアに接続する関係を実体モデルから削除し、削除された部品、関係に依存するパラメータ、パラメータ間の関係をパラメータモデルから削除する。このとき、表 5.5 の述語が対象モデルに適用されるため、パラメータモデル上の各パラメータ値の再決定や新たに付加されたパラメータ値の決定を行わなければならない。例えば、*makeP* 述語により定数値が非連続的に変化した場合は前者に相当し、*addP* 述語によりパラメータが追加された場合が後者に相当する (表 5.6 参照)。

表 5.6: 各影響述語の作用

	状態再決定	パラメータ値決定
<i>addConst</i>	×	×
<i>addEntity</i>	×	○
<i>addP</i>	×	○
<i>changeConst</i>	×	×
<i>changePCConstant</i>	×	×
<i>makeP</i>	○	×
<i>relate</i>	×	○
<i>rmP</i>	×	×
<i>rmEntity</i>	×	×

○: 必要 ×: 不要

#### ● 状態再決定

状態再決定は、与えられた  $V_i$  を初期化し、 $m'_i$  から新たに  $V'_i$  を求める操作である。すなわち、定数以外のパラメータに関して、各パラメータが取り得る全ての値と微分値の組を求め、この値の候補の集合に対して、3. の (b) 以下を実行し、 $V'_i$  を求める。しかし一般に  $V'_i$  を一意に決めることはできない。そこでここでは、Forbus [Forbus89a] と同様、非連続的な遷移の結果として、もとの状態  $V_i$  に近い状態に遷移すると仮定する。すなわち、 $V_i$  と導出された値の各候補  $V'_i$  を比較



し、最も近い状態  $V'_i$  を求める。 $V'_i$  が複数の求まる場合には状態を分岐させる。

● パラメータ値決定

$m'_i$  には、 $m_i$  に含まれないパラメータが含まれている可能性がある。このパラメータの値を決定しなければならない。そこで、値未定の各パラメータが取り得る全ての値と微分値の組を求め、この値の候補の集合に対して、3. の (b) 以下を実行し、 $V'_i$  を求める。

以上により、対象モデル  $m'_i$ 、および、パラメータの値  $V'_i$  を決定することができる。本研究では、このような遷移を時間が進行しない状態遷移として履歴に登録する。すなわち、時間  $t_i$  における  $(m_i, V_i)$  の次の状態は時間  $t_i$  における  $(m'_i, V'_i)$  となる。新たに決定された状態においてさらに現象が生起することがあるので、求められた  $m'_i$ 、 $V'_i$  に対して、1. の状態解析を再帰的に行なう。

2. 静止判定

現在推論の対象としている  $(m'_i, V'_i)$  において、 $V'_i$  の全ての微分値が0の場合、その状態は「静止状態」である。静止状態は状態遷移を起こさないで、それ以上挙動推論を行なわない。

3. 状態決定

1. で求められた最終的な  $(m'_i, V'_i)$  において、 $m'_i = m_{i+1}$  であるので、QSIM アルゴリズムを用いて  $V_{i+1}$  を以下の方法により導出する。すなわち、この段階は次のように書ける。

$$m_{i+1}, V'_i \mapsto V_{i+1} \quad (5.6)$$

ここでは例として、質点の自由落下を考えてみる。与えられるパラメータ間の関係と初期条件は次のようなものである。

$$I + (Y, V) \quad (5.7)$$

$$I + (V, A) \quad (5.8)$$

$$g < 0 \quad (5.9)$$

$$QS(A, t_0, t_1) = (g, 0) \quad (5.10)$$

$$QS(V, t_0, t_1) = ((0, \infty), -) \quad (5.11)$$

$$QS(Y, t_0, t_1) = ((0, \infty), +)^1 \quad (5.12)$$

(a) 各パラメータの遷移の可能性の導出

各パラメータの状態(値と微分値の組)から、表5.7に従って各パラメータの遷移の可能性の候補集合を提示する。例題に対しては、表5.7から、各パラメータ遷移の候補として次のものが導出される<sup>2</sup>。

$$A \quad I1: (g, 0) \Rightarrow (g, 0) \quad (5.13)$$

$$V \quad I5: ((0, \infty), -) \Rightarrow (0, 0) \quad (5.14)$$

$$V \quad I6: ((0, \infty), -) \Rightarrow (0, -) \quad (5.15)$$

$$V \quad I7: ((0, \infty), -) \Rightarrow ((0, \infty), -) \quad (5.16)$$

$$V \quad I9: ((0, \infty), -) \Rightarrow ((0, \infty), 0) \quad (5.17)$$

$$Y \quad I4: ((0, \infty), +) \Rightarrow ((0, \infty), +) \quad (5.18)$$

$$Y \quad I8: ((0, \infty), +) \Rightarrow ((0, \infty), 0) \quad (5.19)$$

表 5.7: 状態遷移則 [Kurumatani89]

Transition rules from time point			Transition rules from time interval		
TYPE	CURRENT	NEXT	TYPE	CURRENT	NEXT
P1)	$(L_i, 0)$	$(L_i, 0)$	I1)	$(L_i, 0)$	$(L_i, 0)$
P2)	$(L_i, 0)$	$((L_i, L_{i+1}), +)$	I2)	$((L_{i-1}, L_i), +)$	$(L_i, 0)$
P3)	$(L_i, 0)$	$((L_{i-1}, L_i), -)$	I3)	$((L_{i-1}, L_i), +)$	$(L_i, +)$
P4)	$(L_i, +)$	$((L_i, L_{i+1}), +)$	I4)	$((L_{i-1}, L_i), +)$	$((L_{i-1}, L_i), +)$
P5)	$((L_i, L_{i+1}), +)$	$((L_i, L_{i+1}), +)$	I5)	$((L_i, L_{i+1}), -)$	$(L_i, 0)$
P6)	$(L_i, -)$	$((L_{i-1}, L_i), -)$	I6)	$((L_i, L_{i+1}), -)$	$(L_i, -)$
P7)	$((L_{i-1}, L_i), -)$	$((L_{i-1}, L_i), -)$	I7)	$((L_i, L_{i+1}), -)$	$((L_i, L_{i+1}), -)$
P8)	$((L_i, L_{i+1}), 0)$	$((L_i, L_{i+1}), 0)$	I8)	$((L_i, L_{i+1}), +)$	$((L_i, L_{i+1}), 0)$
P9)	$((L_i, L_{i+1}), 0)$	$((L_i, L_{i+1}), +)$	I9)	$((L_i, L_{i+1}), -)$	$((L_i, L_{i+1}), 0)$
P10)	$((L_i, L_{i+1}), 0)$	$((L_i, L_{i+1}), -)$	I10)	$((L_i, L_{i+1}), 0)$	$((L_i, L_{i+1}), 0)$

<sup>1</sup>QS() は、ある時刻におけるパラメータの定性的状態(値と微分値)を示す。

<sup>2</sup>ランダムマーク  $\infty$  には到達しないので Y に関して I2 や I3 は除外される。



## (b) Causal Relation filtering

表5.3に示した各パラメータ間の関係に対して、前項で得たパラメータの遷移の可能性の組を作り矛盾するものを取り除く。このとき、関係の充足判定の基準は以下の三点である。

- 値の組が関係を満たす
- 四則演算に関しては、式を微分したものを値の組が満たす  
例えば、 $a = b + c$ を時間微分すると  $\delta a = \delta b + \delta c$ を得るが、微分値の組がこの式を満たさなければならない。
- 関係に記述されたパラメータの量空間の間の関係を満たす  
各関係内の量空間の間の関係に基づいて判定を行う。パラメータ  $a, b, c$ が量空間の関係  $(a, l_1) = (b, l_2)$ 、または、 $(a, l_1) = ((b, l_2), (c, l_3))$ を持つとき、以下の右辺を評価することにより量空間の関係を判定することができる。

$$eq1(a, b) \rightarrow eq1(a - l_1, b - l_2) \quad (5.20)$$

$$Q + (a, b) \rightarrow eq1(a - l_1, b - l_2) \quad (5.21)$$

$$Q - (a, b) \rightarrow eq1(a - l_1, b - l_2) \quad (5.22)$$

$$add(a, b, c) \rightarrow add(a - l_1, b - l_2, c - l_3) \quad (5.23)$$

$$sub(a, b, c) \rightarrow sub(a - l_1, b - l_2, c - l_3) \quad (5.24)$$

$$mul(a, b, c) \rightarrow mul(a/l_1, b/l_2, c/l_3) \quad (5.25)$$

$$div(a, b, c) \rightarrow div(a/l_1, b/l_2, c/l_3) \quad (5.26)$$

自由落下の例でいえば、以下に示す組のうち  $c$  と書かれている組は削除される。

$I + (Y, V)$	$I + (V, A)$
$(I4, I5) \ c$	$(I5, I1) \ c$
$(I4, I6) \ c$	$(I6, I1)$
$(I4, I7)$	$(I7, I1)$
$(I4, I9) \ w$	$(I9, I1) \ c$
$(I8, I5)$	$w$
$(I8, I6)$	
$(I8, I7) \ c$	
$(I8, I9) \ c$	

## (c) pairwise constraints filtering

各パラメータに注目し、そのパラメータが含まれる全ての関係に共通の遷移を選択する。自由落下の例でいえば、上記の組のうち  $w$  と書かれている組が削除される。

## (d) generating global interpretations

以上で得られた各関係に書かれた値の候補から、対象系全体で無矛盾な値の組 (global interpretation) を作る。自由落下の例でいえば、以下の二つの値の組が得られる。

Y	V	A
I4	I7	I1
I8	I6	I1

## (e) global filtering

全体の状態を見て、filteringを行う。ここでは、3.を始める前の状態  $V_i$  と等しい global interpretation を削除する。上記の例ではこの結果、 $(I8, I6, I1)$  のみが導出される。

以上により、遷移後のパラメータ値の状態  $V_{i+1}$  を導出できる。この時、複数の状態が導出された場合は状態遷移に分岐が生じる。

4. 3.で導出された各分岐に対して、1.以降の推論を行なう。推論すべき状態がない場合は、推論を終了させる。

以上により状態遷移の履歴を導出することができる。本研究では、一つの状態を、対象モデル  $m_i$ 、値の組  $V_i$ 、および、その時に生起した現象  $P_i$  の組  $(m_i, V_i, P_i)$  として表現する。

## 5.2.4 故障診断の方法

本節の故障診断手法は以下の五段階からなる。例として、図5.9に示すパラメータモデルを使用することにする。また、パラメータモデルとは独立に現象知識として  $P1, P2, \dots$  が記述されているものとする。

## 1. 故障症状の入力

ユーザが対象機械における喪失機能を指定する。もしくは、対象機械に関するセンサ値を入力する。センサ値が入力された場合、対象モデルの機能階層と比較することにより、喪失機能を同定することができる。指定された喪失機能の



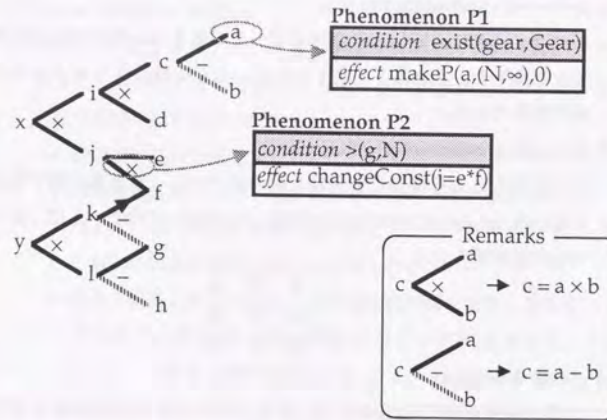


図 5.9: パラメータモデルの例

発現条件とセンサ値より、幾つかのパラメータ値の集合  $S$  が決められる。この  $S$  が、挙動、状態レベルでの故障の症状である。

## 2. 故障原因候補の導出

入力された故障の症状  $S$  を引き起こしている可能性のある現象、すなわち、故障原因の候補を以下の方法により導出する。

### (a) 目的パラメータの選択

まず、 $S$  の中から正常値を取っていない任意のパラメータを一つ選択する。ここでは、喪失機能の中で上位の階層にある機能の発現条件に記述されているパラメータを選択する。

### (b) 変化候補の導出

前項で選択されたパラメータ値の変化を引き起こす可能性のある全てのパラメータ値の変化、および、関連するパラメータ関係をパラメータモデルの探索により導出する。例えば、図 5.9 において、選択されたパラメータ値の変化が「 $x$  が正常値よりも大きい」であった場合、 $x$  の上昇を引き起こす可能性のある、 $i$  の上昇、 $j$  の上昇、 $x=i \times j$  の変化を導出し、さらに、 $i$  の上昇を引き起こす可能性のある、 $c$  の上昇... と探索を繰り返すこ

とにより、 $x$  の変化に直接関係のある変化として以下の候補を導出する。

$a, c, d, e, f, i, j$  の上昇  
 $b$  の下降  
 $x = i \times j, i = c \times d, c = a - b, j = e \times f$

### (c) 故障原因候補の探索

前項で導出されたパラメータ変化、関係の変化の集合に対し、各変化を引き起こす可能性のある現象を現象知識の中から探索する。すなわち、注目するパラメータ変化を引き起こす現象、注目する関係を変化させる現象、および、注目するパラメータを含む部品や部品間の関係を変化させる現象を全て導出する。図 5.9 の例でいえば、パラメータ  $a$  を変化させる現象 P1、関係  $j = e \times f$  を変化させる現象 P2 が導出される。

以上により、2.(a) で選択されたパラメータ変化を直接引き起こす可能性のある全ての故障原因候補を導出できる。単一故障発生過程仮説によれば、全ての故障は因果的に関係しているの、全ての故障の症状  $S$  も互いに因果的に関連しているはずであるから、 $S$  の全ての要素も導出された故障原因と直接的、間接的に関係しているはずである。

## 3. 故障発生過程候補の導出

2. で導出されたそれぞれの故障原因候補に対して、故障発生過程を導出する。注目する故障原因候補が生じた時点で故障原因候補の発生条件は満たされていたはずである。そこで、この発生条件をパラメータモデルに入力し、発生条件に記述されている各パラメータ値の条件を起点として、2.(b) ~ 2.(c) を実行することにより、注目する故障原因候補を引き起こす可能性のある現象を導出する。これにより、故障原因候補の因果的連鎖を一段階遡ることができる。同様の操作を適用できる現象が無くなるまで繰り返すことにより、ある故障原因候補を引き起こす現象の連鎖、すなわち、故障発生過程候補を導出することができる。

## 4. 故障シミュレーション

次に、故障発生過程のシミュレーションを行なう。前項で導出された各故障発生過程候補に対し、それぞれの根本的故障原因を対象モデルで生起させ、第 5.2.3 項で述べた挙動シミュレーションを実行することにより、故障発生時の対象の挙動を推論する。本研究では、現在の故障した機械の状態は安定な状態であると仮定し、推論結果得られる状態遷移の履歴の中で末端に位置する状態



を、ある根本的故障原因に対応する「故障モデル」と呼ぶことにする。このように故障シミュレーションを行なうことにより、以下の特徴が得られる。

- ユーザが入力した故障の症状  $S$  と矛盾する挙動を示す候補を排除できる。  
すなわち、102ページの問題点3で挙げた物理的一貫性の問題に対し、本研究では、記述された知識の範囲で物理的に正しい故障のみを導出することができる。
- ユーザが注目する故障症状、故障以外に同時に発生する可能性のある故障症状、故障を示すことができる。
- シミュレーションの結果として、故障発生過程と故障発生後の対象の状態を知ることができるので、点検、修復、改良設計等に有効な情報を得ることができる。

#### 5. 故障モデルの評価

各故障発生過程候補に対応する故障モデルと故障の症状  $S$  を比較することにより、 $S$  と矛盾する故障モデルを削除する。以上により、 $S$  に対して無矛盾な故障発生過程候補と故障モデルを導出することができる。このとき、故障発生過程候補が存在しない場合は、故障診断不能として推論を終了する。また、一つの故障発生過程候補のみを導出した場合には、その候補が実際に発生している故障発生過程であるとし、推論を終了する。

#### 6. 点検指示

故障発生過程候補が複数導出された場合には、候補の絞り込みを行なうためにユーザに点検を指示する。ここでは、 $S$  に矛盾しない故障モデル間でパラメータの値を比較し、値の一致度の小さなパラメータを導出し、ユーザに提示する。 $n$  個の故障モデルが存在し、あるパラメータが値  $v_1$  を  $n_1$  個、値  $v_2$  を  $n_2$  個、…値  $v_k$  を  $n_k$  個取る時 (ただし、 $\sum_{i=1}^k n_i = n$ )、一致度  $i$  を次式により与えることにする。

$$i = \sum_{i=1}^k n_i^2 \quad (5.27)$$

例えば、表5.8のような状態の場合には、c、a、bと順位付けされる。ユーザは順位付けされたパラメータの中から実機における点検容易性を考慮して点検を行ない、結果を入力し、5.を再び実行する。

表5.8: 故障モデル間でのパラメータ値の分散

パラメータ	モデル1	モデル2	モデル3	一致度
a	(0)	(0)	(0, N)	5
b	(0)	(0)	(0)	9
c	(0)	$(-\infty, 0)$	(0, N)	3

以上が本研究における故障診断手法であり、ユーザが点検を繰り返すことにより、故障発生過程候補数を十分絞り込むことが可能である。ただし、ここで扱っている情報は定性的なものであるため、常に故障発生過程を一つに絞り込むことは可能ではない。実際の故障診断作業においても点検不可能な箇所等があるため、保全作業者は必ずしも故障箇所を確定しているわけではなく、過去の経験等からもっともらしいものを選択しているように見受けられる。本手法でも、あり得る全ての故障発生過程とその因果関係をユーザに提示することにより、保全作業者の判断を支援することをその目的の範囲とする。

#### 5.2.5 インプリメンテーション

以上に述べた故障診断手法の有効性を検証するために Smalltalk-80 [Goldberg87] を用いて実験的な故障診断システムを構築した。図5.10に本システムの構成を、図5.11に本システムのアルゴリズムを示す。

ここでは、図5.12に示す小型普通紙複写機の露光・現像系に対して故障診断を行なった例を述べる。本例で用いたパラメータモデルを図5.13に示す。また、現象知識として以下の現象を記述した。

ハロゲンランプの劣化1	ハロゲンランプの劣化2
テーブルガラスの汚れ	ドラム表面の劣化
トナーのメイン・チャージャへの付着	クリーニング・ブレードの劣化
トナー不足	トナー残留
トナー浮遊	用紙劣化
バイアス電圧不良1	バイアス電圧不良2
チャージャ不良1	チャージャ不良2
チャージャ不良3	

図5.14に故障診断実行画面の例を示す。本例は「出力画像が異常に濃い」という故障の症状に対する診断を行なうものであり、図5.14内の Symptoms ウィンドウに示すように、ここではユーザは機能「適正画像濃度」の評価パラメータ Os の値が限界



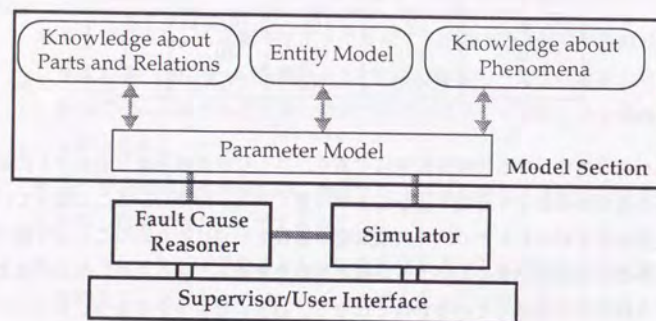


図 5.10: 故障診断システムの構成

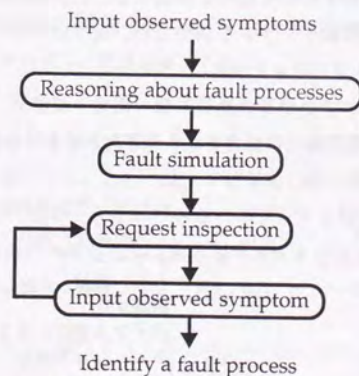


図 5.11: 故障診断システムのアルゴリズム

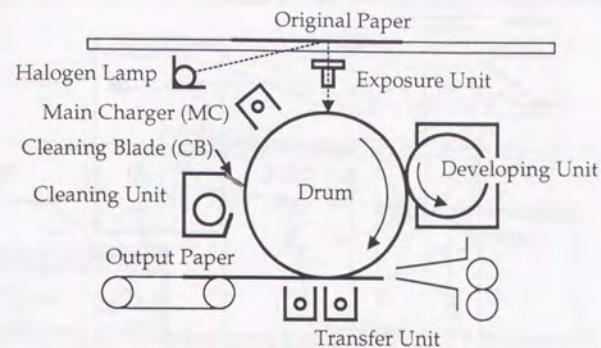


図 5.12: 診断対象の複写機

値を表すランドマーク LimitOs を越えているという形で故障の症状を入力している。いくつかの点検情報を入力することにより、本システムは故障の症状を引き起こしている原因を Fault Process ウィンドウに示す故障発生過程に同定できた。Fault Process ウィンドウに示す故障発生過程は、「クリーニング・ブレード (CB) の劣化」により、「トナーが拡散」し、同時に、ドラム上に「トナーが残留」した結果、「メイン・チャージャ (MC) にトナーが付着」したことを表わしている (図 5.15 参照)。このとき、この故障発生過程により引き起こされた故障のうち、直接に故障の症状の原因になっているものを Faults causing focused Symptoms ウィンドウに、直接は関係していない故障を Other faults ウィンドウに示している。例えば、Faults causing focused Symptoms ウィンドウ内の「MainCharger k is too low」はメイン・チャージャの入力電圧とドラム上への放電電気量の関係を表す動作特性定数  $k$  が異常に低いことを表している。この故障はチャージャ上にトナーが付着するという現象が生じたために発生し、この結果ドラム上へのチャージャの放電量が過大になってしまった。この結果により、ユーザはこの故障症状を修復するためには、例えば、クリーニング・ブレードを交換し、かつ、メイン・チャージャ、ドラム、露光部の清掃を行うことにより廃トナーを取り除く必要があることが分かる。この場合は、単純にメイン・チャージャやドラムの清掃ではこの故障状態を修復することができず、その根本



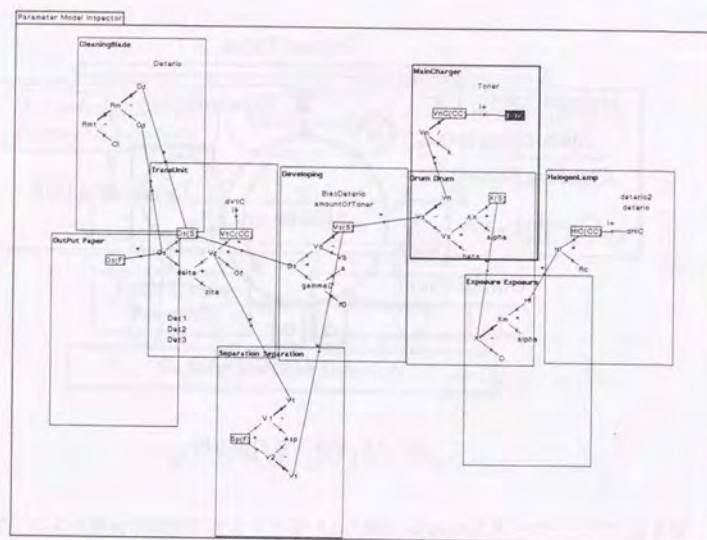


図 5.13: 複写機のパラメータモデル

的な原因を作っているクリーニング・ブレードを交換しなければならないことが分かる点が、本手法の故障発生過程を推論することによる一つの特徴になっている。

本実行例において、故障発生過程候補の導出段階では、例えば、「転写チャージャの劣化」現象により転写効率が過大になり、結果として画像濃度が上昇するという故障発生過程候補も導出するが、この候補は故障シミュレーションを実行することにより、故障モデルのパラメータ値と実際のセンサ値が異なるために排除される。また、本実行例は、正常時にはCBによりトナーはDrumからCleaning Unitへと流れるが、「CBの劣化」現象により、CBとMC間に新たなトナーの流れが生成される。これは一種の構造変更であり、この構造変更により上記の故障発生過程が引き起こされている。この意味で本手法は構造変更を引き起こす故障を診断可能である。

このような故障は実際に発生する。また、トナーの流れに関する構造変更が引き起こされた結果として、画像の劣化を引き起こしたものであり、従来の対象モデルに基づく故障診断手法でも発見が難しい例であると言える。

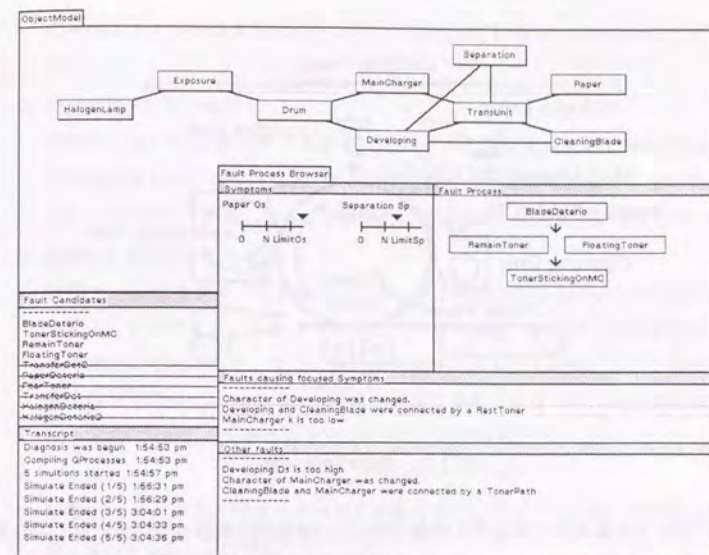


図 5.14: 故障診断システムの実行例

さらに、同一のシステムで自動車のエンジン冷却系に対する故障診断も行ない有効な結果を得た。これにより、本システムが対象モデルを替えることにより様々な対象を扱えるという意味で汎用性を持つことが確認できた。

## 5.2.6 考察

以上のように、実験的な故障診断システムを構築することにより、本稿で提案した故障診断手法が故障、および、故障発生過程を導出可能であることが明らかになった。また、本手法は以下に示す特徴を持つ。

1. 対象のモデルに基づく推論手法を用いている。
2. 単一故障発生過程仮説を用いている。



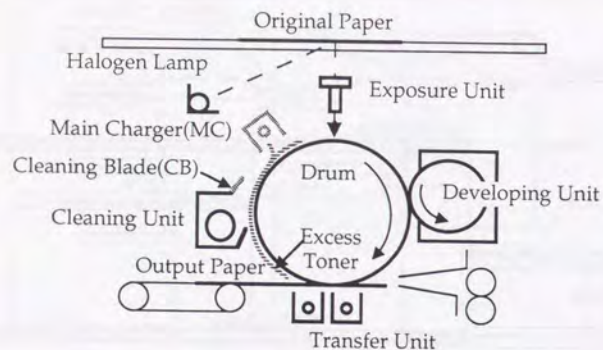


図 5.15: 実際の故障状態

3. 知識を対象機械に依存する対象モデルと対象機械に依存しない原理原則の知識に分けている。
4. 故障原因となる可能性のある物理現象を明示的に記述している。
5. 物理的因果関係を用いて、故障の探索、および、故障シミュレーションを行なう。

これらの特徴により、以下のような利点を持つ。

#### 1. 複数故障の診断

特徴2に基づき、「因果的に関係のある」複数の故障が発生した状態を診断可能である。この意味で、従来の単一故障仮説に基づく故障診断手法よりも故障の診断可能範囲が広いと言える。一方、複数故障仮説に基づいて故障診断を行なう場合は、全ての部品の全ての故障モードの組合せを考えなければならないが、本手法では特徴2、4、5に基づき、因果的に関係のある故障の組合せのみを考えれば良いので、探索空間が複数故障仮説に基づく手法よりも小さくなり、計算が容易になると考えられる。

#### 2. 構造変更を伴う故障の取り扱い

特徴1、4に基づき、本研究では対象モデルとは独立に対象の構造の変化を

引き起こすような現象を記述するので、構造変更を伴う故障を扱うことができる。

#### 3. 物理的無矛盾性の保証

特徴3、4、5に基づき、本手法で導出される故障発生過程は物理的知識として記述されている範囲において、物理的無矛盾性が保証される。そのため、102ページの3.で指摘したような物理的に「奇妙な」故障候補は導出しない。

#### 4. 故障発生過程導出の有効性

特徴2、4に基づき、本手法によりユーザは故障の発生過程や機械の故障状態を知ることができる。これらの情報は、点検、修復、および、改良設計等を行なう場合に有効である。また、自己修復機械においても故障に応じた柔軟な修復を行なうためには、これらの情報が必要になる。すなわち、保全の特徴である不定性に対応した故障診断を実現することができる。

#### 5. 汎用性

特徴1、3により、対象モデルを入れ換えることで様々な対象に対する故障診断を実行することが可能になる。

#### 6. 未発生故障の取り扱い

特徴3に基づき原理原則的知識を用いることにより、本手法では、物理的にあり得る故障発生過程を全て導出する。このため、対象機械では実際に発生したことのない故障の可能性も導出する。ただし、ここで診断可能な未発生故障の範囲は、現象知識を組み合わせで導出可能な範囲であり、知識として与えられていないものを生成する訳ではない。しかし、十分に多量の現象知識を収集することにより、保全の特徴である非繰り返し性に有効に対応できると考えられる。

一方、次のような問題点が明らかになった。

#### 1. 単一故障発生過程仮説の有効範囲

本研究で提案した単一故障発生過程仮説では、因果的に独立な複数故障が発生する状況を扱えない。実際このような故障が発生することもあると考えられるが、その割合は対象機械に依存し明らかではない。ただし、対象系上で位置的に独立な部分で発生する独立な複数故障は、独立な部分問題に分割することにより、本研究の手法を用いて扱うことが可能であると考えられる。



## 2. 計算量の問題

本手法は、物理的に有り得る全ての故障発生過程を導出する。この意味で、本手法は、「非エキスパート」システムである。このため、良く知られている故障に関しては、ヒューリスティクスに基づく手法よりも、計算量が多くなる。また、単一の故障が発生している状態を診断する場合には、単一故障仮説に基づく故障診断手法よりも、計算量が多くなる。また、上記の理由によりあり得る故障発生過程の候補を全て導出してしまうため、場合によってはユーザが本当に発生している故障を判定できないことも考えられる。この問題に対して三つの対応方法が考えられる。

- 第一の方法は、故障診断に成功した場合は、その成功事例をデータベースに記録しておくことにより、次回からの故障診断時にまず、事例に基づく故障診断を行ない、失敗した場合には本手法により故障診断を行なうというある意味で学習を行なう方法。
- 対象モデルを階層的に表現することにより、抽象的なレベルで故障診断を開始し、故障が存在する箇所に関して絞り込みながら詳細なレベルに進むという階層的な故障診断手法。対象モデルを階層的に表現するためには、第4章に述べたFBSダイアグラムが一つの指針を与える。しかし、故障現象という概念を導入したため、故障に関連する部分を限定できないという問題点がある。すなわち、本研究の手法では、故障現象により空間的に近接していない構成要素同士も因果的に関連する可能性があるため、診断対象部分を局所化することが困難である。本研究では、このような階層的故障診断の方法に関して考察を行っている[Wada91a, Oomichi92]。
- 故障診断技術者は、故障の症状を見た時におおまかな「見当」をつけることができ、それにより故障診断を迅速に行なっている。本研究において、故障の症状に応じて、診断を行なう対象モデルの部分を限定したり、利用する原理原則的知識を限定したりすることがエキスパートの「見当をつける」ということに対応していると考えられる。そこで、何らかの知識を収集することにより、診断対象部分、利用知識に優先順位をつけることにより、診断を高速化することが考えられる。

## 3. 知識獲得の方法

本手法では、故障原因を含めて対象上で生起する可能性のある物理現象に関す

る知識を獲得する必要がある。現実の対象に対してこのような知識を収集する方法と知識獲得の実現性に関しては、実際にある程度の量の知識獲得を行ない考察しなければならない。保全の特徴である多様性に対応するためには対象機械と関連する様々な観点からの知識を大量に用意し、利用しなければならない。大量の知識の収集とその利用も今後の課題である。

## 4. 対象表現手法の限界

本研究の対象表現手法はデバイス主義の定性物理と対象モデルの変化としての現象知識の利用という方法を用いている。この方法は汎用性があり有効な表現手法であるが、本研究の対象表現の限界として以下の三点を挙げることができる。

## ● 時間の取り扱い

例えば、複写機は図5.12に示すように、中心にあるドラムが回転することにより、「帯電プロセス」、「露光プロセス」、「現像プロセス」等が順次動作する。一般の機械においても制御プログラム等による動作モードの変更により機械の因果的構造を動的に変化させ異なる動作を実現している。本研究では、このような時間的順序関係を切り落として対象を表現したが、この関係を明示的に表現することにより、より明示的に導出される故障の種類も存在すると考えられる。このため、時間順序関係も含めた対象表現方法が必要になる。この点に関しては吉瀬ら[Kise91]は、複写機専用のモデルベース故障診断システムにおいて、複写プロセスをより明示的に表現する方法を提案している。また、クリーニングブレードがドラムの回転によりビビリ振動を発生するような挙動を明示的に表現するためにも時間的表現が必要になり、本研究では表現が困難な例である。

## ● 空間の取り扱い

第4.2.2項でも述べたように定性物理において空間的、幾何的情報を取り扱うのは困難である。このため、複写機の例でいえば出力画像のムラのように空間的情報を扱う必要のある故障を表現困難である。

## ● 表現レベルの問題

本研究の基本となる表現レベルはパラメータとその関係、および、現象を記述するための表5.4、表5.5に示した述語によって決められている。これらの述語の種類を増やすことにより表現力を高めることは可能であるが、



パラメータ関係で表現できないような曖昧なものの性質は扱えない。実際に知識を収集、利用することにより、本表現手法の限界を明確にすることができると考えられる。

#### 5. 定性物理による曖昧さ

本研究は定性物理に基づいているため、定性的な値のみを用いる。それゆえ、本手法では必ずしも故障候補を一つに同定できるとは限らない。故障候補をより絞り込む方法として、数値計算を部分的に導入することや故障を弁別可能なテストを対象モデルに基づき生成し、対象機械上でテストを実行し情報を収集すること等が考えられる。

### 5.3 ファジィ定性物理を用いた故障診断手法

前節では、定性物理を用いた故障診断手法を述べたが、第7章でも述べるように、この故障診断手法を現実の機器に対して適用した場合、機器から得られる実測値の定性化や劣化現象の取り扱いに問題が生じる。

そこで本節では、定性物理において、現実の機器の状態をより柔軟に表現し、かつ、半定量的な表現を導入することで劣化現象を推論可能にする枠組として、定性物理にファジィ理論を導入したファジィ定性物理とこれを用いた故障診断の手法を提案する。ただし、本節で述べる故障診断手法においても対象表現は第5.2.2項で述べた枠組を用い、第5.3.1項で述べるように量の概念をファジィ化することにする。

#### 5.3.1 ファジィ定性物理

##### (1) 定性物理へのファジィ理論導入の必要性

定性物理を用いて現実の機器に対する故障診断を行なう場合以下の二つの問題点を挙げることができる。

##### 1. 現実世界と定性的記号世界との対応付け

定性物理では変数の値は物理的意味を反映するランドマークにより記号化される。このため、現実の機器の状態を定性物理の枠組で表現するためには、センサ等による測定値で表現される現実世界と物理的記号世界を対応付けなければならない。例えば、「水の沸点」＝「100℃」という対応付けがなされる。逆に、定性的モデルの正しさがこの対応付けに依存する。しかし、この対応付けは測定精度、モデル化の詳細度、周囲の環境変化等に依存し、必ずしも固定的に決定できない。この問題に対して、伊藤ら [Ito90b] は、学習により動的にランドマークとその実測値の対応付けを行なっている。さらに、ある測定値が一つのランドマークや区間値に含まれると必ずしも言えないことがある。例えば故障診断において、ある変数 A の正常値が 5V であり、現在の測定値が 4.9V であった場合、A の値は正常値よりも小さいとも、正常であるとも断定し難く、その中間的状态であるという表現が最も適当である。

##### 2. 劣化の取り扱い

機器の劣化のように徐々に変化する現象を定性物理で表現することは、定性物



理に大きさの概念が無いため困難である。例えば、パイプに流体を流す場合、パイプの厚み  $t$  は正で、劣化により厚みが減少すると考えると時間微分値  $dt$  は“-”である。このとき、変数  $t$  がいつかは 0 に到達することは定性物理で知ることができるが、それにかかる時間や現在の状態のランドマークに対する近さは知ることができない。この問題に対して、鷲尾ら [Washio87] は定性物理をファジィ化し、ファジィ化された値を定量値と見ることにより、半定量的な情報を利用する方法を提案している。

本研究では、上記二つの問題を解決するためにファジィ理論 [Mizumoto88] を定性物理に導入する。ファジィ理論を用いれば、例えば「背の高い人」のように集合として定義できない概念に対してメンバーシップ関数を導入することにより、集合論を拡張した形で数学的な扱いが可能となり、以下の特徴を得ることができる。

1. 上記 1. に関して、物理的概念を反映する定性値は変数値の空間において集合として明確に定義できないと考えられる。すなわち、実数直線上で上限と下限を明確に与えることが現実的には意味が無い、もしくは、上限と下限を明確に与えることが困難であるとする。このため、集合で定義された範囲に入るか入らないかという「不確かさ」を扱う確率論よりも、集合として定義できない「曖昧さ」を扱うファジィ理論を導入することがより適当である。これにより現実の世界と定性的物理世界の対応がより柔軟に表現可能となる。
2. 上記 2. に関して、ファジィ理論の導入により変数値のランドマークに対する部分的な距離概念を表現できるので、劣化現象を推論可能になる。

## (2) ファジィ定性値

本研究では、変数の値を定性値とその定性値に実際の値が属する割合を表すグレードの組として表現する。これをファジィ定性値と呼ぶことにする。式 (5.28) に変数  $X_f$  のファジィ定性値の定義を示す。ただし、 $[x_i]$  は定性値を表し、 $g_i (0 \leq g_i \leq 1)$  はグレードを表すものとする。式 (5.28) は、 $X_f$  が定性値  $[x_1]$  を取るグレード  $g_1$ 、定性値  $[x_2]$  を取るグレード  $g_2 \dots$  という値を持つことを示している。

$$X_f = g_1/[x_1] + g_2/[x_2] + \dots = \sum g_i/[x_i] \quad (5.28)$$

また、定量的な測定値をファジィ定性値に変換するためにメンバーシップ関数を用いて変数の量空間を図 5.16 のように記述する。図 5.16 において、 $N$  は正常値を表すラ

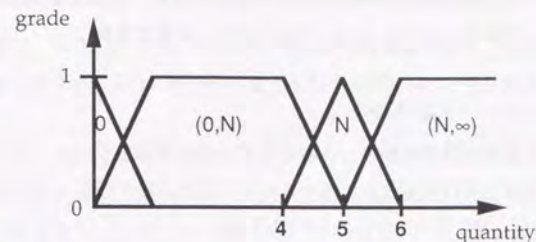


図 5.16: ファジィ量空間の例

ンドマークを表している。例えば測定値 5.5 は、 $0/0 + 0/(0, N) + 0.5/N + 0.5/(N, \infty)$ 、すなわち、ランドマーク  $N$  に対応するか、区間値  $(N, \infty)$  に対応するか決められない中間的な状態であると表現される。

また、前節で述べたように本研究では変数の状態は値と微分値の組により表現される。ここでは、値に関しては上記のようにファジィ定性値を用いるが、微分値は従来と同様に +、0、- という定性値を用いることにする。これは、一般に変数の微分値の符号に関しては曖昧な表現を必要としないと考えられるからである。微分値に関するファジィ定性化も可能であるが、その場合は計算量が組合せ的に大きくなる。

## (3) ファジィ定性値に対する演算則

ファジィ定性値同士の四則演算  $Z_f = X_f * Y_f (* \in \{+, -, \times, \div\})$  は、ファジィ理論の拡張原理に基づき式 (5.29) のように定義できる。

$$\begin{aligned} Z_f &= X_f * Y_f \\ &= \sum_i g_i/[x_i] * \sum_j h_j/[y_j] \\ &= \sum_{i,j} \min(g_i, h_j)/([x_i] * [y_j]) \end{aligned} \quad (5.29)$$



ここで、 $[x_i] * [y_i]$  は従来の定性演算則により計算できる。例えば以下の式 (5.30) ~ (5.32) を計算した場合、 $Z_f$  は、式 (5.33) のように、計算できる<sup>3</sup>。

$$X_f = 0.8/N + 0.2/(N, \infty) \quad (5.30)$$

$$Y_f = 0.7/(0, N) + 0.3/N \quad (5.31)$$

$$(Z_f, N) = ((X_f, N), (Y_f, N)) \quad (5.32)$$

$$\begin{aligned} Z_f &= X_f \times Y_f \\ &= (0.8/N + 0.2/(N, \infty)) \times (0.7/(0, N) + 0.3/N) \\ &= 0.7/(N \times (0, N)) + 0.3/(N \times N) + 0.2/((N, \infty) \times (0, N)) \\ &\quad + 0.2/((N, \infty) \times N) \\ &= 0.7/(0, N) + 0.3/N + 0.2/(0, N) + 0.2/N + 0.2/(N, \infty) + 0.2/(N, \infty) \\ &= 0.7/(0, N) + 0.3/N + 0.2/(N, \infty) \end{aligned} \quad (5.33)$$

さらにここでは、パラメータ値のグレードの合計が1になるように正規化することにする。これは、上記  $N$  や  $(0, N)$  等の定性値は本質的に排反であるので  $N$  に対するグレードが1でかつ  $(0, N)$  に対するグレードが1であるという状態は考えられないためであり、また、グレードの合計をどの変数の値も等しくするためである。このため、求める値は、 $Z_f = 0.58/(0, N) + 0.25/N + 0.17/(N, \infty)$  となる。

二変数関係に関しては、定性比例、反比例関係 ( $Q+$ 、 $Q-$ ) は表 5.3 に示したように微分値同士の関係であるのでファジィ定性化により影響を受けず、前節の計算方法がそのまま適用可能である。定性微分関係に関しては、単位時間を仮定することにより、式 (5.34) に示すような微分関係を式 (5.35) に示す差分関係に変換し、上記の四則演算則より計算する。式 (5.35) において、 $X_i$  は、ある時点  $t_i$  における値であり、 $X_{i+1}$  は、 $t_i$  から単位時間経過後の時点  $t_{i+1}$  における値である。

$$\frac{dX}{dt} = V \quad (5.34)$$

$$X_{i+1} = X_i + V \quad (5.35)$$

<sup>3</sup>式 (5.32) は、109ページで述べた量空間の間の関係を示す。

#### (4) 挙動推論の方法

挙動推論は、第 5.2.3 項で述べた方法と基本的には同様である。しかし、定性値をファジィ化したことにより、幾つかの変更を行なった。以下、その変更点について述べる。

まず第一に、本節では単位時間の概念を導入して微分関係を計算するため、前節で述べた QSIM に基づく挙動推論の方法と本節の挙動推論の方法では、時間の概念が異なるものとなる。すなわち、前節の QSIM に基づく手法では、時間は時点 (time point) と時間 (time interval) を交互に繰り返すものとして表現したが、本節の手法では、単位時間によって等間隔に刻まれた時点の連続として表現される。すなわち、全ての状態はある瞬間の状態になる。

第二に、状態解析時に現象の発火条件の判定方法の変更が挙げられる。QSIM に基づく方法では、発火条件を真偽の二値で判定できたが、ファジィ定性値を導入することで発火条件は 0 ~ 1 の間の値での一致度により判定される。ある現象の発火条件  $c$  は表 5.4 に示した述語  $p$  の論理和、積により一般に式 (5.36) のように書ける。式 (5.36) がある対象モデルの状態に対して発火条件を満たす度合を表す一致度  $C_p$  は、ファジィ論理 (Fuzzy Logic) [Mizumoto88] を適用することにより論理和、論理積はそれぞれグレードの最大値、最小値を取る計算に置換することができ、式 (5.37) のように計算できる。

$$c = (p_{11} \wedge p_{12} \wedge \cdots \wedge p_{1k_1}) \vee \cdots \vee (p_{n1} \wedge p_{n2} \wedge \cdots \wedge p_{nk_n}) \quad (5.36)$$

$$C_p = \max_i (\min_j C(p_{ij})) \quad (5.37)$$

ここで、 $C(p_{ij})$  は、述語  $p_{ij}$  に対する対象モデルの一致度を表しており、量的関係以外は述語が真ならば 1 を偽ならば 0 を与える。量的関係は、条件パラメータのファジィ定性値とそのパラメータが現在取っているファジィ定性値の二値間の一致度  $C(v_1, v_2)$  として計算できる。例えば、述語  $\langle p_1, N \rangle$  に対して実際の  $p_1$  の値が  $0.2/N + 0.8/(N, \infty)$  のとき、 $C(v_1, v_2)$  は以下のように計算できる。

$$\begin{aligned} C(v_1, v_2) &= C(1/(N, \infty), 0.2/N + 0.8/(N, \infty)) \\ &= \max(\min(0, 0.2)/N + \min(1, 0.8)/(N, \infty)) \\ &= \max(0/N + 0.8/(N, \infty)) \\ &= 0.8 \end{aligned} \quad (5.38)$$



このようにして、ある現象がある対象モデルの状態に対して発火する度合  $C_p$  が求められた場合に、現象を発火させる方法に二つの方法が考えられる。

1. ある閾値を設定し、 $C_p$  が閾値以上である場合は、現象を発火させる方法。
  2.  $C_p$  が0でない場合には発火させる場合と発火させない場合に状態を分岐させ、それぞれの分岐にグレード  $C_p$ 、および、 $1 - C_p$  を割り当てる方法。
- 2.の方法は、あり得る全ての場合を導出可能であるが、特に、複数の現象の同時発生を考えた場合計算量が膨大になる。そこで本研究では、1.の方法を採用し、 $C_p >= 0.5$ の時に、現象を発火させることにする。

第三に、状態決定の推論方法を変更する必要がある。すなわち、QSIMに基づく手法では、各パラメータに対して可能な全ての状態遷移を導出しなければならないが、ファジィ定性値に対して可能な状態遷移は無数にあるためQSIMの方法を利用することができない。そこで本研究では、プロバゲーション法[de Kleer84a]を用いて以下の方法により状態決定を行なう。

1. 定性微分関係に関して単位時間経過後の変数値を(3)で述べた方法により決定する。
2. 四則演算関係に関して変数の二項が決まっていれば残りの一項を(3)で述べた方法により決定する。また、微分関係以外の二項関係に関しても一項が決まっていれば、残りの一項を決定する。
3. 2.を未決定な変数が無くなるまで繰り返す。
4. 全拘束関係に関して無矛盾性を検証する。拘束関係に関して矛盾する変数値を持つものがあれば、状態決定不能とする。

以上の方法による挙動推論は従来の定性物理の推論と異なり、推論結果として解の分岐を生じることはない。すなわち、定性演算則の曖昧さによる値の分岐は本表現では、各定性値に対するグレードの分散として表現される。このため、推論結果は従来の解の分岐を全て重ね合わせた一つの解となる。

### 5.3.2 知識表現

本節の手法は、第5.2.2項で述べた知識表現の枠組を用いる。すなわち、以下の三種類の知識を用いて対象を表現する。

### 5.3. ファジィ定性物理を用いた故障診断手法

1. 対象の構造に関する対象依存の知識(対象モデル)
2. 部品、接続関係の物理的性質に関する知識
3. 物理現象に関する知識

ここで、本節では部品の劣化をシミュレートすることを試みる。そこで、第5.3.3節で述べる部品の劣化に関する知識を2.の知識の中に記述する。

### 5.3.3 劣化シミュレーション

#### (1) 劣化のメカニズム

本故障診断では、対象系の劣化情報を故障候補の順位付けに利用する。ここでは高田[Takada77]に従い、機械の劣化は環境からの影響である環境ストレスや機械動作時の使用状況が引き起こす作動ストレスが機械に影響することにより発生すると考える(図5.17参照)。そこで、この劣化現象をシミュレートするためにショートタームとロングタームという二つの異なる時間尺度を導入する。ショートタームは機械が要求機能を発揮するために行なう一サイクルの動作を行なう程度の時間であり、図5.17で言えば作動メカニズムによる機械のシミュレーションを行なうための時間尺度である。また、ロングタームは、機械の部品が劣化を引き起こすのに十分な程度に長い、例えば月単位、年単位といった時間であり、図5.17の劣化・損傷メカニズムに基づく機械の変化をシミュレーションするための時間尺度である。本研究では、これら二つの時間尺度に基づく機械のシミュレーションをそれぞれロングタームシミュレーション(LSIM, Long term SIMulation)、ショートタームシミュレーション(SSIM, Short term SIMulation)と呼ぶことにする。

本故障診断では、以上の劣化メカニズムに基づき、図5.18に示すように、劣化した対象機械の状態を予測するために一定期間毎に劣化シミュレーションを行ない、劣化後の状態を表す対象モデル(劣化モデル)を導出する。故障診断は劣化モデルを用いて行なうことにする。

#### (2) 劣化量の表現

本研究では、図5.17に示す劣化のメカニズムを各部品単位で以下のようにモデル化する。ここで部品単位で劣化を考えるのは、劣化は各部品種類で特有のものであると考えるからである。一般に部品の劣化は、劣化量を  $X$ 、作動ストレス、環境ストレス



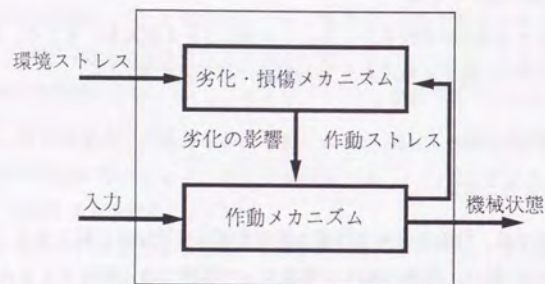


図 5.17: 劣化のメカニズム [Takada77]

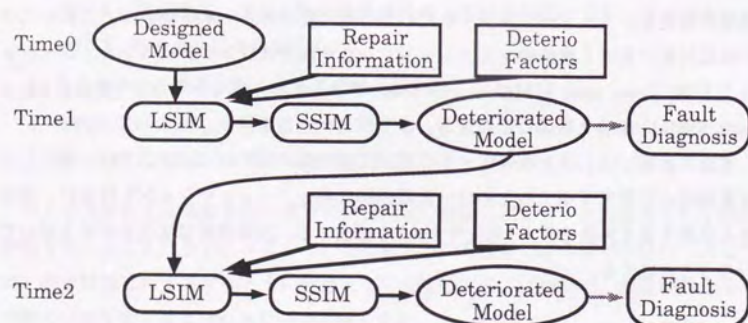


図 5.18: 故障診断の枠組

より影響を受ける劣化速度  $\frac{dX}{dt}$  として、劣化速度の全動作時間に関する時間積分、すなわち、

$$X = \int \frac{dX}{dt} dt \quad (5.39)$$

と表される。図 5.19 における灰色の曲線が劣化を表す曲線である。本研究では、(1)で述べたロングタームの時間尺度の導入により、図 5.19 のようにこの劣化曲線を近似して表現する。この近似により、前回の劣化シミュレーションの結果得られた劣化

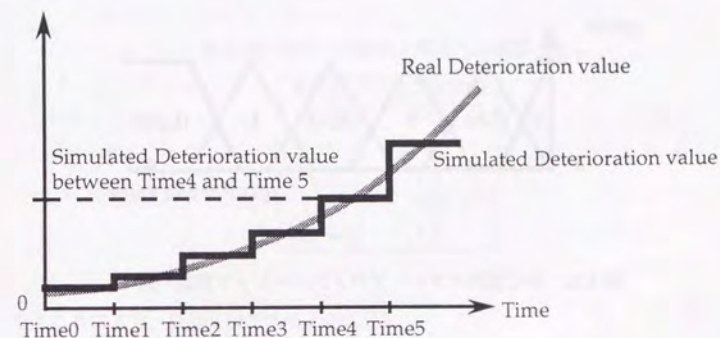


図 5.19: 劣化曲線の近似

量を  $X_{i-1}$ 、前回から今回までの機械のショートタームにおける使用状態から決まる劣化速度を  $\frac{dX}{dt}$ 、および、単位時間を  $\delta t$  とすると、現在の劣化量  $X_i$  は次式により求めることができる。

$$X_i = X_{i-1} + \frac{dX}{dt} \times \delta t \quad (5.40)$$

また、劣化速度  $\frac{dX}{dt}$  は、各部品種類によってモデル化されるが、例えば、ギアの場合は次式のようにモデル化できる。

$$\frac{dX}{dt} = \alpha T + \beta Tor + \gamma V \quad (5.41)$$

ここで、 $T$ 、 $Tor$ 、 $V$  を劣化要因パラメータ、 $\alpha$ 、 $\beta$ 、 $\gamma$  を劣化要因係数と呼ぶ。すなわち、ギアの劣化速度は使用時間  $T$ 、負荷トルク  $Tor$ 、および、回転速度  $V$  に関係する。このとき、各劣化要因パラメータには、図 5.20 に示すようなメンバーシッ



ブ関数が記述される。このメンバーシップ関数の横軸は、時間  $\delta t$  において、対象機械の動作時の定量的な平均値であり、定性値  $S$  は、その部品にとっての標準的使用状態値を示す。このメンバーシップ関数により対象系における対象部品の使用状態がその部品にとって許容範囲か、限界に近い状態で使用しているかが分かる。また、劣化要因係数には各劣化要因パラメータの劣化速度に対する影響度に応じて、 $S$ 、 $M$ 、 $L$  等の定性値を与える。

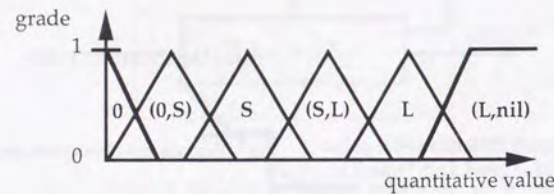


図 5.20: 劣化要因パラメータのメンバーシップ関数の例

### (3) 劣化の影響の表現

以上により表現された劣化量がショートタームにおける部品の性質に与える影響として以下の二つを考えることにする。

#### 1. 部品の特性変化

例えば、電球は劣化すると抵抗値が大きくなるように部品の属性値は劣化により変化する。そこで表 5.9 に示すような劣化量と特性値の関係を表す知識を記述する。

#### 2. 故障現象の発生確率

例えば、電球は劣化が進むと切れやすくなるといったように、劣化が故障現象を引き起こすことが考えられる。そこで、劣化していない対象部品上での各現象の発生度合を 1 とし、表 5.10 に示すように劣化量と現象発生度合の関係を知識として記述する。

以上のようにして、部品の劣化メカニズムのモデル化を行なう。実際の影響要因パラメータとそのメンバーシップ関数、劣化要因係数は各部品に対する耐久試験等を行

表 5.9: 劣化の特性値への影響の例

劣化量 X	抵抗値 R
$(0, S)$	$1/N + 0/(N, \infty)$
$(S)$	$1/N + 0/(N, \infty)$
$(S, L)$	$0.9/N + 0.1/(N, \infty)$
$(L)$	$0.8/N + 0.2/(N, \infty)$
$(L, \infty)$	$0.5/N + 0.5/(N, \infty)$

表 5.10: 劣化の現象発生度合への影響の例

劣化量	電球切断現象
$(0, S)$	1
$(S)$	1
$(S, L)$	1.1
$(L)$	1.3
$(L, \infty)$	1.5

なうことにより、設定可能であると考える。

### (4) 劣化シミュレーション

以上で述べた劣化のモデル化に基づき、ロングタームの一単位時間の劣化をシミュレートするのが劣化シミュレーションである。すなわち、ある劣化モデルから一ロングターム後の劣化モデルを導出する。このシミュレーションは図 5.18 に示すようにロングタームシミュレーションとショートタームシミュレーションから成る。

#### 1. ロングタームシミュレーション (LSIM)

(1) で述べた劣化メカニズムを利用して、各部品毎にロングタームシミュレーションを以下のように実行する。

(a) 入力として、対象部品の現在の劣化量、時間  $\delta t$  間の定量的な平均使用状態、および、部品の交換情報を与える。時間  $\delta t$  の間に部品が交換された場合は、対象部品の劣化量を 0 とする。

(b) (2) で述べた方法により、劣化量を計算する。



- (c) (b) で求めた劣化量から表 5.9、表 5.10 を利用して特性変化、現象発生度合の変化を求める。例えば、劣化量が式 (5.42) で与えられる時、表 5.9 を利用して以下のように特性変化を求める。

$$X = 0.3/S + 0.7/(S, L) \quad (5.42)$$

$$\begin{aligned} R &= (1/N + 0/(N, \infty)) \times 0.3 + (0.9/N + 0.1/(N, \infty)) \times 0.7 \\ &= (0.3 + 0.63)/N + (0 + 0.07)/(N, \infty) \\ &= 0.93/N + 0.07/(N, \infty) \end{aligned} \quad (5.43)$$

## 2. ショートタームシミュレーション (SSIM)

LSIM で導出された各部品の属性値の変化から、対象系全体の劣化後の状態(劣化モデル)をパラメータモデルを用いて第 5.3.1 項の (4) で述べた方法により推論する。これにより、劣化から直接引き起こされる状態変化や故障を推論することができる。

以上の劣化シミュレーションを実行することにより、ある対象機械のある時点における各部品の相対的な劣化の進行度合を評価することができる。

## 5.3.4 故障診断の方法

前項で得られる対象系全体の劣化モデルをもとに故障診断を行なう。故障診断の方法は第 5.2 節で述べた方法に基づいて実現されている。

### 1. 機能の比較

対象機械から得られるファジィ定性化された測定値をもとに各機能の発現状態を判定する。判定条件は、137 ページで述べた現象の発火条件に対する対象状態の一致度  $C_p$  と同様の計算を行なうことにより、各機能知識に記述された発現条件に対する一致度  $C_f$  を求め、 $C_f \geq 0.5$  の時、その機能は正常であるとする。

### 2. 故障発生過程候補の導出

故障発生過程候補の導出は、第 5.2.4 項で述べた故障原因候補の導出、および、故障発生過程候補の導出方法をそのまま用いる。

### 3. 故障シミュレーション

2. で導出された各故障発生過程候補に対して、その故障が発生した場合の対象

系全体の状態(故障モデル)を入力された劣化モデルを用いて SSIM により導出する。

### 4. 一致度の計算

3. で導出された各故障候補に対する故障モデルとファジィ定性化された測定値を比較することにより 137 ページで述べた方法によりパラメータの一致度  $C(v_1, v_2)$  を計算する。故障モデルと測定値との一致度は、各測定パラメータの一致度の二乗平均として与えることにする。ただし、劣化のみによって故障が発生する場合も考えられるので、初期入力された劣化モデルも故障モデルの候補に含めて一致度を計算し、一致度が 0 の故障候補を除外する。

### 5. 故障候補の順位付け

4. で求められた一致度に劣化シミュレーションで導出された故障現象発生度合を乗じた結果をもとに故障候補を順位付けする。このとき、順位付けに使用した数値自体は定量的な意味を持たず、各候補間の相対的な優先順位を表している。その理由は、故障モデルは定性値の分岐による値の分散を含んでいるからである。

## 5.3.5 インプリメンテーション

本節で述べた故障診断手法の有効性を検証するため、Smalltalk-80 [Goldberg87] を用いて実験的な故障診断システムを構築した。システムの構成を図 5.21 に、推論アルゴリズムを図 5.22 に示す。

前節と同様の複写機を対象として故障診断を行なった結果を図 5.23 に示す。ここでは、対象の複写機の仮想的な劣化データを利用し、劣化シミュレーションを行なった。このとき、センサ値の変化や仮想的な劣化の進行にしたがって、図 5.23 に示した故障候補の順位が入れ替わり、実際の機械と比較して正しいと思われる結果を得た。図の例は、「画像濃度が異常に濃い」という故障の症状に対して、「ハロゲンランプの出力低下」現象が故障原因として導出されている。この現象はハロゲンランプの劣化の進行に従って発生度合が高まる現象である。ただしここでは、簡単のため第 5.2 節で示したような互いに因果的関係を持つ故障現象は記述しなかった。この意味で、この実行例は単一故障仮説によるものと同様である。図 5.23 の右側のウィンドウは測定可能なパラメータのメンバーシップ関数と本実行例における測定値が示されている。



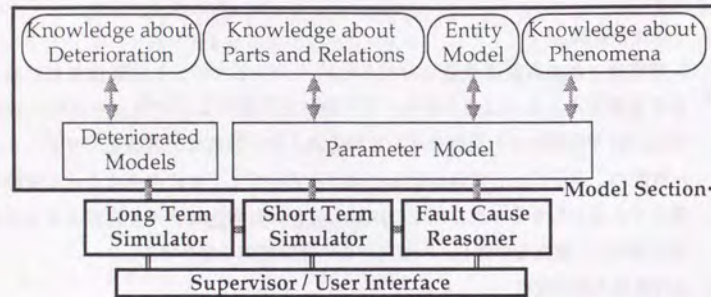


図 5.21: ファジィ故障診断システムの構成

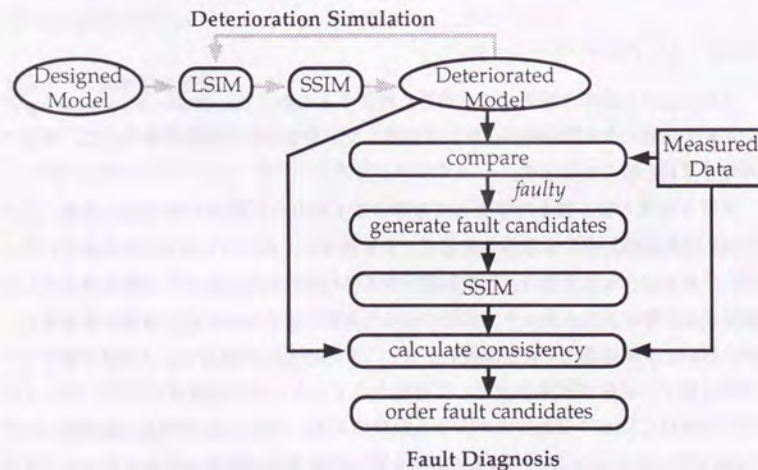


図 5.22: ファジィ故障診断システムのアルゴリズム

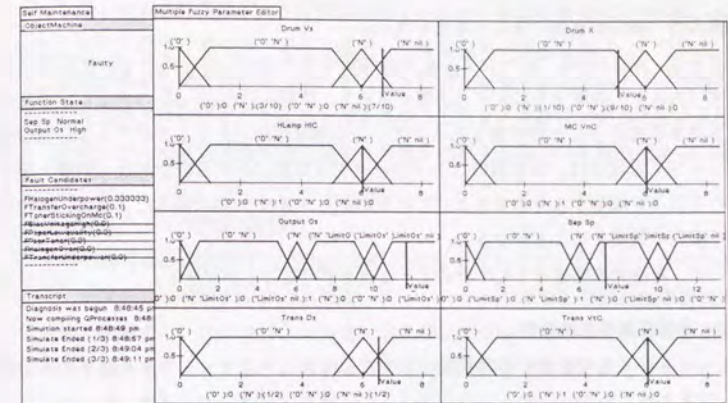


図 5.23: ファジィ故障診断の実行例

## 5.3.6 考察

本節では、前節で述べた故障診断手法に基づき、定性値表現の柔軟化、劣化の取り扱いを行なうためにファジィ定性物理を提案した。ここでは、このファジィ定性物理とこれに基づく故障診断手法の有効性について議論する。

定性物理にファジィ理論を導入した研究としては鷲尾ら [Washio87] の方法がある。この方法では、変数の定性値として  $+$ 、 $0$ 、 $-$  を取り、量空間を  $-1$  から  $1$  の範囲で正規化、定量化して表現している。このため、各変数値は  $-1$  から  $1$  の範囲の実数値に一対一対応させることができる。本節で述べたファジィ定性物理では、変数の量空間として任意のランダムマークを取ることができる。また、ファジィ定性値は必ずしも実数値と一対一対応していないと考え、第 5.3.1 項で述べたような値の表現を行なう。これは例えば、変数  $A$  が定性演算則の分岐により  $0.33/(0, N) + 0.33/N + 0.33/(N, \infty)$  といった値を取り得るからである。この値は特定の実数値と対応しているとは考え難く、 $(0, N)$ 、 $N$ 、 $(N, \infty)$  のいずれかの値であるがどの値を取るかは分からないという状態であると考えられる。これらの特徴により、本手法では従来の定性物理の枠組を変えずにファジィ理論を導入可能で、現実世界と物理的世界のより柔軟な対応表現が可能である。また、保全作業における曖昧さを表現するためには本



研究の値の表現が有効であると考えられる。

さらに、第5.3.1項で述べたように劣化現象を扱うためには、何らかの劣化度合を示す「大きさ」の概念を導入する必要がある。鷺尾ら[Washio87]の方法は定量的な大きさを直接扱うことが可能であるので、この問題に対応することができる。本研究のファジィ定性値は、現在値のランドマークや区間値に対する概念的な「距離」をグレードにより表現するものである。これを利用し、第5.3.3項で述べたように、劣化の進行度合を表現することができた。

一方、本研究の問題点として次のような点を挙げることができる。

### 1. 挙動推論手法の問題

ファジィ定性物理の挙動推論方法には以下のようなヒューリスティックスが導入されている。

- 挙動推論の方法としてプロパゲーション法を利用しているので、パラメータ間の関係がループになる場合には、推論の順序により求まる値が多少変化する可能性がある。
- 現象を発火させる条件を  $C_p \geq 0.5$  とした。これは、グレードが0.5より大きいと言うことは、生起すると考えられるというヒューリスティックスに基づくものである。しかし、全ての有り得る可能性を求めるという立場に立つのなら、グレードが小さい現象についても分岐を生じさせ、その可能性を導出するべきである。ただし、この場合計算量が膨大になり、組合せの爆発を引き起こす可能性が大きい。また、0.5という判定基準の取り方にも問題がある。なぜならば、定性推論の解の分岐により各パラメータ値のグレードは分散してしまうため、必ずしも正しく現象を引き起こさない場合も考えられる。しかし、正しい判断基準を決定する方法は明らかではない。

これらのヒューリスティックスは現在のところ大きな問題とはなっていないが、複雑な対象を扱う場合には問題を引き起こす可能性がある。

### 2. 劣化に関する知識獲得

本研究の劣化の取り扱いには、第5.3.3項で示したように各部品に関する多くの劣化データ、すなわち、劣化要因パラメータの選定とメンバーシップ関数の決定、劣化要因係数の設定、劣化影響知識の記述が必要になる。ベアリングのよ

うにこのような劣化データが既に収集されている部品もあるが、一般的には、このような形で劣化データを収集することは今後の課題である。このため、本研究で提案した劣化の取り扱いが、本当に劣化現象を扱えるかという問題に対して現時点では答を出すことはできない。ただし、本研究で提案したような劣化データの記述形式は劣化に関する情報を収集する場合にも有効であると考えられる。

### 3. メンバーシップ関数の記述

本手法では、測定値を定性物理世界に変換するためにパラメータのメンバーシップ関数を利用する。このため、メンバーシップ関数の形態が故障診断の性能に影響する。ここでは、正常時、および、異常時における対象機械の状態測定を繰り返すことにより、このようなメンバーシップ関数を決定できると考えている。また、測定結果から動的にメンバーシップ関数を変形させるような一種の学習機構を付加することも考えられる[Shimomura92]。



#### 5.4 第5章のまとめ

本章では、第5.1節において従来の故障診断手法を概観し、保全の特徴である非繰り返し性、不定性、および、多様性に対応するためにはモデルベース推論、特に定性物理に基づく故障診断手法が有効であることを述べた。しかし、モデルベース推論を用いた手法にも問題点があることを指摘し、複数の故障を扱え、不定性に対応するために構造変更を引き起こす故障を扱い、さらに、故障後の物理的状態を知る必要があることを明らかにした。

上記の要求項目に対応するために、第5.2節で単一故障発生過程仮説を提案し、この仮説を用いた故障診断手法を提案した。本手法は、対象モデルに基づく故障診断手法の特徴である、汎用性、未発生故障導出の可能性、及び、知識獲得の容易性を持つ。さらに、単一故障発生過程仮説を導入することにより、構造変化を伴う故障、因果的に関係する複数の故障を物理的な原理原則に基づいて診断可能であるという特徴を持つ。本手法により、非繰り返し性、不定性には対応可能となったが、多様性への対応は知識獲得の問題として残されている。

さらに第5.3節では、定性物理を用いた故障診断手法を現実の機器に適用した場合に問題となる現実世界と計算機上の定性物理の世界の対応を柔軟に表現し、さらに、半定量的な表現を導入することで劣化現象を推論可能にする枠組として、定性物理にファジィ理論を導入したファジィ定性物理を提案した。また、ファジィ定性物理を用いた故障診断の手法を提案した。測定値と定性値との間の表現の柔軟性に関しては本手法でその有効性を検証できたが、劣化の取り扱いの有効性は今後実際のデータを収集して検証しなければならない。

本研究では、本章で提案した手法により、自己修復機械システムの知的判断部の故障診断機能を実現する。その実際の適用例については、第7章、および、第8章で述べることにする。

## 第6章

### 修復計画の自動化





本章では、自己修復機械の修復計画機能を実現する方法を提案する。まず、従来の修復自動化、支援手法を概観し、これらの手法の問題点を整理する。次にこの考察に基づき、第6.2節において制御型自己修復機械のための修復計画手法を述べる。また、第6.3節で機能冗長型自己修復機械のための修復計画手法を提案する。

## 6.1 従来の修復計画手法と問題点

本研究では佐藤 [Sato90] に従い、修復作業を図6.1 (再掲) に示すように三段階に分ける。

正常時、故障時の機械に関する情報

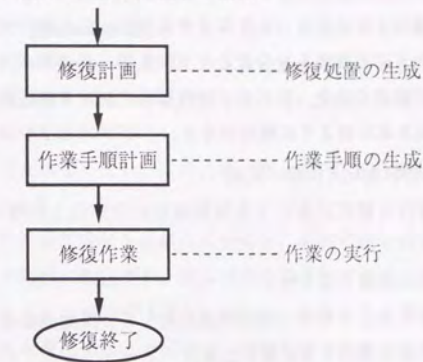


図6.1: 修復のタスク (再掲)[Sato90]

図6.1において、「修復計画」は、設計情報から得られる対象機械の正常時の情報、および、故障診断作業の結果得られた故障時の情報から故障を修復するための目標とその処置を決定する知的判断作業である。ここでいう、修復処置とは「部品 A を交換する」、「機能 B が発現するようにボリューム C を調整する」等である。「作業手順計画」では、修復計画で導出された修復処置を実現するための動作手順を生成する。例えば、「部品 A を交換する」ためには、「外カバーを外し、部品 A を固定しているボルトを外し、…」というように実際の作業手順を決定する。「修復作業」は、作業手順計画により生成された手順にしたがって実際の動作を行ない、修復を実現する。

本章では、修復計画作業の自動化を対象とする。作業手順計画を対象としない理由は、本研究で考える自己修復機械における修復作業は実際にはパラメータの制御や構造のスイッチングであるため、部品の取り外しや交換といった複雑な作業手順を必要としないため、作業手順計画の必要性が低いと考えたからである。自己修復機械が手



順として複雑な修復作業を実現可能になった場合は、このような作業手順計画の自動化が必要になると考えられる。浅間 [Asama89] は、対象の接続関係の知識を利用した作業手順計画の自動化と冗長自由度を持つ保全ロボットによる修復作業の自動化手法を提案している。

さて、修復計画の自動化に関しては、第2.2.3項で述べたように、列車に対する応急的修復処置を導出するエキスパートシステム [Shinohara88] や Crow ら [Crow91] の故障診断とシステムの再構成を統合化した FDIR 等いくつかの方法が提案されている。しかし、修復計画の自動化、特に自己修復機械における修復計画の自動化を目的とした場合、問題点を以下のように整理できる。

#### 1. 故障診断の自動化と同様の問題

修復計画の自動化においても故障診断の自動化と同様に以下の点が問題となる。

##### (a) 保全の非繰り返し性

修復したことのない故障状態に対して、既知の修復方法を組み合わせて修復計画を導出する必要がある。

##### (b) 保全の不定性

故障により対象の構造や性質は、一般に変化していると考えられる。このため、対象機械に同一の修復処置を行なった場合にも、故障の種類によって対象系の振舞いが異なり、かえって対象系を破壊してしまうこともある。この問題に対処するためには、修復計画を行なう前に、あらかじめ充分に対象系の故障状態を知る必要があり、また、修復計画も故障状態に応じて決定しなければならない。

##### (c) 物理的正当性

前項とも関係するが、導出された修復計画は正しく対象系を復旧させなければならないし、また、修復の実行によって対象系を破損してはならない。この問題に対処するためには、システムが物理的知識を持ち、修復の物理的実現性や物理的影響を評価する必要がある。松本 [Matsumoto83] の手法では、電力系統の過負荷という項目に関して、修復の影響を評価しているという特徴を持つ。

#### 2. 修復計画特有の問題

故障診断は喪失機能から故障、故障原因を同定する作業であるという意味

で、入力と出力が固定的であるが、修復計画の場合は、その修復目的自体が第3.1.3項で述べた機能トレードオフにより変更される場合がある。この機能トレードオフの取り扱いが、柔軟な修復計画を生成するための一つの鍵となると考えられる。

#### 3. 自己修復機械を目的とした場合の問題

自己修復機械を実現することを目的とした場合、修復計画を自動化する必要がある。このため、故障診断との統合化を行ない、人手による操作を必要とせずには故障診断と修復計画を実現しなければならない。また、修復支援システムのように幾つかの候補を提示するというものとは異なり、実行する修復計画の一つに決定可能でなければならない。このため、何らかの方法で修復計画候補の評価を行なう必要がある。

しかし逆に、自己修復機械を対象にした場合、実現可能な修復操作はあらかじめ決められた有限個に限定され、修復操作を発見的に導出し新たな方法で修復を行なうということは現状では考えられない。すなわち、既知の修復操作の組合せで修復計画を導出することになり、修復可能な問題の範囲は限定されるが、推論の範囲も限定され問題解決が容易になる。

これらの問題点を解決するために、本研究では次のような方法をとることにする。まず、1. に対しては、第5章の故障診断手法で述べたモデルベースト推論手法と知識表現方法を利用することが有効である。そこで本研究では、対象モデル、原理原則的知識、および、挙動シミュレータを故障診断システムと共有することにする。これにより、1. に対応可能であり、かつ、3. に述べた故障診断との統合化を実現することができる。本研究の故障診断 / 修復計画システムの構成を図6.2に示す。

さらにこの構成の上で、2. の機能トレードオフの問題や3. に述べた修復計画の決定性の問題を解決しなければならない。これらの問題については、次節以降の修復計画の方法において議論することにする。



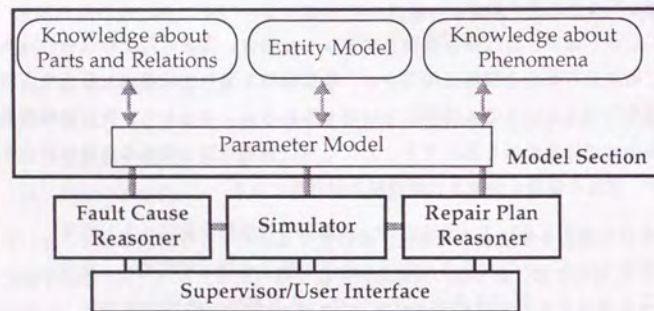


図 6.2: 故障診断 / 修復計画の構成

## 6.2 制御型自己修復機械の修復計画

本節では、制御型自己修復機械のための修復計画の自動化手法について述べる。ただし、本節で述べる修復計画手法はパラメトリックな修復を実現する範囲においては、自己修復機械に限定されない一般的手法である。

### 6.2.1 知識表現

前節で述べた通り、本研究では知識表現の方法として第5.2.2項で述べた枠組を用いることにする。ただし、この枠組に加えて、対象機械で実現可能な修復のための操作知識を対象モデルに記述する。各操作知識は、以下のような形式で記述される。すなわち、Name で表される操作は、対象系において Parameter で示されるパラメータを Direction で示される方向に変化させることができる。以下の例は実際の対象機械では、ハロゲンランプ用のトランスのボリュームを上げる操作に対応している。

#### Control Method

Name:	lamp up
Parameter:	Hl
Direction:	up

### 6.2.2 修復計画の方法

本研究の修復計画は第5章の故障診断手法により導出された故障時の対象モデル(故障モデル)  $m$  から、モデル  $m$  内の喪失機能の集合  $F_l$  を回復する修復操作列  $s_r$  を導出する作業である。ここで、単一の修復操作  $r$  は、次のような形式で導出され、 $s_r$  はこの  $r$  の順序列として導出される。

ある修復操作は Control Method で示される操作知識を実行することで実現され、その時の操作目標は Objectives で示される述語を満たすことである。またこの時、Subeffects で示される条件が対象系上で満たされなくなる可能性がある(副次的影響)ことが記されている。ここで言う副次的影響とは、現在の状態では満たされている機能が修復操作を行なうことにより喪失してしまうことをいう。

#### Repair Method

Control Method:	lamp up
Objectives:	$<(x, limit)$
Subeffects:	$>(y, N)$



修復計画は以下の方法で導出される。

### 1. 修復目標の決定

与えられた故障モデル  $m$  には、全機能  $F$  の内、その故障状態での喪失機能の集合  $F_l \subseteq F$  が記述されている。このとき、機能のトレードオフを考えると、必ずしもこの喪失機能  $F_l$  を全て修復する必要はない。そこで対象機械において必ず発現しなければならない機能 (必要機能) の集合  $F_0 \subseteq F$  を決定し、 $F_n = F_l \cap F_0$  を修復の目標とする。この  $F_0$  が少ないほど修復の実現可能性は高くなると考えられる。

一般には、使用者が現状の価値判断に基づき、この  $F_0$  を決定すると考えられるが、本研究ではあらかじめ必要機能集合  $F_0$  を与えることにする。

以上のようにして求められた  $F_n$  に対して、各機能の発現条件のうち満たされていない述語を抽出することによりパラメータ値に関する修復目標  $P_o$  を求めることができる。逆に、この条件にパラメータ値に関する条件以外の条件が入っていた場合には、制御型の修復計画の範囲では修復不可能であると考えられる。この、パラメータ値の条件の各要素と故障モデルを比較することにより、パラメータを  $a$ 、 $b$ 、 $\dots$ 、パラメータ値を現在値よりも上げる必要がある場合には  $+$ 、下げる場合には  $-$  と表現した場合、以下のようなパラメータ変化目標  $P_n$  を求めることができる。

$$P_n = need(a, +) \wedge need(b, -) \wedge \dots \quad (6.1)$$

例えば、条件  $<(a, limit)$  に対して現在の状態が  $a = (limit, \infty)$  の場合には、 $need(a, -)$  が導出される。

### 2. 修復候補の導出

ここでは、故障モデル  $m$  と修復目標  $P_n$  から以下のように修復操作候補の集合  $R$  を導出する。

- (a) 故障モデルのパラメータモデルを用いて、 $P_n$  に含まれる目標パラメータのベクトル  $P_o$  と操作知識に記述されているパラメータのベクトル  $P_a$  間の因果関係を表すマトリックス  $A$  を以下の形で導出することができる。ここで、この因果マトリックス  $A$  は故障により変化しているので、一般には正常時の因果マトリックス  $A_0$  と異なると考えられる。この  $A$  を故障に応じ

て正しく導出するために、本研究では第5章で述べたモデルベース故障診断により故障時の正しいパラメータモデルを構築する必要がある。

$$P_o = \begin{pmatrix} p_o^1 \\ p_o^2 \\ \vdots \\ p_o^m \end{pmatrix}, P_a = \begin{pmatrix} p_a^1 \\ p_a^2 \\ \vdots \\ p_a^n \end{pmatrix}, A = \begin{pmatrix} a_{11} & a_{12} & \dots & a_{1m} \\ a_{21} & a_{22} & \dots & a_{2m} \\ \dots & \dots & \dots & \dots \\ a_{n1} & a_{n2} & \dots & a_{nm} \end{pmatrix} \quad (6.2)$$

$$P_a = AP_o \quad (6.3)$$

ここで  $A$  の要素  $a_{ij}$  は、パラメータモデルの探索により、 $p_a^i$  と  $p_o^j$  が定性的比例関係にあれば  $+$ 、反比例関係にあれば  $-$ 、関係がなければ  $0$ 、また、関係が不明であれば  $*$  となる。例えば、 $p_a^i$  と  $p_o^j$  の間にフィードバックループを含む場合は  $*$  となる。ただしここでは、 $p_a^i$  と  $p_o^j$  間に  $I+$ 、 $I-$  を含む場合はそれぞれ  $+$ 、 $-$  を与える。例えば図6.3において、 $P_n = need(x, +) \wedge need(y, -)$ 、 $P_o = (x \ y)$ 、 $P_a = (b \ f \ h)$  のとき、 $A$  は次のように求められる<sup>1</sup>。

$$A = \begin{pmatrix} - & 0 \\ + & 0 \\ 0 & - \end{pmatrix} \quad (6.4)$$

以上の表現を用いることにより、修復候補の導出は式 (6.3) において  $P_o$  が  $P_n$  を満たすように  $P_a$  を決めることに置き替わる。

- (b) 本研究では、 $P_o$  に  $P_n$  を代入することにより上記の式 (6.3) を解くことにする。上記の例でいえば  $P_o = (+ \ -)$  を代入することにより、 $P_a = (- \ + \ +)$  を得る。
- (c) ここでは、同時に複数の操作を行なうことは考えない。これは第7.2節で詳しく述べるが、本研究の自己修復機械の範囲では定性的な修復操作を同時に複数実行することが難しいからである。この修復実行部からの要請のため、(a) で求められた解を満たし得る単一操作を導出する。上記の例でいえば、 $(- \ 0 \ 0)$ 、 $(0 \ + \ 0)$ 、 $(0 \ 0 \ +)$  が候補となり、修復目標  $P_o = >(x, 0) \wedge <(y, limit)$  としたとき修復操作候補  $r$  は以下の三種類になる。

<sup>1</sup>ここで、因果の方向性から  $f$  を操作しても  $y$  は変化しない。



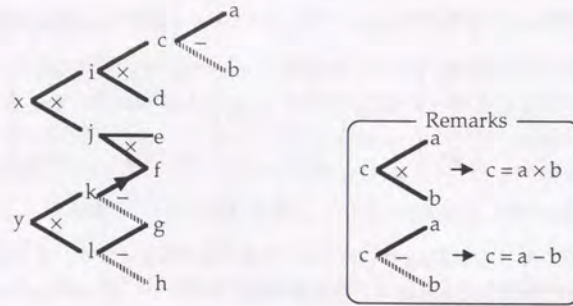


図6.3: パラメータモデルの例

<b>Repair Method 1</b> <b>Control Method:</b> b down <b>Objectives:</b> $> (x, 0)$ <b>Subeffects:</b>	<b>Repair Method 2</b> <b>Control Method:</b> f up <b>Objectives:</b> $> (x, 0)$ <b>Subeffects:</b>
<b>Repair Method 3</b> <b>Control Method:</b> h up <b>Objectives:</b> $< (y, limit)$ <b>Subeffects:</b>	

## 3. 修復シミュレーション

前項で求められた  $R$  の各要素に対して、その修復操作を実行した場合の対象モデルの挙動シミュレーションを行なう。すなわち、ある修復操作  $r \in R$  を故障モデルに対して入力し、修復目標が達成されるか、副次的影響が発生するか、または、定常状態になるまでの状態遷移の履歴をシミュレートする。挙動シミュレーションの方法は第5.2.3項で述べた方法を用いる。以上のようにして導出された履歴において、各分岐の末端の状態が修復操作実行後の対象機械の定常状態であることと見ることができるので、この各末端の状態を表す対象モデルを修復モデル  $m(r)$  と呼ぶことにする。すなわち、 $m(r)$  を評価することにより、修復に成功するか、副次的影響が発生するかがわかることになる。

## 4. 修復操作候補の順位付け

修復シミュレーションで得られた各修復操作候補  $r$  に対する修復モデルの集合  $M(r)$  を評価することにより修復操作候補の順位付けを行なう。

- (a) まず準備として、各修復操作候補  $r \in R$  に対して、各修復モデル  $m(r) \in M(r)$  と修復目標  $P_c$  を比較することにより修復により回復できるパラメータの条件の組  $C_s(m(r))$  を導出する。ここで、 $\bigcup_{m(r) \in M(r)} C_s(m(r))$  は修復候補  $r$  により回復する可能性のある修復目標であり、これを修復操作  $r$  の Objectives に記述する。

また、必要機能の集合  $F_0$  に対する発現条件と各修復モデル、 $P_c$  を比較することにより、修復を始める前の故障状態では満たされているが、修復モデルでは満たされなくなった条件  $C_u(m(r))$  を求める。ここで、 $\bigcup_{m(r) \in M(r)} C_u(m(r))$  は、修復操作によって発現する可能性のある副次的影響であり、修復操作  $r$  の Subeffects に記述する。

- (b) Objectives が空である修復操作候補  $r$  は削除する。これは例えば、対象系に操作を加えることにより何らかの現象が発生し修復目標が実現できない場合である。
- (c) 修復目標を満たす割合の高いものの順に順位付けする。すなわち、各操作候補  $r$  に対して、 $M(r)$  の中で修復目標を満たす修復モデルの個数を  $M(r)$  の個数で割った数が多い順に修復操作候補を順位付けする。
- (d) 成功の割合が同じものに対しては、副次的影響の数が少ない順に修復操作候補を順位付けする。
- (e) 副次的影響が同数の場合は、満足する修復目標の数が多いものを優先する。

## 5. 副次的影響に対する修復

以上のアルゴリズムにより選択された最高順位の修復操作を対象機械に適用することで、機械は修復、あるいは、改善されるはずである。しかし一般的には副次的影響が発生したり、 $P_c$  の中で満足されていない条件が残る場合もある。そこで、満足されていない  $P_c$  の部分と副次的影響から新たに修復目標  $P_c$  とパラメータ変化の目標  $P_u$  を作り直し、修復モデル  $m(r)$  に対し2.以下を再帰的に実行する。ただしこの時、以前の操作と逆の修復操作、例えば、a up に対して a down、は導出ししない。



以上により、図6.4に示すような修復計画  $s_r$  を求めることができる。図6.4において、 $r_i$  は修復操作、 $m_0$  は故障診断で得られる故障モデル、 $m^i(r_i)$  は操作  $r_i$  後の修復モデル、丸で囲まれた修復モデルは修復に成功することを示している。この修復計画に基づき修復を実行する場合、故障モデル  $m_0$  の状態から出発し、修復操作を実行しながら対象系の状態を測定し、現在の状態が図6.4上のどこにあるかを判定する。これにより、機械の状態に応じて修復操作を柔軟に選択することが可能になる。例えば、図6.4上で灰色の線で示されるような軌跡により修復が実現される。

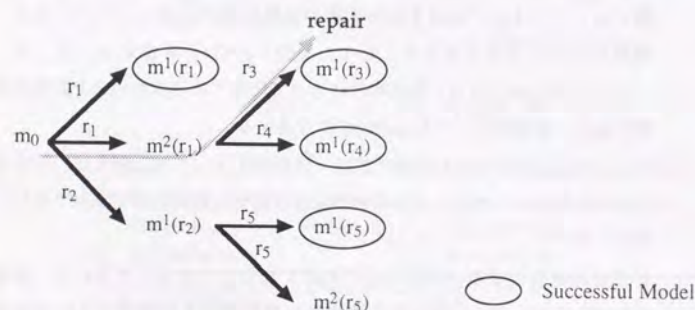


図6.4: 修復計画の例

本節で述べた修復計画手法の有効性を検証するために実験的な修復計画システムを構築したが、そのインプリメンテーション、および、実行例については第7章の制御型自己修復機械の実験例において述べることにする。

### 6.2.3 考察

#### 1. 機能トレードオフの問題

本節で述べた手法では、機能トレードオフに関する情報をあらかじめ与えたが、本来の働きから言えば、自己修復機械においてもユーザとインタラクティブに会話しながら機能トレードオフを行ない、柔軟に修復計画を導出することが必要になってくる。この点は、ユーザインタフェースの問題も含めて今後の課題となる。

#### 2. 構造変更を引き起こす制御型修復

本節の修復計画手法では、与えられた故障時のパラメータモデルの構造を変更せずに、パラメータ値の調整の範囲で修復を行なった。このような修復は、機械の劣化により引き起こされる故障等に有効である。しかし、制御型の修復計画においても、パラメータ制御により正常時には発現しないような現象を対象系上に引き起こし、それにより修復を行なう方法が考えられる。このような修復計画も現象知識の因果関係の探索により、本研究の枠組で導出可能であるが、一般に発見的に導出されたこのような修復計画は有効性が低いと考え、本研究では扱わなかった。

#### 3. 故障モデルが同定できなかった場合

本節で述べた修復計画手法では、単一の故障モデルを対象として修復計画を生成している。しかし、故障診断システムとの統合化を考えた場合、第5章で述べた通り故障モデルを必ず同定できるとは限らず、修復計画を実行できなくなってしまう。実際には、第7章、第8章で述べる故障診断/修復計画システムでは、故障モデルが複数入力された場合にも対処可能になっている。

故障モデルが複数入力された場合に導出される修復計画は図6.4に示した修復計画において、故障モデルが複数存在する形になる。この場合、計算量は組合せ的に大きくなるという問題を含んでいる。計算量を少なくするためには、図6.4を一度に全て計算するのではなく、修復の実行に合わせて必要な部分のみに関してインタラクティブに修復計画を実行することで対応している。このとき、修復操作に対する対象系の状態変化を測定することで、修復計画の中での現在の状態に対応する修復モデルを同定できる。その結果、故障モデルを絞り込むことが可能になる。この修復計画と修復実行をインタラクティブに行なう方法については第7章で詳しく述べる。

#### 4. 修復計画の順位付け

修復計画を順位付けする部分は、修復の実現可能性を高めるため、また、副次的影響により対象系を破壊しないためにも重要であるが、本節で提案した定性的情報に基づく順位付けには限界がある。

第一の問題は、定性物理に関する問題である。例えば、本研究では修復目標を満たす割合として成功モデルの数の割合を用いているがこれには本質的には意味がない。すなわち、ある故障モデルに対するある修復操作をシミュレート



した結果、成功する修復モデルaと失敗する修復モデルbを得たとする。この分岐には、等しい確率でどちらか一方の振舞いをするという意味は含まれておらず、情報量が少ないための解の曖昧さでしかない。このため、実際の機械は常にモデルbのように振舞うことも考えられる。同様に、副次的影響の数という項目も目安とはなるが、本質的な意味はない。この問題は、定性物理の実用化を考える場合には、重要な問題となると考えられる。

第二の問題は、定性物理世界の情報でのみ評価しているということである。この問題に対して、佐藤 [Sato90] は実際に工場において行なわれている修復作業から「修復作業の実行に要する時間」、「要する経費」、「機械に対する影響(劣化の促進等)」という項目についてデータを収集し、本節で述べた方法と合わせて総合的に修復操作を評価する方法を提案している。

#### 5. 修復操作のループ

本節で提案した手法は一度に一つの操作パラメータを操作するという原則に基づいたものであった。このため、修復実現可能性の範囲が狭いという問題点がある。例えば、図6.5に示すようなパラメータモデル上で、 $c_1$ と $c_2$ を適当に調整することにより修復に成功するという可能性を導出できない。この問題は主に第7章で述べるように修復実行能力の限界に由来する問題であり、和田 [Wada91b] は本節で示した因果マトリックスを用いる手法により複数の修復操作を同時に実行する修復計画手法を実現している。

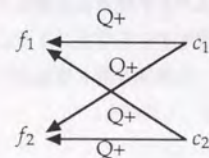


図6.5: 修復操作がループを描く例

### 6.3 機能冗長型自己修復機械の修復計画

本節では、前節で述べた制御型自己修復機械に対する修復計画手法を拡張し、機能冗長を利用した修復計画の方法について述べる。

本節の修復計画と前節の方法の違いは、修復実現の方法として、前節のアクチュエータの操作に加えて機能冗長を利用することである。第3.3.2項で述べたように、機能冗長は部品の潜在機能を利用して喪失機能を代替するために、機械の構造を変更させることにより修復を実現する。本研究の範囲では、このような機能冗長を修復時に発見的に導出し利用する方法は取らず、設計者が設計時に対象機械に付加した機能冗長系を利用して修復を行なうことにする。またこのとき、前節で述べたアクチュエータの操作による修復も同時に利用する。

#### 6.3.1 知識表現

本節の修復計画手法における知識表現も第6.2.1項で述べた知識表現方法と同じものを用いる。ただし、対象系に関する機能冗長候補の知識を記述する必要がある。機能冗長候補は以下に示す枠組で対象モデルに記述する。

##### Function Redundancy

Name:	Main Charging with TC
Target Function:	Charge on Drum
Redundant Function:	a function tree
Alternative Model:	an object model

このとき、Nameで示される機能冗長候補は、Target Functionに関して機能冗長性を発揮し、代替機能としてRedundant Function内に記述されている機能階層構造を持つ。すなわち、ある対象モデルにこの機能冗長候補を適用するとそのモデルが持っている本来の機能階層構造からTarget Function以下の部分機能階層構造とRedundant Functionを置換し、新たな機能階層構造を生成する。この時の挙動、状態レベルのモデルはAlternative Modelに記述されている部品接続グラフ、および、パラメータモデルとなる。ある対象モデルに機能冗長候補を適用した時のモデルの変化を図6.6に示す。



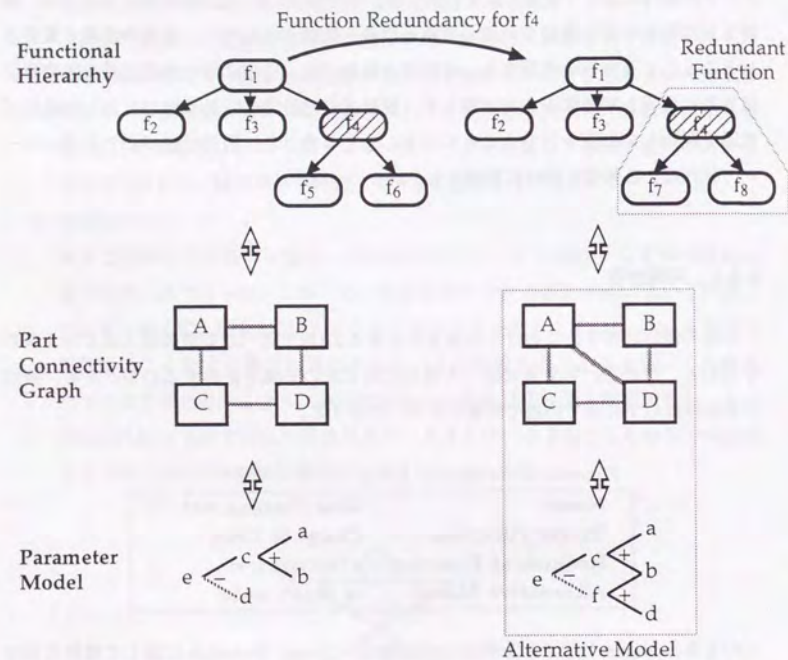


図 6.6: 機能冗長の適用によるモデルの変化

## 6.3.2 修復計画の方法

本節の修復計画手法は第 6.2.2 項で述べた方法に基づいており、機能冗長候補を利用する部分のみが異なる図 6.7 に示すアルゴリズムにより修復計画を実行する。

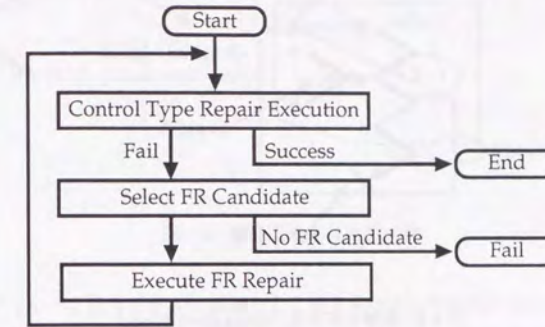


図 6.7: 機能冗長修復計画のアルゴリズム

図 6.7 において、制御型修復計画は第 6.2 節で述べた修復計画手法である。本研究の機能冗長型修復計画は制御型の修復計画に失敗した後に実行される。これは、制御型の方が修復の実現が容易であり、また、修復後の機械の動作の予想が容易であると考えられるからである。

機能冗長の選択は対象モデルに記述されている機能冗長候補から利用可能な候補を以下の二つの選択基準により導出する。

1. Target Function が修復目標機能
2. Alternative Model に故障を含まない機能冗長候補

上記 2. の基準は以下の理由による。機能冗長の適用により、挙動、状態レベルの対象モデルは Alternative Model に入れ替わる。このとき、図 6.8 に示す故障 A のように、入力された故障モデルと Alternative Model の共通部分に故障が存在した場合、対象モデルを Alternative Model に入れ換えると対象系の状態を正しく表わさなくなってしまう。そこで本研究では、Alternative Model と故障モデルの共通部分に故障が存在しない場合にその機能冗長候補を利用することにする。図 6.8 の例におい



て、故障Bに対してはこの Alternative Model を持つ機能冗長候補は利用可能である。

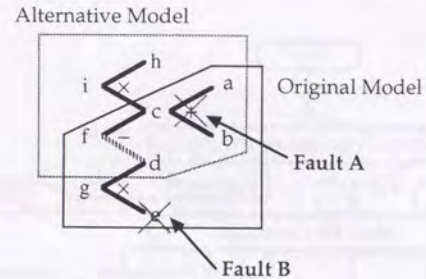


図 6.8: 故障モデルと Alternative Model

本節の修復計画手法は、以上により導出された機能冗長候補の Alternative Model のそれぞれに対し、制御型修復計画を実行する。制御型修復計画を行なった後も、修復目標機能集合  $F_m$  を全て回復できない場合には、原理的には機能冗長系を複数組み合わせ合わせて修復を実行可能であるが、ここではこの複数の機能冗長系の利用については考えないことにする。その理由は、第8章で詳しく述べるが本研究の機能冗長型自己修復機械では対象系上で複数の機能冗長を同時に発現させることができないからである。

以上のアルゴリズムにより、本節の方法で導出される修復計画は機能冗長操作を  $fr_i$  として、図 6.4 を拡張した図 6.9 のような形になる。

本手法の有効性を検証するために実験的な機能冗長型修復計画システムを構築したが、そのインプリメンテーション、および、実行例については第8章の機能冗長型自己修復機械の実行例のところで述べることにする。

### 6.3.3 考察

#### 1. 代替モデルの表現方法と機能冗長の適用範囲

本研究では、機能冗長候補の代替モデルを直接与えているため、機能冗長候補の適用範囲が故障モデルと代替モデルの共通部分に故障を持たない場合に限定

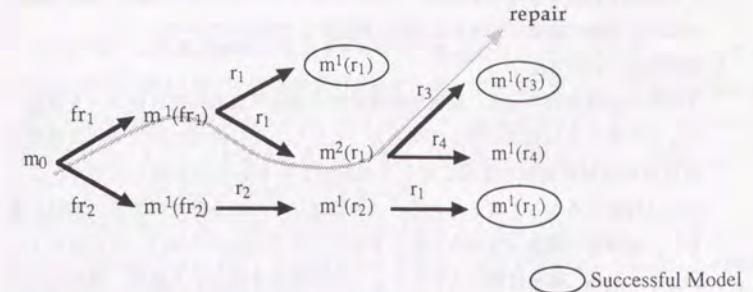


図 6.9: 機能冗長修復計画の例

されている。代替モデルの表現として正常モデルとの差異を用いるとその適用範囲は広くなると考えられるが、故障モデルにモデルの差異を適用したものが対象系の状態を正しく表現する範囲についての考察が必要になる。例えば、図 6.8 において、故障 A が発生している故障モデルに対して、差異として表現された代替モデル、すなわち、関係  $g = d \times e$  の削除と関係  $i = h \times c$  を適用しても正しい対象モデルを作成できる。この点に関しては、本研究の方法よりも適用範囲が広くなるといえる。しかし、例えば故障により  $d$  と  $e$  の間に新しい関係が生じた場合は、代替モデルの適用後、パラメータ  $e$  を残すのが正しいモデルであるのか、削除するのが正しいモデルであるのかは、故障発生過程とその故障の物理的な意味に依存し、あらかじめ決定することは難しい。このような問題に対処するためには、正常モデルと代替モデルの境界に故障による変化が生じていないことを条件とする方法が考えられる。

#### 2. 複数の機能冗長系の組合せ利用

本節で提案した方法は複数の機能冗長系を同時に組み合わせることをしないが、機能冗長系を組み合わせることで、修復可能な範囲を広くすることができると考えられる。しかし、前項の代替モデルの問題とも関連するが、本研究の代替モデルの表現では、複数の機能冗長系を組み合わせた対象モデルを生成できない。このためには、正常時との差異を表現した代替モ



デル表現を利用する必要がある。さらに、第8章で述べるが複数の機能冗長系の利用を実際の機械で実現する方法も課題として残されている。

### 3. 機能冗長の利用方法

本研究の修復計画手法は、制御型修復計画を機能冗長型修復計画よりも優先するという考え方に基いたものである。しかし、逆に適当に設計された機能冗長系はその対象範囲の故障に対して制御型よりも副次的影響を持たず、より有効に作用することも考えられる。この問題は、対象分野や対象系の設計に依存し、制御型と機能冗長型のどちらを優先するのが良いのかは一概にはいえない。そのため、修復計画システムとしては制御型を優先する場合、機能冗長型を優先する場合、および、両者を同時に評価する場合を使い分けられるようにすべきである。

## 6.4 第6章のまとめ

本章では、自己修復機械を実現するために、修復計画を自動化する方法について述べた。まず、第6.1節で、修復計画の自動化に際しても第5章で述べた故障診断の自動化手法と同様の問題があるため、第5章で述べた定性物理を用いたモデルベースト推論を利用することが有効であること、また、修復特有の問題として「機能トレードオフ」を考えなければならないことを述べた。

第6.2節においては、制御型修復操作を対象とする修復計画手法について述べた。また、第6.3節においては、機能冗長系を利用した修復を実現するために、この制御型修復計画手法を拡張した機能冗長型修復計画の方法について述べた。

これらの修復計画手法に基づき、修復を実行する方法は第7章、第8章でそれぞれ述べる。また、これら修復計画手法の有効性の検証についても第7章、第8章における実行例とその考察において行なうことにする。



## 第 7 章

# 制御型自己修復機械設計方法論



本章では、制御型自己修復機械の実現方法と実験について述べる。まず、制御型自己修復機械の対象機械部の設計方法を述べる。次に、実際に制御型自己修復機械の実験機を構築し、実験を行なったので、この実験機の製作、および、実験結果について考察する。この実験の結果、制御型自己修復機械は故障の状況に応じて柔軟に修復方法を定めることができ、有効であることが明らかになった。

## 7.1 制御型自己修復機械の設計方法

### 7.1.1 制御型自己修復機械設計の要求項目

第3章で述べたように、本研究の自己修復機械は、本来発揮する機能の他に、状態監視機能、故障判定機能、故障診断機能、修復計画機能、および、修復実行機能を発現する機械であると定義できる。特に、制御型自己修復機械は制御型の修復実行機能を実現する自己修復機械である。ここで、故障判定機能、故障診断機能、修復計画機能は、第4章～第6章で述べた方法により故障診断/修復計画システムを計算機上に構築し、機械部と接続することで実現することができる。本節では、残りの機械部の設計方法について考察する。このとき、機械部において本来の機能を発現させるための設計は、一般の機械の設計と同等であり、その設計方法論は他の研究に従い、本研究では議論しない。ここでは、自己修復機械の機械部を特徴付ける状態監視機能、修復実行機能を機械部で実現するための設計方法を議論することにする。

本研究が対象とする一個のメカトロニクス機械システムにおいて、一般に、状態監視機能はセンサにより実現され、また、制御型修復実行機能は、その定義からアクチュエータの制御により実現されると考えられる。このため、状態監視機能、修復実行機能を機械部で実現するための設計問題は、ここでは、以下の三点を満たす機械の構造、および、センサ、アクチュエータの配置問題と捉えることができる。

1. 機能のトレードオフを実現し、故障時においても要求機能発現が可能な機械構造
2. 機械の状態を適切に把握できるセンサの配置
3. 広い故障の範囲に対して、柔軟に修復を実現できるアクチュエータの配置

以下、本節では、上記の機械の構造、および、センサ、アクチュエータの配置問題に関して設計者を支援する方法について考察することにする。

### 7.1.2 機械部の設計方法

前項の要求を満たす機械部を設計する方法として、ここでは、図7.1に示すような対象機械の保全用のFBSダイアグラム表現をもとにして制御型自己修復性を発揮し



やすいように、対象機械の構造<sup>1</sup>の評価やセンサ、アクチュエータの配置を決める方法を考える。このとき、設計者からみた対象機械の必要機能は既に表示されているものとし、以下、図7.1の例を用いて改良設計の方法を議論することにする。

図7.1は、機能階層構造とパラメータモデルから成る簡単化された保全用FBSモデルを表している。機能階層構造において、 $f_1 \sim f_6$ は機能ノードを表しており、 $f_3 \sim f_6$ には condition タグに発現条件が記述されている。例えば、 $f_3$ の発現条件はパラメータ  $a$  の値がランドマーク  $LowA$  より大きく、ランドマーク  $HighA$  よりも小さいこと ( $<(LowA, a) \ \& \ <(a, HighA)$ ) である。この発現条件により、各機能ノードはパラメータモデル上のパラメータと図中の破線により関係付けられている。本例では、必要機能を  $f_1$ 、 $f_2$ 、 $f_3$ 、 $f_5$ 、 $f_6$  とし、 $f_4$  は必ずしも必要のない機能とする。

#### (1) アクチュエータの配置

自己修復機械におけるアクチュエータは、最小のアクチュエータで機能に対する修復をより柔軟に行なえるものでなくてはならない。これを実現するために、アクチュエータの配置に関して幾つかのヒューリスティックスを導出できる。

##### 1. 技術的、物理的要因からの限定。

技術的、コスト的な要因からアクチュエータの配置が制限される。さらに、パラメータには物理的に直接操作可能なものと不可能なものがある。

##### 2. 全ての必要機能を操作可能にしなければならない。

図7.1の例でいえば、機能に関連するパラメータ  $a$ 、 $b$ 、 $c$ 、 $i$  を操作可能でなければならないが、機能  $f_4$  は必ずしも発現しなくても良い場合は、 $i$  は操作可能でなくても良い。この条件は、パラメータモデル上の関係で自動的に判定することが可能である。例えば、上記の条件を満たす最小のアクチュエータの組として  $j$  と  $l$  を得ることができる。またこのとき、故障によりパラメータモデルが変化する可能性があることを考えれば、1つの機能を操作可能なアクチュエータは複数であることが望まれる。これを「アクチュエータに関する冗長性」と呼ぶことにする。

<sup>1</sup>ここでの「構造」は、設計終了時に決まる対象機械の部品やその関係、および、それによって決まる対象機械の動作特性のことである。

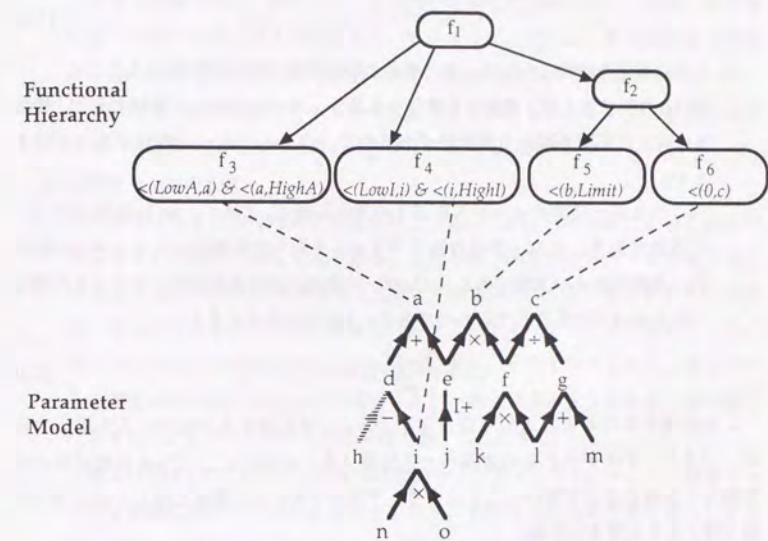


図7.1: 制御型自己修復機械に対するFBSダイアグラムの例



そこで、このヒューリスティクスを評価する関数  $C_n$  を以下のように定義する。ここで、 $n$  は必要機能の数、 $c_i$  は  $i$  番目の必要機能に対するアクチュエータ冗長性、すなわち、ある機能からアクチュエータへの因果的に意味のあるパス数、であり、 $C_n$  は、対象系における各必要機能の平均操作パラメータ数を表わす。 $C_n$  が大きい方が望ましく、 $c_i$  が1以下の機能は故障時に制御できず危険であると表現できる。

$$C_n = \frac{\sum_i c_i}{n} \quad (i = 1, 2, \dots, n) \quad (7.1)$$

3. 1つの機能を操作した時に、他の機能に影響が無い箇所が望ましい。

図7.1の例で言えば、機能  $c$  を操作する場合、パラメータ  $k$  を操作すると、機能  $b$  に副次的影響が現れる可能性があるので、パラメータ  $m$  を操作の方が好ましい。

このヒューリスティクスを以下の評価関数  $C_f$  で表す。 $m$  は操作パラメータの数であり、 $f_i$  は  $i$  番目の操作パラメータから必要機能パラメータへの因果的に意味のあるパス数である。 $C_f$  は、対象系における各操作パラメータの機能に対する独立性を評価するものであり、1に近いほど望ましい。

$$C_f = \frac{\sum_i f_i^2}{m} \quad (i = 1, 2, \dots, m) \quad (7.2)$$

1. を表現するために、適用可能なアクチュエータに関する知識ベースを構築すれば、パラメータモデルとこの知識ベースに基づき、上記の  $C_n$ 、 $C_f$  を目的に合わせ、適当に重み付けして用いることにより、アクチュエータの配置に関して設計者を支援可能であると考えられる。

図7.1の例でいえば、2～4個のアクチュエータを付加する場合、 $C_n$  が大きい ( $h j k l$ ) や、 $C_n$  が大きく  $C_f$  が小さい ( $h j l m$ ) が適当な候補として導出される。

## (2) センサの配置

自己修復機械におけるセンサは、最小のセンサ数で常に機械の状態、特に、機能の発現状態を適切に把握可能であることが必要である。このセンサの配置問題に対し、幾つかのヒューリスティクスを導くことができる。

1. 技術的、物理的要因からの限定。

例えば、複写機のドラム上の電位を正確に測定するよりも、入射光量やチャー

ジャーの電圧は容易かつ安価に測定することができる。また、ベアリングの摩耗量は直接、動的に測定することはできないが、アコースティック・エミッション [Akizuki91] により間接的に測定可能である。

2. 必要機能の発現の有無を直接観測可能な箇所 (図7.1では、 $a$ 、 $b$ 、 $c$ ) には、センサが必要である。

これは、知的判断部の故障判定機能を実現するための要請である。このとき、「正常な画像を得る」機能等センシングの自動化が困難であり、かつ、使用者による判定が容易である機能もある。このように、センシングの自動化と使用者による判定の割り振りを設計時において適当に定めなければならない。

すなわち、必要機能パラメータの数を  $n$ 、そのうちセンシングされているパラメータ数を  $m$  とした時、 $S_f = \frac{m}{n}$  が1に近いほど望ましい。

3. 均等なセンサの配置

センサは対象系全体に均等に配置されていることが、故障診断における故障の同定をより容易にすると考えられる。このとき、あるパラメータを測定するということはそのパラメータより「上流」の対象系<sup>2</sup>の状態を測定するものである。そこで、あるセンサの「測定部分」を「パラメータモデルにおいて、そのセンサよりも上流の部分ネットワークで、かつ、そのセンサよりも上流のセンサの測定部分に含まれていない部分」とし、測定部分の大きさをその測定部分に含まれているパラメータと関係の数で表すことにする。

例えば図7.1で、 $i$  にセンサをつけた場合、パラメータ  $i$ 、 $n$ 、 $o$ 、関係  $i = n \times o$  が測定部分でその大きさは4であり、さらに  $a$  にセンサを付加した場合は、パラメータ  $a$ 、 $d$ 、 $e$ 、 $h$ 、 $j$ 、関係  $a = d + e$ 、 $d = i - h$ 、 $I + (e, j)$  が含まれ、 $a$  の大きさは8である。そこで、評価関数  $S_d$  を以下のように表現し、このヒューリスティクスを  $S_d$  が小さくなるようすることとする。ここで、 $m$  はセンサの個数、 $g_i$  は  $i$  番目のセンサの測定部分の大きさである。

$$S_d = \frac{\sum_i g_i^2}{m} \quad (i = 1, 2, \dots, m) \quad (7.3)$$

上記1. を表現するセンサに関する知識ベースを構築すれば、パラメータモデルとこの知識ベースに基づき、かつ、評価関数  $S_f$ 、 $S_d$  を目的に応じて適当に重み付けし

<sup>2</sup>そのパラメータに因果的に影響を及ぼし得るパラメータモデルの部分ネットワーク



て利用することにより、センサ配置に関して設計者を有効に支援することが可能になる。

例えば、必要機能が  $f_1$ 、 $f_3$ 、 $f_5$ 、 $f_6$  である図 7.1 に 5 個のセンサを配置する場合、 $(a b c d f)$  が  $S_f = 1$ 、 $S_d = 28$  となり、適当であることがわかる。

### (3) 自己修復に適した機械構造

(1) で述べた、冗長性と操作独立性を考慮したアクチュエータの配置により、故障時にも必要機能を回復可能な制御型自己修復機械を構築可能である。このとき、これら二つの要求を実現容易な機械の構造はパラメータモデル上で考えると、機能に対する因果関係が直列的に続くよりも、並列的に広がる形態をしていることが必要である。例えば、図 7.2 において、(a) の構造を持つよりも (b) の構造を持つ必要がある。

また、機能に影響を与えている各因果的パスが長いほど故障の影響を受ける可能性が高いと考えられる。これは、いわゆる信頼性工学 [Shiomi68] において要素の直列連結の信頼性が各要素の信頼性の積として与えられるという事実に基づいている。この項目を評価するために以下の  $C_d$  を導入する。 $d_{ij}$  は、 $i$  番目の機能と  $j$  番目の入力パラメータ (定数パラメータ、可操作パラメータ) が因果的に関係を持つ場合にのみ定義され、パラメータモデル上での距離を両者間の関係の個数として表し、 $n_d$  は  $d_{ij}$  の数を表す。このとき、 $C_d$  は入力パラメータと機能の対象系における平均距離を表し、これが小さいほど望ましい。

$$C_d = \frac{\sum_{i,j} d_{ij}}{n_d} \quad (7.4)$$

以上、冗長性、操作独立性のあるアクチュエータの配置が容易で、かつ、故障の影響を受けにくい機械の構造は、パラメータモデル上の全入力パラメータに対して (1) で述べた評価関数  $C_n$ 、 $C_f$ 、および、上記の  $C_d$  を利用して評価できる。

以上、アクチュエータの配置方法、センサの配置方法、および、機械の構造の評価方法について述べた。実際に機械部を設計する場合には、これらの三視点からの評価を交互に行ないながら、機械の構造とアクチュエータ、センサの配置を決定することが必要である。

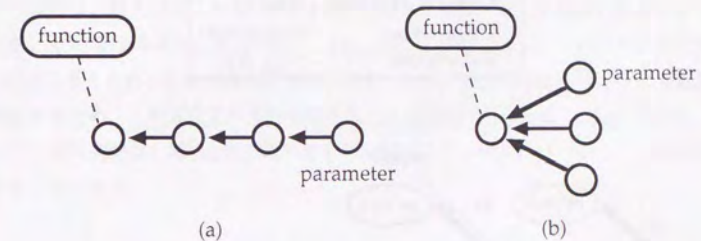


図 7.2: パラメータモデルの構造



## 7.2 制御型自己修復機械における修復の実現

制御型自己修復機械において修復実行機能を実現するためには、第6.2節で述べた手法により導出された修復計画を第7.1節で述べた機械部に対して実行しなければならない。この部分は知的判断部と機械部を接続する部分に相当し、制御型自己修復機械の特性に大きく影響を与える部分である。本節ではこの修復実行の方法を述べる。

第6.2節で述べたように修復計画は一般に図7.3に示す形をしており、各修復操作 $r$ は以下のような形で与えられる。このとき修復実行部は、Control Method に記述されたアクチュエータの自動制御により Objectives の状態を実現しなければならず、また、Subeffects の発現を監視しなければならない。

Repair Method	
Control Method:	lamp up
Objectives:	$< (x, limit)$
Subeffects:	$> (y, N)$

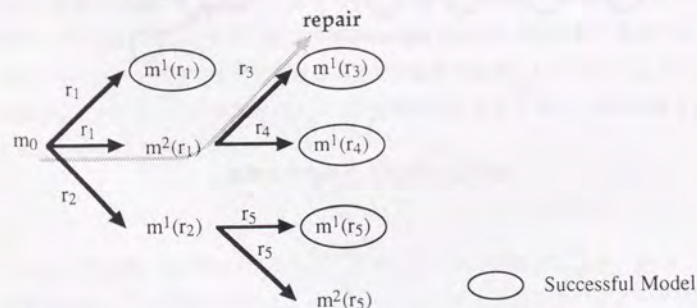


図 7.3: 修復計画の例 (再掲)

ここで問題となるのは、修復計画により導出される操作は定性的、つまり、操作の方向のみを示しており、かつ、故障により対象系の特性は変化している可能性があるため、操作パラメータに対する操作目的パラメータの感度は一般に不明であることである。このため、操作量を決定できない。そこで本研究では以下のようにこの修復操作命令を実行する。

## 7.2. 制御型自己修復機械における修復の実現

1. 操作パラメータを  $a$ 、現在の定量値を  $v_a$  としたとき、各操作パラメータに最小操作量  $\delta v_a$  を定義し、操作命令が増加であれば、操作量を  $+\delta v_a$ 、減少であれば  $-\delta v_a$  とする。このとき、操作パラメータには可操作範囲をあらかじめ与えることにする。
2. 1. で決められた操作を対象系に対して実行し、状態を測定する。測定値と図 7.3 の修復モデルを比較することにより、現在の状態を同定する。
3. 対象系に副次的影響が発生したら、修復操作を中止する。
4. 対象系で修復目標を達成していた場合は、修復操作を成功、終了する。
5. 操作量の限界に到達したら、修復操作を停止させる。

以上の修復実行方法では、複数のアクチュエータの同時操作は実現が困難である。その理由は、例えば図 7.4 に示すようにパラメータ  $a$  および  $b$  を同時に増加させたい場合、その操作の組として  $(\delta v_a, 0)$ 、 $(\delta v_a, \delta v_b)$ 、 $(\delta v_a, 2\delta v_b)$ 、... というように合計 120 通り考えられ、これは操作パラメータ数、操作可能量が増えるにつれて指数的に増加するため、これら全ての実行を試みることは困難であると考えられるからである。そこで、本研究では、第6.2節で述べたように同時に一つのアクチュエータのみを操作することにする。

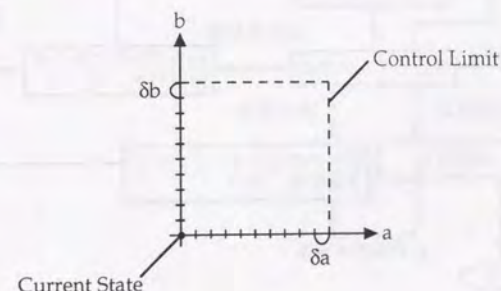


図 7.4: 二つのアクチュエータを操作する場合の操作空間

さらに、図 7.3 のような修復計画を修復計画推論時の全て導出することは、その大部分を使用しないことを考えると無駄である。そこで本研究では、図 7.5 に示すよう



に、修復計画と修復実行をインタラクティブに行なうことにする。図7.5において、「修復計画生成」は、図7.3における一操作分の修復計画を行なうものである。また、「副次的影響に対する修復計画、実行」や「現在の修復モデルに対する修復計画、実行」では、本アルゴリズムが再帰的に実行される。ただしこのとき、対象系に対して既に適用されている修復操作やその逆の修復操作は選択しない。また、結果として可能な修復操作が無くなり推論が停止した場合には、状態を修復操作実行前の状態に戻し、別の修復計画を実行することにする<sup>3</sup>。「修復モデルの選択」は、測定値から現在の状態を表す修復モデルを修復シミュレーションの結果与えられる状態遷移列から選択するものである。

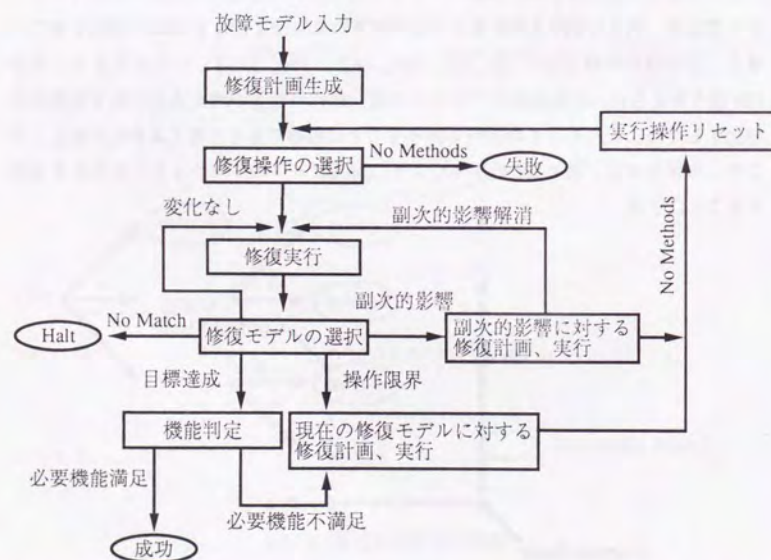


図7.5: 制御型修復計画、実行のアルゴリズム

<sup>3</sup>ここでは、修復操作の可逆性を仮定している。逆に、可逆でない操作はシミュレーションの結果から判定することが可能であると考えられる。

以上のアルゴリズムによる典型的な修復実行のパターンを図7.6に示す。(a)は単純に修復に成功する例であり、(b)はパラメータbにより副次的影響を抑えつつパラメータaにより修復に成功する例である。(c)はパラメータaによる操作が限界に達した後、パラメータbにより修復が成功する例である。また、(d)はパラメータaが限界に達した後、パラメータbにより修復を実行したが、修復に失敗したので、操作を一旦リセットし、パラメータbの操作後にパラメータaの操作を実行し、成功した例である。しかし、本節で提案した方法では(e)に示すような場合には修復に失敗する。この問題に対処し、複数のアクチュエータを利用した修復を実現することは今後の課題であり、第7.5節で考察することにする。



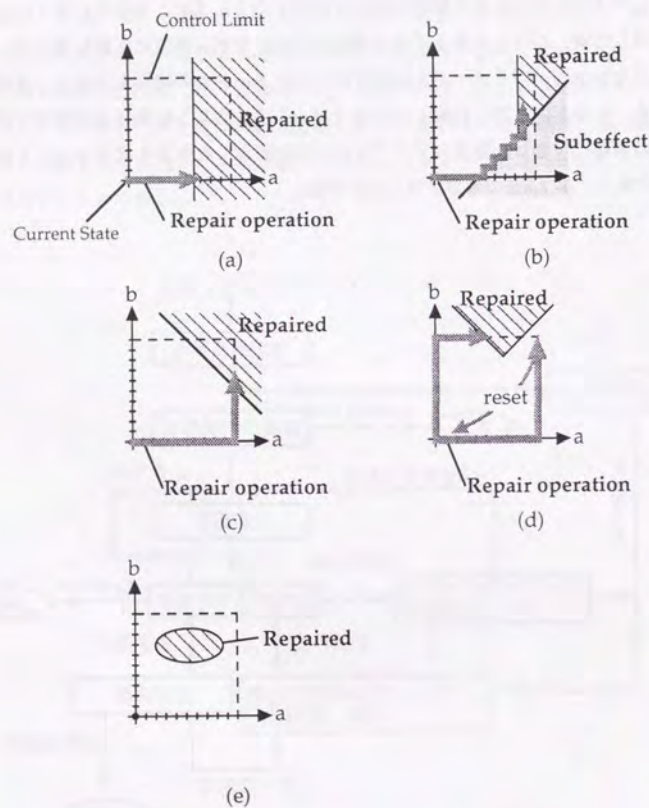


図 7.6: 修復実行の主なパターン

## 7.3 実験機の設計

本研究では、今まで述べてきた手法の有効性の検証と問題点を明らかにするために、制御型自己修復機械の実験機を構築した。本節ではこの実験機の構成、知識の記述、および、推論部と機械部の接続に関して述べる。

## 7.3.1 実験機の構成

本研究では、既存の複写機を改造して機械部とすることにより制御型自己修復実験機を構築することにした。複写機を対象とした理由は、典型的なメカトロニクス機械のため、計算機との接続が容易であり、かつ、計算機から柔軟に制御を行なうことが可能だと考えたからである。また、自己修復型複写機を新規に設計することは様々な理由から困難であるので、既存の機械にセンサ類、アクチュエータ類、および、計算機とのインタフェースを付加することにより、機械部を構築することにした。

表 7.1 に実験機の諸元を、図 7.7 に実験機の内部構造を示す。図 7.7 内の Densitometer は出力紙上の濃度を測定する器具であり、ここでは出力画像濃度を人手により測定することにする。本研究では実験対象を実験機のドラム周辺部に限定した。すなわち、紙送り系の故障や定着系の故障は扱わない。また、例えば、画像が歪む等の画像の詳細な情報を必要とする故障は画像処理技術が大きな要因となるので扱わないことにした。しかし、これらの限定は基本的にセンサ、アクチュエータ、および、知識の量の問題であり本実験システムを拡張することにより、容易に取り扱い可能である。

図 7.8 に一枚のコピーを取るための複写機の基本的な動作を示す。すなわち、まず、ドラムをメインチャージャにより正に帯電させ (1)、ハロゲンランプから発し原稿の白色部分で反射した光が帯電したドラムに入射する。ドラムは光半導体であるため光が当たるとその部分のみがアースされ電位が 0 となる (2)。次に、現像ローラで負に帯電させられたトナーがドラムに静電気力により吸着する (3)。さらにドラムが回転し転写部に入ると、転写チャージャはドラムの電位より強い正の電位を持つため、トナーは静電気力によりドラム上から出力紙に引き付けられる (4)。さらに、分離チャージャがトナーの乗った出力紙に交流放電を行ない、ドラムと出力紙を分離する (5)。本実験機の対象はここまでであるが、トナーの乗った出力紙はその後定着部で熱ローラにより加熱加圧されトナーが溶着し、出力される (6)。また、ドラムはクリーニングブレードにより余ったトナーを除去され (7)、除電ランプにより電位を



## ・基本仕様

形式	卓上型
複写方式	乾式間接式静電複写方式
複写サイズ	最大 A4 最小 A6
複写速度	12 枚 / 分 (A4)
複写倍率	等倍
給紙方式	シングルカセット自動給紙、および、手動給紙
使用ペーパー	普通紙、特殊紙
連続複写	1 ~ 99 枚
主帯電方式	コロナ放電
露光方式	原稿台固定によるスリット露光方式
光源	ハロゲンランプ
現像方式	乾式 (磁気ブラシ現像方式)
トナー補給	トナー自動補給機構付き
転写方式	コロナ放電
分離方式	AC コロナ放電
定着方式	熱ローラ方式 熱源: ハロゲンヒーター
クリーニング方式	ブレードクリーニング方式・除電ランプによる除電
感光体	セレン感光体
大きさ	幅 535mm 奥行 505mm 高さ 279mm

## ・追加改造仕様

状態監視機能	表面電位センサによる感光体表面電位 ( $V_s$ ) の測定 反射型フォトセンサによる感光体画像濃度 ( $D_s$ ) の測定 フォトセンサによるランプ光量 ( $X$ ) の測定
修復実行機能	メインチャージャの出力電圧 ( $V_nC$ ) 制御 転写チャージャの出力電圧 ( $V_tC$ ) 制御 ハロゲンランプの出力光量 ( $HIC$ ) 制御

表 7.1: 制御型自己修復実験機の諸元

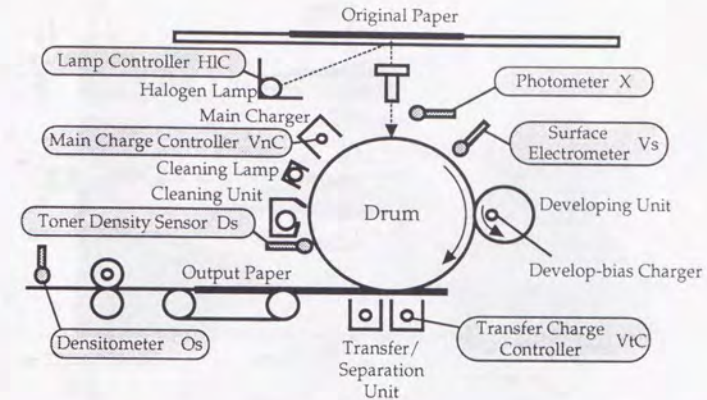


図 7.7: 制御型自己修復実験機の内部構造

アースされ (8)、一回転のプロセスを終了する。

以上の複写機の基本的動作、特に設計者が持つ複写機の動作特性に関する知識に基づいて作成した実験機の対象モデルが第 7.3.2 項に示す図 7.12、図 7.13 である。本実験機は図 7.12 のパラメータモデルに基づいて以下のように設計された。まず、対象機械の構造は本実験機が市販機の改造ということから変更することはできず、あらかじめ与えられた。センサ、アクチュエータの配置に関しては物理的、技術的制約条件が最も大きい要因となった。また、実験機の必要機能に関連するパラメータとして出力紙の濃度  $O_s$  と出力紙の分離特性  $Sp^4$  を指定した。

以上の機械部を用いて図 7.9 に示すような制御型自己修復機械システムを構築した。図 7.10 に本システムのアルゴリズムを示す。図 7.9 中で、PC9801 は接続ボードを介して対象機械部と接続しており、機械部からのセンサ値を受け取り定性化し、マッキントッシュに送り、また、マッキントッシュから送られる修復操作命令に従い定量的なアクチュエータの制御値を機械部に送る働きをする。対象機械部である複写

<sup>4</sup>このパラメータは、紙が実験機から出力されれば正常、そうでなければ異常と目視により測定、入力することにした。



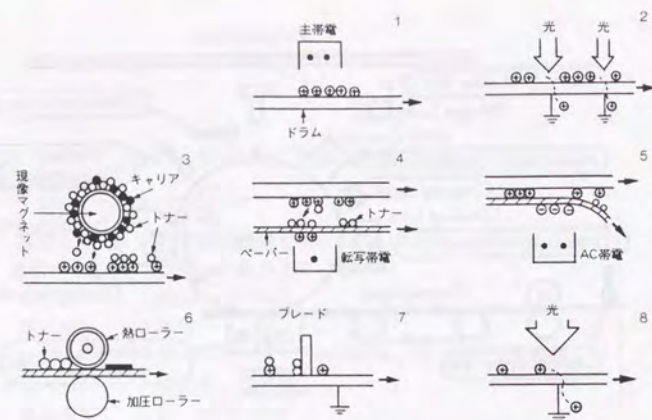


図 7.8: 複写機の動作

機は受け取った制御値を用いて複写動作を行なう。マッキントッシュ上に第4.6節、第5.2節、第6.2節で述べた対象のモデリング手法、推論手法に基づく故障診断 / 修復計画システムを Smalltalk-80 を用いてインプリメントしており、PC9801 からの定性的測定値に基づき、故障判定、故障診断、および、修復計画を行ない、修復操作を PC9801 に出力する。

自己修復の一サイクルが終了した場合、一般的には機械部の状態は劣化の影響等により設計時の状態と異なっているが、正常であるという状態になる。すなわち、現在の状態が新しい正常状態になると考えられる。そこで本研究では、図 7.10 内の "Make New Object Model" において、現在の状態と最も一致している対象モデルを選択し、それを今後の正常モデルとする。また、このモデル内では各パラメータ値は必ずしも正常値を表すランドマーク N を指してはいないので、図 7.11 に示すように、正常モデル内で各パラメータが指している値の場所に現在の正常値を意味する CN (Current Normal) というランドマークを生成することにする。この CN を基準として今後の故障診断 / 修復計画が実行される。

### 7.3.2 知識の記述

次に本実験システムにおいて実験機を対象として記述した知識について述べる。

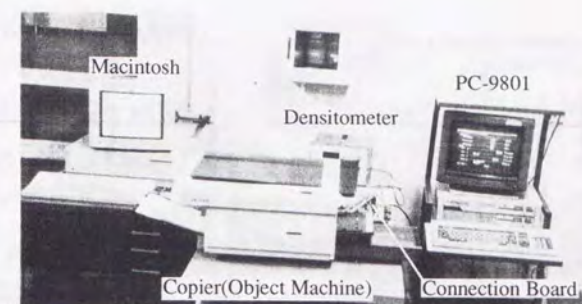


図 7.9: 制御型自己修復機械システムの構成

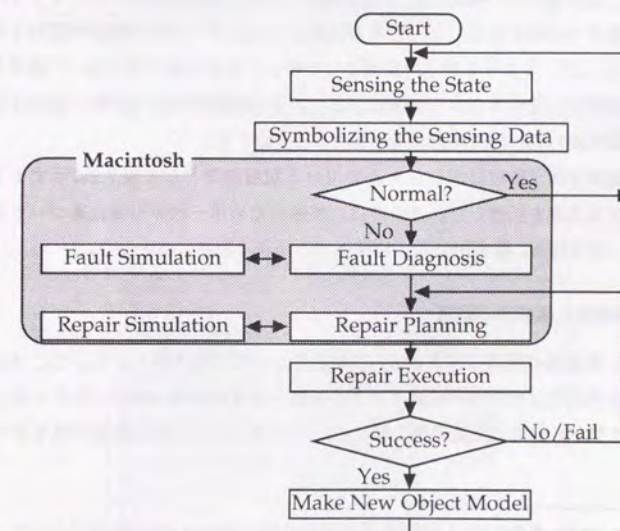


図 7.10: 制御型自己修復機械システムのアルゴリズム



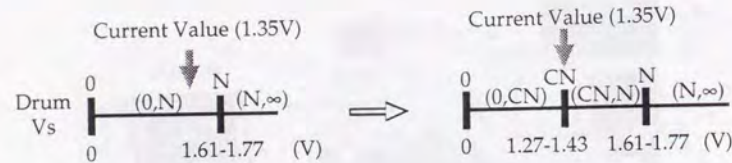


図 7.11: 現在の正常状態を表すランドマークの生成

図 7.12、図 7.13 に、本実験機を表現する保全用 FBS モデルを示す。図 7.12 が挙動・状態レベルの表現であり、図 7.13 が機能レベルの表現である。本実験では、必要機能として、図 7.13 内で灰色のノードで示される「複写を取る」機能、および、この機能の部分機能である「出力画像がカブラない<sup>5)</sup>」機能、「出力用紙が分離される」機能、「出力画像を得る」機能とその上位機能の合計六機能を指定した。カブリの判定は、基準濃度の用紙を複写した出力紙の濃度が一定以下であれば機能が満たされると判定される<sup>6)</sup>。カブリを特に対象例としたのは、実際の複写機において画像不良の中で比較的頻繁に発生するからである。また、出力画像の判定も同様に基準原稿を用い、一定濃度以上であれば機能が満たされると判定する。

また、故障診断 / 修復計画システムで用いる現象知識として表 7.2 に示すような故障に関連する現象を記述した。ここでは、簡単のため単一故障仮説に基づいて知識を記述した。すなわち、因果的に互いに関連する現象は記述しなかった。

### 7.3.3 推論部と機械部の接続

推論部と機械部の接続は図 7.9 の PC9801 において実現されている。ここには、センサデータの定性化のための知識とアクチュエータ操作知識が記述されている。典型的なセンサデータ定性化知識の例を図 7.14、アクチュエータ操作知識の例を図 7.15 に示す。

<sup>5)</sup> 「カブリ」とは、新聞のような原稿のコピーを取った場合に出力画像の地肌が黒くなってしまう状態を表す用語である。

<sup>6)</sup> 基準原稿や基準濃度は実際に複写機メーカーで使用されているものを用いた。

<sup>7)</sup> Drum Vs の正常値はこの場合  $1.68 \pm 0.09$  (V) となる。

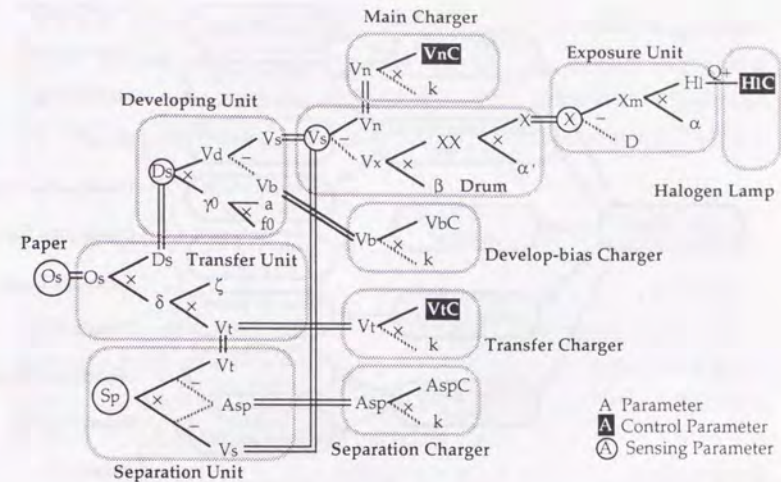


図 7.12: 実験機の対象モデル (挙動・状態レベル)

表 7.2: 記述した現象知識

名前	内容
BadOpticalPath	光路障害
ChargerCut	チャージャ線断線
ChargerDetHigh	チャージャ劣化
ChargerDetLow	チャージャ劣化
DrumDeterio	ドラム劣化
DrumEarth	ドラムがアースされる
HalogenCut	ハロゲンランプ断線
HalogenDeterio	ハロゲンランプ劣化
HalogenDeterio2	ハロゲンランプ劣化
PaperDeterio	出力紙劣化
PoorToner	トナー不足
TonerDeterio	トナー劣化
TonerStickingOnMc	チャージャへのトナー付着



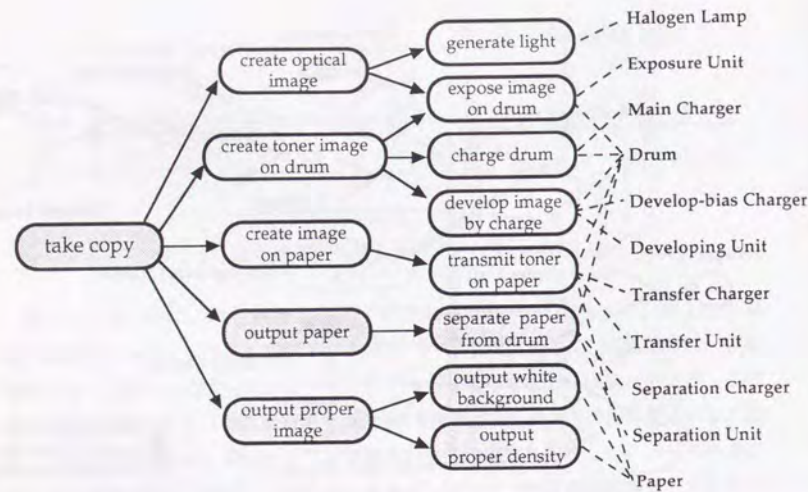


図 7.13: 実験機の対象モデル (機能レベル)

Parameter:	<i>Drum</i> <i>Vs</i>
Map:	$0 \leftrightarrow 0$ (V)
	$N \leftrightarrow 1.68$ (V)
Error:	$0.09$ (V) <sup>7</sup>

図 7.14: センサデータ定性化知識の例

Parameter:	<i>MainCharger</i> <i>VnC'</i>
Range:	$0 - 5$ (V)
Unit Value:	$0.1$ (V)

図 7.15: アクチュエータ操作知識の例

#### 7.4 制御型自己修復実験

前節で述べた自己修復実験機を実際に作動させて実験をいくつか行なった。実験の目的は、本実験機が実際に自己修復性を発揮可能であることを検証すること、および、問題点を洗い出すことである。実験の方法としては、実験機に対して意図的に故障を与えてやり実験機を動作させた。このとき、基準原稿に対する複写動作を行なった。

##### 7.4.1 実行例 1

ハロゲンランプの出力を低く設定して実験機を動作させたところ、カブリが発生した。この場合、故障診断部で「ハロゲンランプの劣化」現象が発生していると正しく故障原因を同定することができ、ハロゲンランプの出力を上昇させるという修復操作により、正常に機能を発揮する状態に回復することができた。この例は、一般の複写機で行なっている画像濃度の自動調整とほぼ同様である。そこでさらに、ハロゲンランプの制御範囲を小さくした。この場合は、ハロゲンランプの出力上昇のみでは修復することはできなかったが、さらにメインチャージャの出力電圧を低下させることにより図 7.6 の (c) のパターンにより修復に成功した。図 7.16 に修復実行中の故障診断 / 修復計画システムの画面の例を示す。図中の FunctionModeler ウィンドウは現在の機能の発現状態を示しており、黒塗のノードは発現していないことを、白塗のノードは発現していることを、斜線のノードは発現状態が不明なノードをそれぞれ示している。Fault Candidates ウィンドウは故障原因が「ハロゲンランプの劣化」現象に同定されたことを示しており、Repair Plan ウィンドウには実行可能な修復操作候補が示されている。この状態では、ハロゲンランプの出力増加は既に限界に達しているので候補から除外されている。Repair History ウィンドウには修復の実行履歴が示されており、ハロゲンランプの出力上昇が限界に達した後、メインチャージャの出力低下操作が行なわれていることを示している。Parameters ウィンドウには PC9801 から得られたセンサ、アクチュエータの定性値が示されている。また、図 7.17 に修復成功時の PC9801 の実行画面を示す。DN と示されているのが設計時における正常値であり、この状態では各センサ、アクチュエータ値がそれぞれ正常値とずれているが、機能を表すパラメータ *Os* の値は限界値 *LIMIT* 以下になっているので、機能を発現している状態にあることがわかる。また、本例における修復前の出力画像と修復後



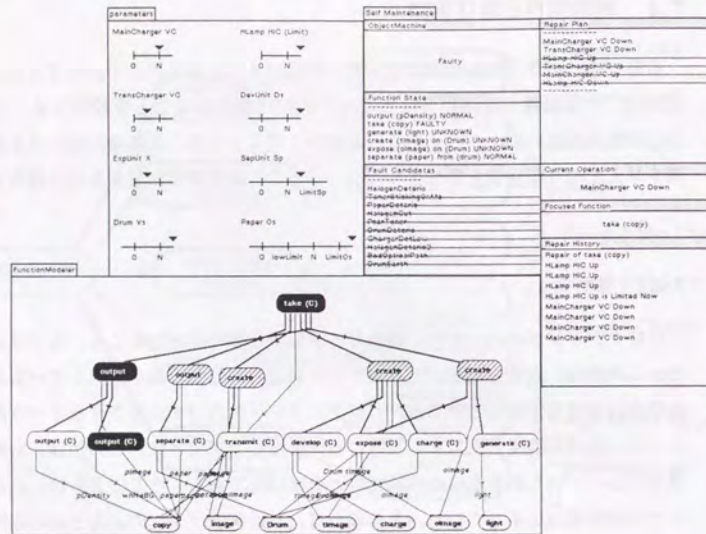


図 7.16: 故障診断 / 修復計画システムの実行画面 (実行例 1)

の出力画像を図 7.18 に示す。実際に Macintosh IIx、EPSON PC-286 を用いた場合に、実行例 1 を行なうのに要した時間は合計約 8 分である。一複写動作に約 10 秒を要し、ここでは、ハロゲンランプの上昇 3 回、メインチャージャ下降 6 回実行した。推論に要した時間は合計約 3 分である<sup>8</sup>。

この実行例 1 により、本実験システムにおいて本研究が目的とした故障診断、修復計画、修復実行を行なうことができ、また、これにより自己修復性を発揮することができたと結論付けることができる。さらに本例より、本実験システムは対象機械の状況に応じて柔軟に修復を計画、実行することができるという特徴を持つことが明らかになった。すなわち、本システムは故障パターンとその修復方法という形の知識を利用せずに、対象の因果的な関係と操作知識から適切な修復操作を導出することができる。

<sup>8</sup> この他、複写動作の起動や濃度測定に人手による操作を要す。

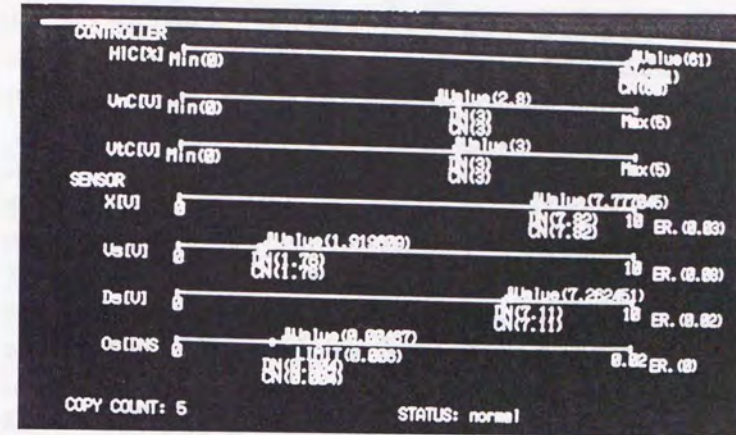
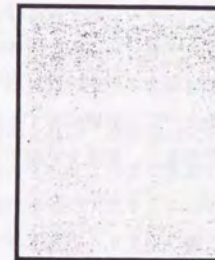
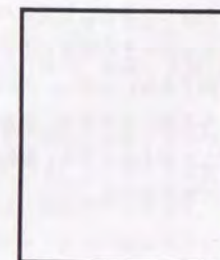


図 7.17: PC9801 の実行画面 (実行例 1)



修復前の出力画像



修復後の出力画像

図 7.18: 修復前後の出力画像



## 7.4.2 実行例2

実行例2として、現像バイアス・チャージャを取り除いて実験機を動作させ、カブリを引き起こした。現像バイアス・チャージャは図7.7に示すように画像のコントラストを与えるために、現像部において正の電位を与え、トナーのドラムへの吸着に対して閾値を与えるものである。この状態に対する修復実行中の画面を図7.19に示す。図に示すように、故障原因は「チャージャ(この場合は現像バイアス・チャージャに対応)の断線」現象か「チャージャの劣化」現象に原因を絞り込んでいる。これらが引き起こす症状の違いはチャージャの出力が0か(0,N)ということであり、ここでのセンサ、および、モデル情報ではこれ以上絞り込むことはできない。これを区別するためには目視による点検や定量情報の利用が必要になる。しかし、結果としてこの程度まで故障を絞り込むことができれば、修復に成功するので、大きな問題とはならない。前者のチャージャ断線現象は図7.12に示す対象モデル内で、部品 Develop-bias Chargerの属性値とパラメータ関係を変化させ、結果として出力電圧Vbが0となる。このような故障状態に対してシステムはハロゲンランプの上昇とメインチャージャの低下により修復を実現し、必要機能を回復した。

本修復例は、構造欠損を伴う故障に対して本システムが有効に修復を実現することができることを示している。このように、故障の種類によっては構造変更を引き起こすようなものに対しても制御型の修復操作により機能を回復することができる。一般の制御技術を利用した場合にも現像バイアス・チャージャの切断を設計時に予測し、切断時のアクチュエータの制御方法をあらかじめ与えておけばこれと同等の動作を行なうことは可能である。しかし、このような方法では保全の不定性に対応することが困難である。本実験システムは、現像バイアス・チャージャ断線に対する対処方法を明示的に記述しておかなくても、動的に修復方法を発見し、修復を実現することができるという点で優れている。

## 7.4.3 実行例3

実行例3として、メインチャージャを断線させて複写動作を行なった。この場合はカブリは生じなかったが、出力画像は真っ白になり正常出力を得られなかった。故障診断部は「チャージャ(この場合はメインチャージャ)の断線」現象、または、「チャージャの劣化」現象が発生していることを診断できた。さらに、それに対する

## 7.4. 制御型自己修復実験

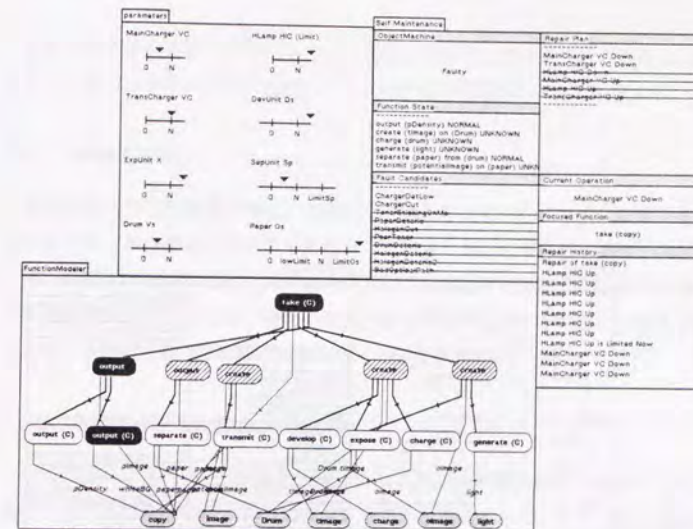


図7.19: 故障診断 / 修復計画システムの実行画面 (実行例2)

修復計画としてメインチャージャの出力上昇、ハロゲンランプの出力低下、および、転写チャージャの出力上昇が得られたが、実行の結果、修復に失敗した(図7.20参照)。

この修復の失敗は、制御型の本システムは重要な部品が欠損する等の大きな構造変化を引き起こす故障に対処できないことを示している。ここでいう重要な部品とはその部品の動作を他のアクチュエータで補完できない部品のことであり、例えば、ハロゲンランプの不点灯、ドラムの傷、転写、分離チャージャの断線等は種類の故障に含まれる。



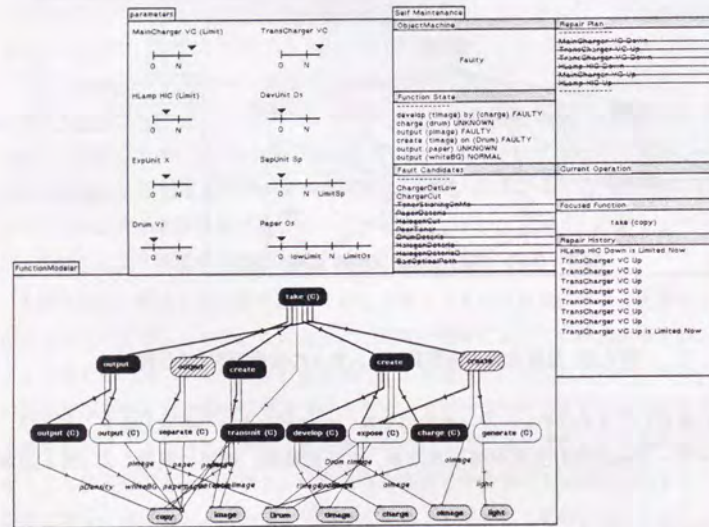


図 7.20: 故障診断 / 修復計画システムの実行画面 (実行例 3)

## 7.5 考察

本節では前節の実験の結果をもとに、本研究で構築した制御型自己修復実験機、および、一般の制御型自己修復機械システムの特徴と問題点について考察する。

## 7.5.1 実験の特徴

前節で述べた自己修復実験は、意図的に故障を与えており、自然に故障や劣化が発生する状況と多少異なる可能性もあるが、実験機の基本的な自己修復性の確認という目的は達成できた。ただし、実用面での有効性を検証するためには、故障に関する十分な知識量を記述し、自然な劣化や故障が発生するのに十分な時間の試験を必要とする点で、本研究と異なる実験が必要となると考えられる。

## 7.5.2 制御型自己修復機械の有効性

前節で述べた実行例 1、2 より、制御型自己修復実験機は機能保全という考え方に基づき自己修復性を発揮することができることを確認できた。特に、第 4 章～第 6 章で述べた手法による知的判断部を持つことで、故障の状況に応じて柔軟に修復を実現できた。すなわち、実行例 1 では故障「ハロゲンランプの劣化」に対し、アクチュエータの可操作範囲の変化に応じて、ハロゲンランプの出力操作のみでなく、一般的には利用されないが、因果的に関係のあるメインチャージャの出力操作も合わせて利用することにより修復を実現できた<sup>9</sup>。また、実行例 2 では対象の構造が一部変更される故障に対しても修復を実現できた。

しかし、実行例 3 が示すように、本実験機では重要な部品が働かなくなるというような故障には対応できない。これは、知的判断部の限界ではなく、修復方法としてアクチュエータの制御を用いるという制御型自己修復機械の修復実現方法の本質的な限界を示している。この問題に対応するために、修復実行機能をより広範囲の故障に対応可能なものにしなければならず、第 8 章で述べる機能冗長型の修復実現方法が有効と考えられる。

本研究では、センサ、アクチュエータ、計算機は故障しないと仮定した。ただし、アクチュエータに関しては例えば出力を上昇させたにも関わらず出力が低下してしま

<sup>9</sup> 転写チャージャを用いても同様の修復を実行できることを確認した。



うというような、操作に対して逆の特性を示さない限り、知的判断部により対処可能である。しかし、センサが正しくない情報を出力する場合は正しい故障診断/修復計画を行なえない。この問題に対しては、各センサの値を対象モデル上で互いに比較する方法や、テスト作業を生成し、対象機械上で実行することにより、疑わしいセンサ出力を同定することが可能になると考えられる。例えば意図的に白紙を出力させ、各センサ値の変化の有無から故障したセンサの同定を行なうことも可能である。これらの方法によりセンサの故障に対処することは今後の課題の一つである。一方で、計算機の故障に関しては現在のところ対処の方法がなく、自己修復機械システムとして計算機自身の故障診断/修復を含めることは困難であると考えられる。この問題に対して、フォールトトレラント・コンピュータの研究 [Watanabe86] や知的判断部の分散化の研究 [Koike92b] 等の手法を利用することが考えられる。

### 7.5.3 制御型自己修復機械と制御技術の相違

制御型自己修復機械と制御技術は入力としてセンサによる測定値を、出力としてアクチュエータの操作を行なうという点で具体的動作としては等しいが、その目的、操作を導出する方法が異なっていると考えられる。すなわち、本研究の目的である自己修復性を発現可能であれば、従来の制御技術を利用することも考えられる。

式(7.5)(式(3.1)の再掲)に示す制御モデルにおける制御点  $i_i$ 、制御目的パラメータ  $o_i$ 、伝達関数  $f$  は、制御型自己修復機械の操作パラメータ、機能の発現、および、対象モデルにそれぞれ相当する。しかし、第3.3.1項で述べたように制御型自己修復機械は故障の状況に応じて動的に伝達関数、制御点、制御ループを決定するという特徴を持つ。

$$(o_1, o_2, \dots, o_n) = f(i_1, i_2, \dots, i_m) \quad (7.5)$$

制御技術で本章で提案したような自己修復機械を実現する場合、扱う変化の範囲が一般の制御に比較して大きいと考えられる保全問題に対して、故障に対応可能な柔軟な伝達関数の記述可能であるのかという疑問が残る。具体的には、以下の四点に関して伝達関数の記述が困難であると考えられる。

#### 1. 機能トレードオフの実現

機能トレードオフを実行すると、式(7.5)の制御目的パラメータが変更される。

このような制御目的の選択に関する議論は制御技術の範囲外の問題である。

#### 2. 制御範囲の限界

例えば、本章で例題にした複写機において、「画像濃度を一定にする」という目的に対しても、実行例1で示したように、ハロゲンランプにより調整可能な場合と、ハロゲンランプ単一では調整不能であり、メインチャージャ等複数の制御点により調整しなければならない場合がある。これら両者の場合には制御技術における伝達関数が異なると考えられる。この制御範囲の問題を一般化すると、 $m$  個の制御点に関して制御範囲の上限、下限により伝達関数を変更せるとすると合計  $3^m$  個の伝達関数を用意しなければならないことになる。

#### 3. 故障による伝達関数の変化

一般に、構造変更を引き起こす故障により伝達関数に変化する可能性がある。このとき、故障による変化に対応した伝達関数を用意すれば制御を行なえるが、正常時の伝達関数では制御を適当に行なえない場合があると考えられる。この問題に対応するためには、制御系設計時に発生する可能性のある故障を予測し、それに対応して伝達関数を記述しなければならないが、この方法では保全の非繰り返し性に対応できない。この問題、すなわち、制御技術では取り扱えず制御型自己修復機械で扱える例を整理することは今後の課題である。

#### 4. 機能冗長の利用

本章では直接関係ないが、第8章で述べるように機能冗長型自己修復機械は機能冗長系を発現後、制御型自己修復を実行する。制御技術でこれを実現するためには、各機能冗長系発現後の伝達関数を設計時にあらかじめ記述しなければならない。

以上のような問題点があるため、上記2.～4.に対応するためには伝達関数を変更する故障パターンが  $x$  個、利用可能な機能冗長系が  $y$  個ある対象機械の場合、設計時に伝達関数を最大  $x \times y \times 3^m$  個記述しなければならない。このとき、故障という不定性を持つ現象に対して各伝達関数を正しく記述可能であるか、これだけの数の伝達関数を記述可能であるか、さらに、記述された伝達関数を正しく使い分けることができるかという問題がある。特に、伝達関数の使い分けの問題は故障同定の問題と本質的に同等であり、制御技術では議論されていない問題である。

本研究で提案した手法は、本章で示したようにモデルベース推論を利用することにより、定性的な範囲で以上の問題を解決する。このため、修復実行能力自体は、両



者ともあらかじめ設計時に与えられた制御点の制御範囲により決まる制御可能空間に限定されるという意味で本質的に等しい能力を持つといえるが、伝達関数を全て記述し尽くすという制御技術のみを用いた手法が実現不可能であると考え、本研究の手法がより広い範囲で修復操作を実現可能な手法であると結論付けることができる。

#### 7.5.4 故障診断 / 修復計画システムの有効性と課題

##### (1) 故障診断 / 修復計画システムの特徴

本実験機において、知的判断部は故障診断、および、修復計画の導出を行なうことができた。ここで重要なことは、本手法はモデルベースト推論手法に基づき、故障とその修復方法の対応という知識を明示的に持たずに、因果関係に基づき動的に修復方法を導出、実行できたことである。このことは、保全の特徴である不定性、非線り返し性に本知的判断部が対応できることを示している。さらに、この知的判断部の特徴として以下の二点を挙げることができる。

- 知的判断部は汎用性を持ち、対象モデルを入れ換えることにより、様々な対象に対して故障診断、修復計画の導出を行なうことができる。
- 修復計画部は修復シミュレーションにより、副次的影響を評価して修復操作を導出するため、故障に対して適当な修復計画を導出することができる。

またここでは、システム全体の基本的動作を確認することを目的とし、簡単のため故障診断部で単一故障仮説を用いたが、第5章で提案した単一故障発生過程仮説を用いた故障診断も同一のシステムで行なうことができる。両者の違いは現象知識の内容のみであり、これにより広い範囲の故障に対応可能な知的判断部に拡張可能である。

##### (2) 故障診断 / 修復計画システムの課題

本故障診断 / 修復計画システムの課題として以下の三点を挙げることができる。

1. 劣化に対する評価
2. 知識の記述に関する問題
3. 幾何情報の取り扱い

#### 劣化に対する評価

本章で述べた自己修復の範囲では、故障の主な原因である劣化現象を扱わなかった。本自己修復システムにおいて劣化の取り扱いは二つの意味を持つ。一つは、故障診断時における故障原因同定のための情報として劣化状態を推定することであり、もう一つは修復計画時において劣化への影響を評価して修復操作を選択することである。例えば、ドラムに過大な電荷を与えるとドラムが劣化することからメインチャージャの電圧を上げる操作は優先順位が低い。

このように自己修復機械にとって重要である劣化を扱うためには、知識やデータの収集が大きな課題として残されているが、基本的には第5.3節で述べた枠組が有効であると考えている。これにより、本章で提案したショートタームの自己修復機械を拡張し、機械のライフサイクル全般を管理するロングタームの観点も含めた総合的な自己修復機械を構築可能であると考えられる。

#### 知識の記述に関する問題

知的判断部を有効に動作させるために重要な点は、知識の記述に関する問題である。すなわち、

1. 対象モデルが実際の機械の性質を十分に表しているか。
2. 実際の機械上で起こりうる故障を予測するに十分な物理現象の知識が収集されているか。
3. 実際の機械で測定される定量的な値と計算機上の記号的な値をいかに無矛盾に対応付けるか。

という点である。実際、修復方法が無いためではなく、知的判断部が故障診断や修復計画に失敗した場合は上記の三点に問題があると考えられる。これらのうち1.、2.は知識獲得問題として盛んに議論されている問題であり、理論的にうまく行なう方法は今のところ無い。特に、故障時の1.の問題は保全の特徴である不定性と、また、2.は多様性と関係する問題である。

しかし、本研究の原理原則的な知識と対象モデルを定性物理の枠組により記述するという方針は、例えば、ヒューリスティックスを多量に集めるという手法よりも知識獲得や知識の正当性の評価が容易であると考えられる。すなわち、原理原則的知識はある程度体系的に収集可能であり、対象モデルは設計情報を利用することにより比較的容易に記述可能であると考えられる。



しかし、上記の3. は若干性質が異なり定性物理にとって本質的な問題である。この点については第5.3節でも述べたが、実際に実験システムを動作させた場合にも、微妙なセンサ値の変化で正しいモデルの構築に失敗することが発生した。例えば正常値の範囲を1.68-1.84(V)とした時に、1.67(V)というセンサ値は正常値よりも小さいと判定されるが、他のセンサ値との関係、機械の状態、センサの誤差等からこれが正しい定性値であるとは必ずしもいえない。この問題に対処するためには、第5.3節で述べたファジィ定性物理を用いて知的判断部を構築することが有効であると考えられ、このファジィ定性物理を用いた実験を行なうことが今後の課題の一つである。

#### 幾何情報の取り扱い

機械においては幾何的な関係から干渉等が発生し故障の原因になるという意味で幾何情報を扱うことは重要である。また、本実験機の出力画像情報のように面としての情報やさらには三次元的空間情報を利用しなければならない場合もある。このような場合には、何らかの記号表現に変換し本研究の枠組で利用する必要がある、画像処理技術等を利用した記号化の枠組が必要である。このとき、センサ情報の抽象化技術として、ニューラルネットワーク [Amari89] の利用も考えられる。

近年、幾何情報を記号的に処理し、推論を行なおうとする「幾何推論」と呼ばれる分野の研究が行なわれているが [Umeda90]、本研究の手法を拡張し幾何推論を行なうためには様々な問題が残されている。桐山 [Kiriya91] はメタモデルと呼ばれる機構により様々な観点からのモデルを統合化する方法を提案しているが、幾何情報を扱うモデルをメタモデルのような方法により本研究の対象モデルと統合化することが一つの有望なアプローチとして考えられる。

#### 7.5.5 推論部と機械部の接続

定性物理に基づく知的判断部と実際の物理的存在である機械部を接続する点に幾つかの問題が発生した。第一の問題は前項で述べた定量情報の定性化の問題である。第二の問題は、修復計画部で導出される修復操作が定性的であるということであった。このため本研究では単一アクチュエータの微小操作の繰り返しという方法で修復操作を実現したが、この方法は、修復に時間がかかり、また、操作がループを描いてしまいう修復が実現できないような場合があるという問題点が残されている。

この問題に対して、下村ら [Shimomura90] は既に実行した修復操作を事例として記

述し、その事例を組み合わせることにより複数アクチュエータの同時操作を含む柔軟な修復計画手法を提案している。また、例えば図7.6の(e)において、操作パラメータに対する目的パラメータ特性の単調性を仮定すればパラメータの操作に対する定量的な変化を測定しながら、山登り法により正常状態を探索することが可能になる。

#### 7.5.6 機械部の設計方法

第7.1節では、制御型自己修復機械の機械部の設計方法について述べた。実際の実験機では、既存機種の改造であり、また、センサ、アクチュエータの配置に関しては、技術的、コストの要因が支配的であったため、この設計方法の有効性は十分に検証することができなかった。しかし、先に述べたように制御型自己修復機械の限界は主に修復実行能力によって規定されているので、第7.1節で述べた方法に基づき、できるだけ多くのアクチュエータを対象系に付加すること、および、対象を適当な因果的構造にすることにより、アクチュエータに関する冗長性を高めることが有効であると考えられる。

しかし、第7.1節で述べた設計方法の一つの問題点は、対象系において発生し得る故障の影響を十分に考慮していない点である。この点に関して二つの対処方法が考えられる。第一は、従来の信頼性工学に基づき、各部品、センサ、アクチュエータ等に関する信頼度情報を利用することである。第二の方法は、知的判断部の現象知識を利用し、予想される故障を同定し、これらの故障の影響を受けにくい構造、アクチュエータの配置や、故障をより識別可能にするためのセンサの配置を考える方法である。ただし、この場合、あり得る全ての故障の可能性を提示してしまうので、有意義な対応を行えない可能性がある。Gershenson ら [Gershenson91] のサービス性を目的とした設計は、故障データベースを構築、利用して設計対象の保全に対する評価を設計時に行なうものである。



## 7.6 第7章のまとめ

本章では、制御型自己修復機械の実験機の設計、および、動作実験を中心に、制御型自己修復機械の特徴と問題点を整理した。まず、第7.1節で制御型自己修復機械の機械部の設計方法について、センサ、アクチュエータの配置問題、対象構造の決定問題という観点から整理した。第7.2節では、修復実行機能の実現方法について述べ、本研究では単一アクチュエータの微小量操作の繰り返しにより修復を実行すること、および、推論の単純化のために修復計画と修復実行を交互に行なうことを述べた。第7.3節では、実際に構築した実験機について述べ、第7.4節で、実験機を用いた自己修復実験の結果を例示した。これにより、本実験機が有効に動作する範囲や修復ができない範囲が明らかになった。最後に、第7.5節で実験の結果をもとに、制御型自己修復機械システムの特徴と問題点を考察した。この結果、本実験機は有効に自己修復を行なったが、重要な部品が故障する場合にはその修復実行能力の問題から対処不能になることが明らかになった。機能冗長型の自己修復機械によりこの問題に対応可能であると考えられるが、この点に関しては第8章で議論する。

## 第8章

### 機能冗長型自己修復機械設計方法論



本章では、機能冗長型自己修復機械の設計、および、実験について述べる。まず、機能冗長型自己修復機械の機械部の設計方法として、設計対象に対して機能冗長系を付加する方法について述べる。次に、この方法にしたがって設計した機能冗長型自己修復実験機、および、実験結果について述べる。最後に、実験結果に基づき、機能冗長型自己修復機械の特徴と問題点について述べる。

## 8.1 機能冗長型自己修復機械の設計方法

### 8.1.1 機能冗長型自己修復機械設計の要求項目

第7章の実験結果から、制御型自己修復機械はその修復実行能力に限界があるものの、知的判断部や基本的な機械部の構成に関しては自己修復能力を持つことが明らかになった。そこで、制御型自己修復機械に機能冗長系を付加することにより修復実行能力を高めた機能冗長型自己修復機械を構築することが有効であると考えられる。すなわち、機能冗長型自己修復機械に要求される設計項目は、第7.1節で述べた設計項目に加えて、機能冗長系を機械部に付加することである。

そこで本節では、機能冗長型自己修復機械の機械部の設計方法として、制御型自己修復機械の設計方法と唯一異なる点である、機能冗長系を機械部に付加する設計に関して設計者を支援する方法を考察することにする。これにより、機能冗長型自己修復機械の修復実行機能を実現することができる。

### 8.1.2 機能冗長設計の方法

ここで、メカトロニクス機械を対象とした設計は次の三段階によってなされと考える。

#### 1. 設計対象の構造決定

まず、概念設計段階として設計対象に必要なとされる機能とそれを実現する基本的構造を決定しなければならない。この時同時に、機械がとり得る動作モードも決定される。FBSダイアグラム上で言えば、この構造決定は機能階層構造、および、挙動・状態レベルのネットワークを無矛盾、十分に決定することに対応する。また、動作モードは各機能に付加された mode タグにより表される。

#### 2. パラメータの決定

1. で決定された基本構造と部品の定性的な特性に基づき、各部の定量的な値、配置を決定する。具体的には、ドラムの大きさ、電源の出力等を決定することである。

#### 3. 対象機械の制御シーケンスの決定

メカトロニクス機器は、一般に制御プログラムにより動作モードを切り替えて複雑な動作を行なう。そこで、1. で決定された基本構造と2. で決定された定







## (b) 同種の機能の利用

注目機能階を同じ種類の機能で置換すれば、機能冗長系を構成できる可能性がある。

図8.1の例でいえば、dで示される「電気エネルギーを出す」機能をeで示される「電気エネルギーを出す」機能以下の部分で代替することを試みることに相当する。

## (c) 新たな機能階層の構築

一般に、機能のプロトタイプ知識には複数の機能展開方法が記述されている。そこで、注目機能に対して現在対象上で用いていない別の機能展開方法を対象系上で実現することを試みる。そして、展開された各部分機能に対して(a)～(c)を適用することにより機能冗長を実現する。

図8.1の例でいえば、fで示される「電気を伝達する」機能を展開し、「電気を回転に変換」し、「回転を伝達」し、さらに、「回転を電気に変換する」という3種類の部分機能に展開することに相当する。

これらのうち、(a)は与えられた機能構造をそのまま使用するのに対し、(b)、(c)では、機能構造も操作する。

## 3. 機能冗長候補に対する実現条件の導出

2.で得られた機能階層構造を実現する挙動、実体を設計対象上に存在させなければならない。すなわち、改良前の設計対象モデル  $m^0$  に基づき2.で導出されたモデルの部分  $m^* = (n_f^*, n_{bs}^*, R_{fb}^*)$  を含む無矛盾な設計解  $m^R (m^* \subseteq m^R)$  を導出しなければならない。

$n_{bs}^0$  と  $n_{bs}^*$  を比較することにより、 $n_{bs}^*$  は式(8.1)に示すように、三つの部分に分けることができる。

$$n_{bs}^* = n_{bs}^g \cup n_{bs}^i \cup n_{bs}^c \quad (8.1)$$

$n_{bs}^g$  は設計対象上  $n_{bs}^0$  で実現されている挙動・状態の部分(gはgoodの略)、 $n_{bs}^i$  は設計対象と矛盾はしないが、実現条件が満たされていない挙動・状態(iはinadequateの略)、および、 $n_{bs}^c$  は設計対象と矛盾する挙動・状態(cはconflictの略)である。このとき、動作モードを設計対象上で適切に切り替えることにより、機能冗長発現時に  $n_{bs}^i$ 、 $n_{bs}^c$  が発現し、かつ、正常時に  $n_{bs}^g$  が発現しないように対象を設計しなければならない。

## 4. 実現条件を満たすための設計

3.で得られた実現条件、動作モードの条件をもとにして、設計者は概念設計を行なう。この部分は設計者の主観的判断に任されている部分であり、結果として機能冗長系を含み無矛盾なFBSモデルを作成しなければならない。

## 5. 機能冗長設計解の評価

4.で導出された機能冗長設計解候補の中から、採用する機能冗長系を選択しなければならない。ここでは、式(8.2)、(8.3)に示す二つの評価式を導入し、機能冗長設計解を順位付けすることにする。

$$Redundancy(f^*) = \frac{n(n_{bs}^R(f^*) \cap \overline{n_{bs}^0})}{n(n_{bs}^R(f^*))} \quad (8.2)$$

$$Robust(f^*) = 1 - \frac{n(n_{bs}^R(f^*) \cap n_{bs}^0(f^*))}{n(n_{bs}^0(f^*))} \quad (8.3)$$

ここで、 $n(n_{bs})$  は挙動、状態ネットワーク  $n_{bs}$  に含まれている要素数であり、 $n_{bs}(f^*)$  は、ある機能  $f^*$  に関係する挙動、状態ネットワークの部分ネットワークである。このとき、 $Redundancy(f^*)$  は注目する機能  $f^*$  の機能冗長系実現のために付加しなければならない要素の割合を示す。この  $Redundancy(f^*)$  が1に近ければより部品冗長的であり、0に近ければより機能冗長的であると言える。 $Redundancy(f^*)$  は定性的にコストに比例すると考えられる。また  $Robust(f^*)$  は機能  $f^*$  に関して、元々の設計における実現構造と機能冗長系における実現構造との間の一致していない度合いを表す。両者の共通部分に故障が発生した場合は付加した機能冗長系を利用不可能であると考えられるので、 $Robust(f^*)$  は定性的には機能冗長系の信頼性に比例すると考えられる。

機能冗長候補を選択する場合は、設計者の考え方により  $Redundancy$  の低いものを重要視するか、 $Robust$  の高いものを重要視するかが変わってくる。また、これら二つの評価パラメータは独立ではなく、定性的に反比例関係にあると考えられる。

## 8.1.3 FBS モデラーを用いた機能冗長設計支援

## (1) FBS モデラーを用いた機能冗長設計支援システム

ここでは、前項で述べた方法に基づき、第4.4節で述べたFBSモデラーを用いて設計者の機能冗長を付加する概念設計を支援する実験的なツールを構築した。図8.2に



本システムの構成を、図 8.3 に本システムのアルゴリズムを示す。

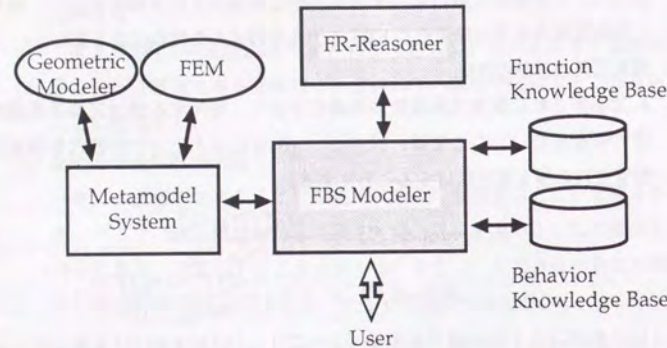


図 8.2: 機能冗長設計支援システムの構成

図 8.2 内の FR-Reasoner に対する入力、設計対象に関する FBS モデル  $m^0 = \langle n_f^0, n_{bs}^0, R_f^0 \rangle$  であり、第 4.4 項で述べた機能知識ベース、挙動知識ベースが十分な知識量を持って構築されているものとする。

FBS モデラー上では、 $n_f^0$  は機能ノード  $f$  を階層関係、もしくは、動詞-目的語関係により接続したネットワークとして表現されており、機能ノード  $f$  は機能知識ベース内の対応する機能プロトタイプ  $p_f$  のインスタンスになっている。クラス・インスタンス関係をここでは、 $p_f = \text{class}(f)$  と表すことにする。各機能プロトタイプ  $p_f$  は第 4.4 項で述べたように、機能名  $n$ 、展開方法  $d$  の集合  $D$ 、および、実現挙動としてのフィジカル・フィーチャー  $p$  の集合  $P$  から構成され、 $p_f = \langle n, D, P \rangle$  と書ける。このとき、各  $d$  は、 $p_f$  のネットワークとして表現され、各  $p$  は後述する  $p_v$  のネットワークとして表現される。

一方、挙動・状態レベルの記述  $n_{bs}^0$  は、第 4.4 項で述べたようにビュー・ノード  $v$  の依存関係を表すネットワークとして表現される。このとき、あるビュー  $v$  は挙動知識ベース内の挙動プロトタイプ  $p_v$  と  $p_v = \text{class}(v)$  の関係にある。ここで、ある機能  $f$  に関して機能プロトタイプ  $\text{class}(f)$  内の実現挙動  $p \in P$  を指定すると、 $p$  が挙動・状態レベル上にインスタンス化されビュー・ネットワーク  $n_{bs}(f)$  が生成される。

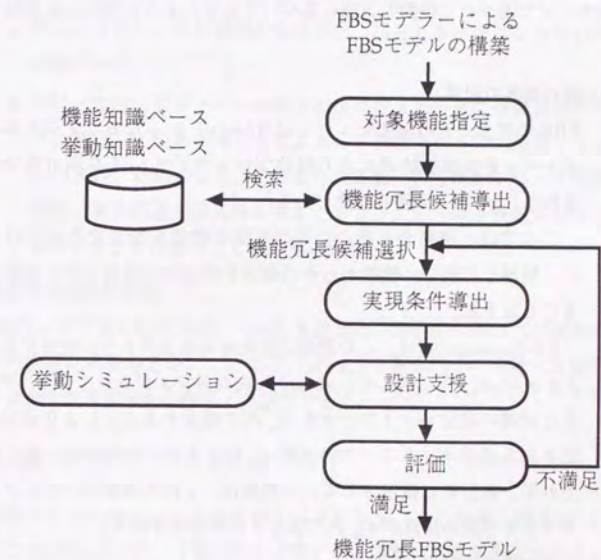


図 8.3: 機能冗長設計支援システムのアルゴリズム



このとき、 $f$  と  $n_{bs}(f)$  を結び付ける関係  $r_f$  が  $R_f$  内に記述される。

以上の FBS モデルに対し、本システムは次のように機能冗長設計を支援する。

1. 冗長性を付加したい設計対象の機能とその部分機能構造に注目  
機能階層  $n_f^0$  内の機能  $f^*$  を指定する。このとき、 $n_f^0$  内で  $f^*$  を頂点とする部分機能階層ネットワークを  $subnet(f^*)$  と表すことにする。

2. 注目機能に対する機能冗長の候補を導出  
212ページで述べた三種類の方法に基づいて以下のように機能冗長候補を導出する。

#### (b) 潜在機能の利用

FBS モデラー上のあるビュー  $v$  は  $class(v) \in p$  なるフィジカル・フィーチャー  $p$  を実現挙動  $P$  に含む機能プロトタイプ  $p_f$  を発現可能である。すなわち、 $p_f$  が  $v$  の潜在機能である。

ここでは、挙動からそこに発現可能な機能を探索するのではなく、逆に、発現して欲しい機能からその機能を潜在的に発現可能な挙動を探索することにする。

FR-Reasoner では、この機能冗長候補を以下のように導出する。各機能  $f \in subnet(f^*)$  に対して、 $class(f)$  内の各フィジカル・フィーチャー  $p \in P$  と同種の部分ネットワークを  $n_{bs}^0$  内で探索することにより導出する。導出される部分ネットワークの候補  $n_{bs}$  は  $p$  との一致度が高い順に順位付けされて、設計者に提示される。一致度は、 $p$  内の各挙動プロトタイプとプロトタイプ間の関係が  $n_{bs}'$  内で満たされる割合を示す。

#### (c) 同種の機能の利用

同種の機能は、各機能  $f \in subnet(f^*)$  と同種の機能を  $n_f^0$  内で探索することによりインプリメントされる。

#### (d) 新たな機能階層の構築

これは設計者による各機能  $f \in subnet(f^*)$  のプロトタイプ  $p_f = class(f)$  において、現在使用しているものと異なる展開知識  $d$  を利用することとしてインプリメントされる。

### 3. 機能冗長候補に対する実現条件の導出

2. で導出された機能冗長候補を設計者が選択することにより、システムはその

候補と初期の設計対象モデル  $m^0$  を比較し、挙動・状態ネットワークにおける共通部分  $n_{bs}^0$ 、満たさなければならない部分  $n_{bs}'$  を設計者に提示する。

#### 4. 実現条件を満たすための設計の支援

3. の条件を満たすための設計は、一般の FBS モデラーを用いた概念設計と同様である。ただし、以下の点については FBS モデラーが設計者を支援することが可能になっている。特に、挙動シミュレーションは、ビュー・ネットワークの無矛盾性の管理や要求機能の実現性の検証に有効である。

- 機能冗長系と正常状態間の共通部分、満足しなければならない挙動、矛盾挙動の提示。
- 機能冗長系を発現させるための前提条件を設計者が各機能部分の mode タグに整理して記述することによる、各動作モードでの挙動シミュレーションの実行。これにより、前提条件、挙動・状態ネットワークの無矛盾性の管理、要求機能の発現性の検証、および、正常時と機能冗長時の挙動の干渉チェックを行なうことが可能である。

#### 5. 機能冗長設計の評価

215ページで示した式 (8.2)、(8.3) を計算し、設計者の機能冗長候補の利用のための判断材料を与える。ここでは、 $n(n_{bs})$  をある部分の挙動・状態ネットワークに含まれるビューの数により表現する。

### (2) 実行例

(1) で述べた FR-Designer の実行例を以下に示す。ここでは、図 8.4 に示す自動車のエンジン前部において、「蓄電池を充電する」機能に冗長系を付加することを考える。図 8.5 に本機能冗長設計部に入力として与えられた自動車前部の FBS モデルを示す。

この FBS モデルに対し機能冗長候補の探索を行なった結果を図 8.6 に示す。この例はモータの潜在機能として「回転による発電」機能の発現可能性を示している<sup>1</sup>。図中で斜線で示されたノードは初期モデルと機能冗長系で共通して使用されるものであり、黒塗りされたノードは機能冗長系として発現させる必要がある現象、状態を示している。これから、モータ上で発電現象を引き起こす必要があり、そのためにはモータ

<sup>1</sup>ここでは、スタータ・モータとして直流モータを利用しているとした。











## 8.2 機能冗長型自己修復機械における修復の実現

機能冗長型自己修復機械における修復実行機能の実現方法は、修復計画と修復実行をインタラクティブに行なう点、および、アクチュエータによる修復操作を微小操作の繰り返しにより実現する点で第7.2節で述べた制御型の修復実行方法と同様である。唯一異なる点は、修復操作として機能冗長系への切り替えを実現しなければならないことである。メカトロニクス機器におけるこの機能冗長系への切り替えは制御シーケンス・プログラムの変更により実現することができる。前節で示した自動車の発電機能冗長系の例でいえば、スイッチ、オーバーランニング・クラッチに自動制御機構を付加することにより、正常時にはエンジン回転中スイッチ off、オーバーランニング・クラッチ off であるものを、機能冗長発現時にはスイッチ on、オーバーランニング・クラッチ on に変更することに相当する。

制御シーケンス・プログラムを変更する方法として、設計時にあらかじめ機能冗長系を利用する制御シーケンスを作成しておき、プログラムの切り替えにより実現する方法と、修復実行時に動的にシーケンスを組み替えて新たなシーケンスを生成する方法が考えられる。後者の方法を用いる場合、一種の自動ソフトウェア生成の問題になる。自動ソフトウェア生成の研究は近年盛んになっている [Ohsuga91, Saeki91, Honiden91] が、実用的な意味では今後の発展を待たなければならない。

そこでここでは、設計時に機能冗長系利用の制御プログラムを用意し、使用時に切り替える方法をとることにする。このとき、一つの対象系上に機能冗長系を  $n$  個付加した場合、全ての組合せを考えて制御プログラムを作成すると、機能冗長系発現の時間的順序関係を考えない場合においても  $2^n$  通りのプログラムを用意しなければならない。これだけの数のプログラムを用意することは現実的な意味で困難であると考えられるので、ここでは第6.3節で述べたように複数の機能冗長系を同時に利用しないことにする。また、設計時に制御プログラムを作成する場合に全ての故障状態を予測することは困難であるので、機能冗長系として使用する部分に故障が存在する場合はその機能冗長系を利用しない。複数機能冗長系の同時利用や故障状態に応じて柔軟に機能冗長系の利用を実現するためには、修復実行時に動的に制御シーケンスを決定することが必要になると考えられる。この問題は今後の課題とする。

以上の理由により単一機能冗長系を用いる機能冗長型自己修復機械における修復計画、実行のアルゴリズムを図8.8に示す。このアルゴリズムは図7.5で示した修復計

画、実行アルゴリズムを拡張したものである。

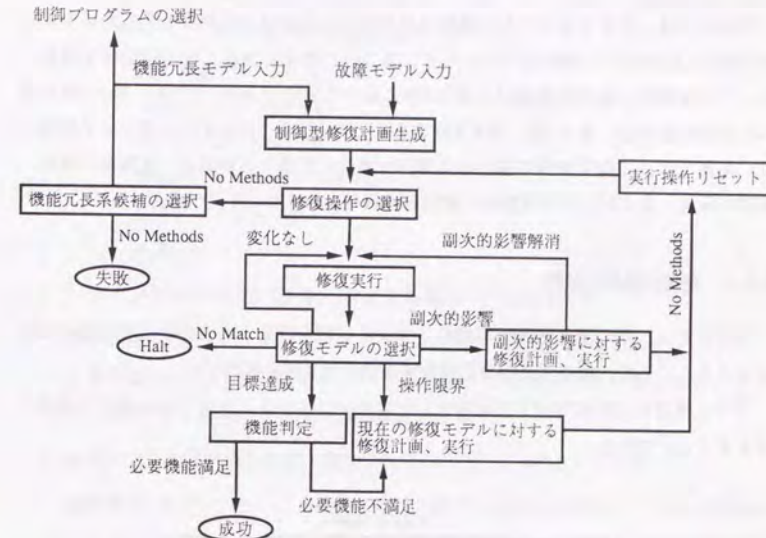


図8.8: 機能冗長型修復計画、実行のアルゴリズム



### 8.3 実験機的设计

本研究では、今まで述べてきた機能冗長の有効性の検証と問題点を明らかにするために第8.1節で述べた機械部の設計方法に基づいて簡単な機能冗長型実験機を構築した。この実験機の知的判断部は、第4.6節に述べた保全用FBSモデル、第5.2節で述べた故障診断手法、第6.3節、第8.2節で述べた修復計画、実行手法に基づいて構築した。本節では、この実験機に関する具体的な機能冗長系の設計結果、実験機の構成、知識の記述、および、知的判断部と機械部の接続方法について述べる。

#### 8.3.1 機能冗長系の設計

本研究では、第7章と同様複写機のドラム周辺部を対象とし、既存の複写機を改造することによって、機能冗長型自己修復実験機の機械部を構築した。

ドラム周辺部(図8.9参照)を対象とした場合、以下のような幾つかの機能冗長系を考えることができる。

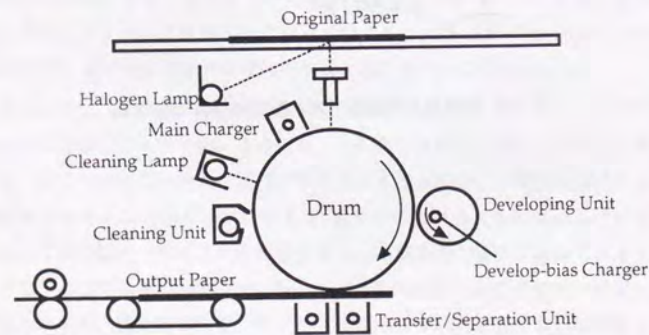


図8.9: 複写機のドラム周辺部

1. 除電ランプによる「ドラム除電」機能をハロゲンランプにより代替

選択理由 除電ランプとハロゲンランプは同種の「発光」現象を示す。

### 8.3. 実験機的设计

必要な動作モード制御 除電ランプ不点灯の場合、一複写プロセスあたりドラムを二回転させ、一回転目は除電を目的とし、以下の条件で複写動作を行なわせる。二回転目は通常の複写動作を行なう。

原稿 白紙  
ハロゲンランプ出力 on  
メインチャージャ off  
現像バイアスチャージャ on  
転写チャージャ off  
給紙なし

2. ハロゲンランプの「原稿照射」機能を除電ランプにより代替

選択理由 除電ランプとハロゲンランプは同種の「発光」現象を示す。

必要な動作モード制御 除電ランプから原稿への光路を作る必要がある。さらに、1.と同様の二回転の複写制御を行なう。

3. 除電ランプの「ドラム除電」機能をメインチャージャ<sup>4</sup>により代替

選択理由 メインチャージャの潜在機能の利用。必要な機能は「ドラムの表面電位を0にする」ことであることから、メインチャージャから負の電荷を与え、表面電位をキャンセルすることが可能であると考えられる。

必要な動作モード制御 一複写プロセスあたりドラムを二回転させる。一回転目は除電を目的とし以下の条件で動作させ、二回転目は通常の複写動作を行なう。

メインチャージャ 極性を逆にし、on  
現像バイアスチャージャ on  
転写チャージャ off  
給紙なし

4. メインチャージャの「ドラム帯電」機能を転写チャージャにより代替

選択理由 転写チャージャ上でメインチャージャと同種の「放電」現象が生起している。

<sup>4</sup>同様のことは、除電ランプ→転写チャージャ、除電ランプ→分離チャージャを用いても実現可能であるとえられる。



必要な動作モード制御 一複写プロセス当たりドラムを二回転させる。一回転目はドラムを帯電させることを目的として以下のような条件で動作させ、二回転目は除電ランプ off で通常の複写動作を行なう。

ハロゲンランプ off

現像バイアスチャージャ on

転写チャージャ on

除電ランプ off

給紙なし

第8.1.3節で述べた FBS モデラーによる機能冗長設計方法を用いると、上記の各候補、その選択理由、および、必要な動作モードを知ることができる。例えば、4. に述べた例を導出した FBS モデラーの実行画面を図 8.10 に示す。図 8.10 中で斜線の機能ノードが機能冗長系、斜線の挙動ノードが機能冗長発現時に生起する現象を、また、黒塗ノードが機能冗長発現時に使用されないメインチャージャに関連する挙動、状態を示している。

しかし、これらで導出されるのは定性的な範囲の条件であるため、2. で述べた光路を変更することやその方法は導出しない。また、4. の逆としてメインチャージャにより「転写」機能を代替させるという候補も導出されるが、これは紙送り系の幾何的拘束条件から非常に実現困難な候補であるということは分からない。これらの拘束条件はパラメータ値の決定や制御シーケンスの決定段階のより詳細な情報によって決定されるものである。実際、ここで列挙したものは導出された候補の一部であり、導線同士の交換、各種トランスの置換等多数の候補が導出され、その中から適当なものを選択する段階を支援することも今後の課題となる。

第8.1節では触れなかったが、同種の部品の交換を考えるとさらに機能冗長性を増やすことができる。例えば、上記のように転写チャージャの機能をメインチャージャにより代替することはできないが、この両者が共通の部品でできていれば、断線した転写チャージャとメインチャージャを交換し、上記 4. を利用することにより機能冗長性を実現できる。ただし、この交換は使用者による交換、もしくは、保全ロボットによる自動交換により実現する必要があると考えられ、その意味では本研究の自己修復機械の枠組から出るものである。

以上、上記の機能冗長候補と部品交換を考えるとドラム周辺部は例えば図 8.11 のようになる。図 8.11 において大きな設計変更は除電ランプから原稿への光路を付加する

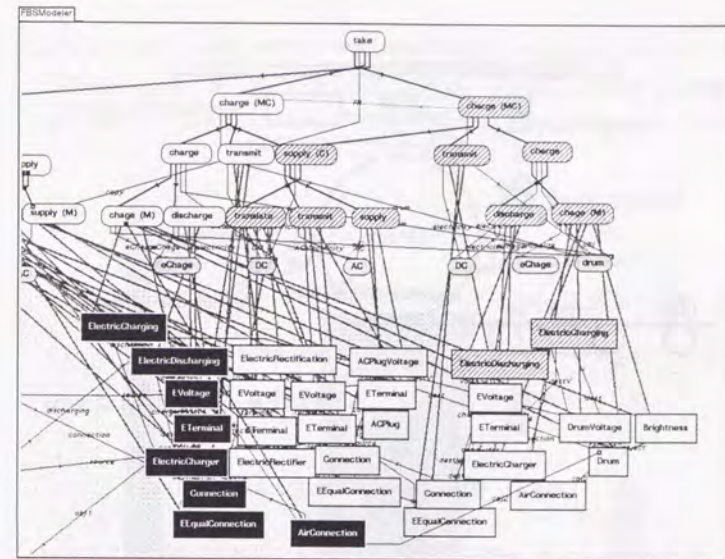


図 8.10: メインチャージャの機能を転写チャージャで代替させる機能冗長系

こととメインチャージャの極性を変更するトランスを付加する程度である。このように、機能冗長系をメカトロニクス機械で実現すると、一般に機械構造の変更は大きくなく、制御プログラムの変更が大きな割合を占めるといえる。この意味で、機能冗長系の実現は容易である。

### 8.3.2 実験機の構成

本研究では、前項で述べた機能冗長系を実現する実験機の構築を試みたが、技術的、経済的制約から全ての機能冗長系を付加するに至らなかった。その理由として、既存の機械を改造する方針を取ったことや開発やテストに十分な時間を持てなかったことが挙げられる。結果として前項の 1. ~ 4. を以下のように扱った。

#### 1. 除電ランプによる「ドラム除電」機能をハロゲンランプにより代替

この機能冗長系は実現した。このとき、特に付加、または、改良する部品は必



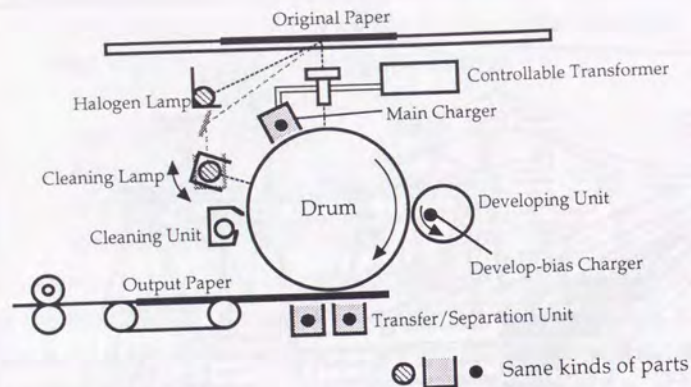


図 8.11: 機能冗長を付加したドラム周辺部

要ではなかったが、制御シーケンスプログラムを新たに作成する必要があった。

2. ハロゲンランプの「原稿照射」機能を除電ランプにより代替  
この機能冗長系は実現しなかった。除電ランプから原稿への光路を作ることが困難であったという理由もあるが、主な理由はハロゲンランプと除電ランプの発光特性が異なるためであった。これらの部品の特性を可換にすれば、この冗長系は実現可能である。
3. 除電ランプの「ドラム除電」機能をメインチャージャにより代替  
この機能冗長系も実現しなかった。主な理由は、高電圧を印加するメインチャージャの極性を逆転させることが難しかったことと実現した場合に有効に作用するかという点が不明であったためである。
4. メインチャージャの「ドラム帯電」機能を転写チャージャにより代替  
この機能冗長系は実現した。部品の変更としては、メインチャージャの機能を代替する転写チャージャの特性変化の可操作領域を拡大するためにトランスを変更した。また、制御シーケンスプログラムは新たに作成した。

以上により作成した実験機の諸元を表 8.1 に、実験機の外観を図 8.12 に示す。ただし、知的判断部は第 7 章に述べた PC9801、および、Macintosh を利用した同様のシステムを用いた。表 8.1 から分かるように本機能冗長実験機ではセンサ、アクチュ

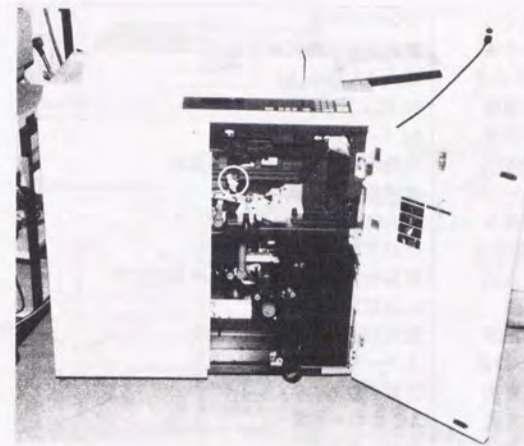


図 8.12: 機能冗長型自己修復実験機の外観

エータを計算機と接続することができなかった。これは、本実験機固有の限定である。

### 8.3.3 知識の記述

本実験システムで用いた知識は基本的に第 7.3.2 項で述べたものと同様である。機能冗長不使用時の対象モデルは第 7.3.2 項で述べた図 7.12、図 7.13 を基本とし、「除電」機能を付加した図 8.13、図 8.14 を用いる。さらにこのモデルに図 8.15、図 8.16 に示すような機能冗長モデルを付加した。このとき、図 8.15、図 8.16 における機能階層構造は図 8.14 と同様である。









図 8.14: 機能冗長実験機の対象モデル (機能レベル)

## 8.3.4 推論部と機械部の接続

ここでは、枠組として第7.3.3項で述べた制御型自己修復機械と同様のアプローチを取るが、実際問題としてセンサ、アクチュエータの接続はできなかった。そこで、センサ情報として、濃度計による画像濃度の測定値、用紙出力の目視、メインチャージャの断線信号、および、除電ランプ断線信号を用いた。ただし、推論の方法としては第7章と同様のモデルベース推論を用い、メインチャージャの断線を検知した場合にすぐに機能冗長系を利用するというはしない。これは将来的には多数のセンサ、アクチュエータ、および、機能冗長系を付加した場合も推論を実行可能とするためである。

また、本来ならば図8.8に示したように、機能冗長系を適用した後に制御型の修復を実行し、各パラメータのチューニングを行なうことにより修復を実現するが、本実験機に関しては機能冗長系適用後の転写チャージャの出力値等をあらかじめ調整、決定しておき、その値を適用することにした。

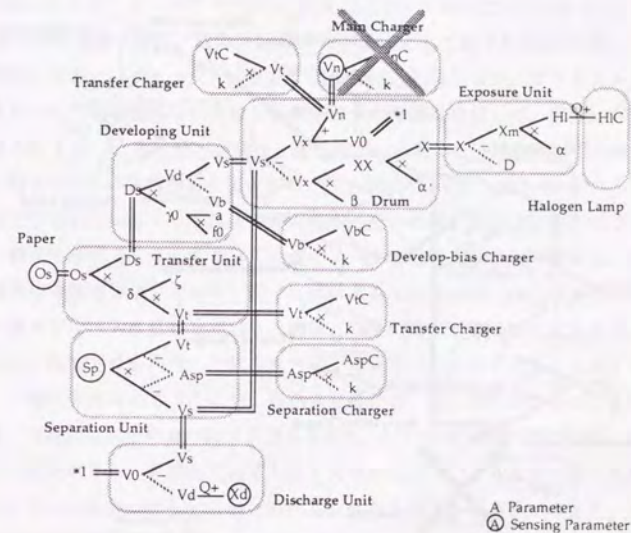


図 8.15: 機能冗長系候補 (A)







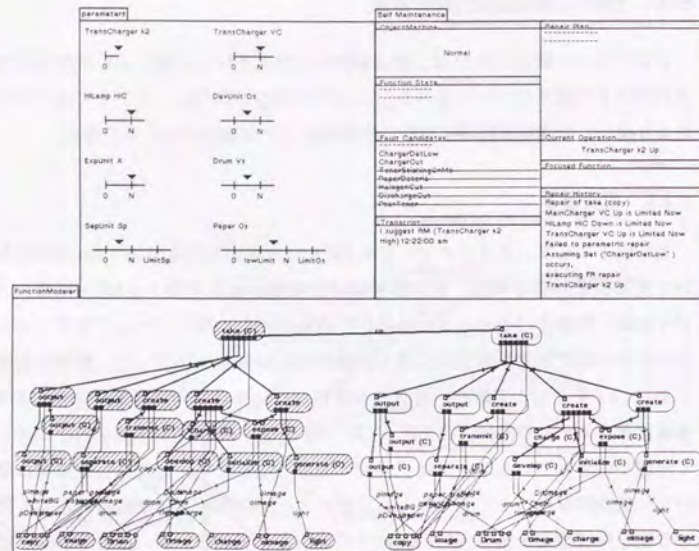


図 8.17: 機能冗長型推論システムの実行画面 (実行例 1)

をする転写前除電ランプ、ブランクランプが付加されており、既にある意味で機能冗長系が構成されていたため、一つの除電ランプを取り外しただけでは機能に大きな変化が生じないためである。転写前除電ランプやブランクランプを同時に取り外して複写動作を行なうことは、この機械特有の複雑な制御シーケンスに由来する問題のため不可能であった。ただし、正常の状態では機能冗長系 (B) を動作させた場合も正常の出力画像を得ることができた。しかし、故障に対処できないという意味では価値のない機能冗長系といえる。

このように、この例では本研究で付加した機能冗長系 (B) を有効に動作させることはできなかった。制御シーケンスをより単純なものに置き換えることにより、本機能冗長系の有効性を検証することは可能であると考えられる。

対象機械に既に存在していた機能冗長系として除電ランプ、転写前除電ランプ、および、ブランクランプの三者の関係は考察に値する。これら三者の機能はそれぞれ微



図 8.18: 機能冗長系 (A) を利用した出力画像

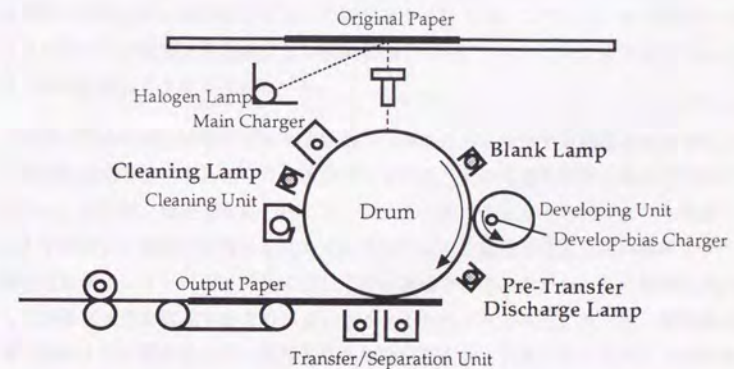


図 8.19: 機能冗長型自己修復実験機のランプ配置



妙に異なっている。例えば、ブランクランプは縮小複写時に余白を白くするためのものであり、転写前除電ランプは転写効率を上げるためのものである。この意味で、これら三者は単純な2 out-of 3 部品冗長ではない。しかし、結果として「除電」機能も果たしているために機能冗長性を保持しており、本実験により故障を引き起こすことができなかった。

## 8.5 考察

### 8.5.1 機能冗長系の評価

第8.4節で述べた実験の結果から、機能冗長系(A)は制御型自己修復機械が修復できない故障に対して機能保全を行なえたという意味で有効であることが確認できた。特にこの結果は必要機能は満たしているが速度の低下等が認められる典型的な「機能保全」の実現であり、この点で本研究の目的の一つであった、機能冗長による機能保全を実現できたといえる。一方、機能冗長系(B)は十分に正しい設計を行なわなかったために、有効に修復を行なうことができなかった。

また、本実験では同時に一つの機能冗長系の利用である点、および、制御型の修復を行なわないという点で機能冗長一般の評価を本実験のみから充分に行なうことはできない。本実験の意義は、機能冗長系による機能保全を用いた自己修復機械の実現可能性を検証できた点である。

この結果から、一般に「機能冗長」という考え方は有効であり、これにより自己修復機械の修復可能な範囲を広げることができるといえる。このとき、本実験では実現できなかったが機能冗長型修復と制御型修復を組み合わせることによりさらに広い範囲の修復を実現できると考えられる。

従来の部品冗長と比較すると本実験機での機能冗長は付加する部品数を少なくして冗長系を発揮可能であるという特徴は持つものの、その有効な故障の範囲は同等ではない。例えば、機能冗長系(A)とメインチャージャをもう一本持つという部品冗長系とを比較した場合、両者ともにメインチャージャを動作させるための高圧トランス等を含むメインチャージャ系統に対してのみ有効である。ただし、この対象範囲に関して部品冗長系を実現するためには、メインチャージャ、高圧トランス、導線等全てを二重化する必要があるが、機能冗長系では転写チャージャの出力トランスの仕様を変更するのみであった。しかし、制御シーケンスプログラムの変更が必要な範囲は機能冗長系を利用するほうが大きいと考えられる。しかし、現時点の機能冗長系が理想的なものであるとはいえず、その今後の方向性については第8.5.4項で述べることにする。



### 8.5.2 故障診断／修復計画システムと機能冗長

機能冗長型自己修復機械における知的判断部の特徴は、機能冗長系を実現するために対象モデルを変更する操作を実現することである。本研究の知的判断部は実行例1が示すように、この操作を実現することができた。また、第8.2節で述べたインタラクティブな修復計画、実行方法により、効率的に機能冗長型自己修復を実現できることを確認した。しかし、第6.3節で述べたように正常時と機能冗長発現時の対象モデルの共通部分に故障が発生している時はその冗長系は使用しないという限定をつけたが、モデルの変更方法をより高度化する必要がある。

### 8.5.3 機能冗長設計の方法

本研究で提案した機能冗長設計という設計思想は、ある意味で現在実際に行なわれている設計方針の逆を提案するものである。すなわち、現実に行なわれている設計においては、最小のコストで最大の性能を発揮するために軽量化、各部品の特長化、高度化を追求するという方針がとられる。例えば、第8.3.2項で述べたように複写機のリソゲンランプと除電ランプは出力光の特性が異なるため可換ではなかった。この二つの部品はランプという同一の性質を持つが、「露光」機能、「除電」機能という各部品の機能に対する特性を追求した結果、出力光の特性等が異なったものとなった。上記の「性能」には多くの場合、寿命、信頼性、故障の影響に対する安全性といった要素は含まれていないか、もしくは、含まれていたとしても最低限度たさなければならない条件という負の意味合いが強い。しかし、社会的状況から今後の機械の設計を考える場合は、この信頼性、アベイラビリティ、故障の影響等の要素がより重要な正の性能となると考えられる。このとき、本研究で提案した機能冗長設計のように、各部品に潜在機能を与えるために少しずつ直接にはいわゆる性能に貢献しない「無駄」を付加することにより、高信頼、かつ、故障に強い機械を設計するという方法が重要になってくると考えられる。

機械部の設計は機能冗長系(B)が示すように部品冗長系の設計と比較すると複雑になっている。この問題に対して、FR-Reasonerを用いることにより機能冗長設計を有効に支援することができると考えられる。その理由として、以下のFR-Reasonerの特徴が挙げられる。

1. 機能冗長設計にとって最も重要な機能と F-B 関係を FBS Modeler により直接

的、明示的に扱うことが可能であること。

2. 機械の物理的動作を表す挙動を明示的、直接的に表現、シミュレーション可能であること。
3. 機能冗長候補を導出できること。

これらにより、機能や機能冗長系に関する判断を機能レベルで主観的に行ないつつ、その物理的实现性、無矛盾性を挙動、状態レベルで判定することが可能になった。

しかし、第8.1.2項で述べた設計対象の構造決定、パラメータの決定、および、制御シーケンスの決定という機能冗長設計の三段階のうち、本研究では第一段階の設計対象の構造決定の部分のみを支援したが、残りの二段階の部分の支援を行っていないという問題が残されている。この問題のために実験機の機能冗長系(B)は有効に動作しなかった。この問題は、設計一般に関連する、概念設計から詳細設計までの支援という大きな問題であり、今後の研究課題である。ただし、本研究で扱った基本構造を決定する段階の情報から要求する動作モードの条件を抽出し、パラメータに関する情報を付加することにより制御シーケンスを生成させることは可能である。小池[Koike92b]は、本研究で用いた対象表現から以上の方法により制御シーケンスを導出する方法を提案している。

### 8.5.4 機能冗長型自己修復機械の拡張

本実験機は既存の機械を改良したこと、および、制御型の修復を実現できなかった点で、機能冗長系が有効に動作することは検証できたが、機能冗長機械の可能性を全て検証するには至らなかった。そこで、今後の機能冗長系の実現、利用方法について考察してみると、三段階の発展形態を考えることができる。

第一の段階は、機能冗長型修復と制御型修復を両立させ、かつ、一つの対象機械に多数の機能冗長系を付加し各系を使い分けることである。これは、本研究で提案した範囲の実現であり、これによっても故障のある程度広い範囲に対応する自己修復機械を実現可能であると考えられる。このとき、第8.3節に触れたような部品の共通化による部品交換を含めることも有効である。

第二の段階は、複数の機能冗長系の組合せ利用である。これにより複雑な故障に対応することが可能になり修復実現の範囲を広くすることができる。複数の機能冗長系の利用を実現するためには、対象モデルの変更方法と制御シーケンスの作成方法とい



う二点が問題になる。

これらにより、機能冗長系の特徴である部品冗長よりも柔軟な故障への対応を実現可能になる。例えば、メインチャージャとハロゲンランプが両方とも故障した場合、除電ランプとハロゲンランプの交換により、ハロゲンランプの機能を実現し、かつ、ドラムを三回転させることにより、ハロゲンランプによる除電、および、転写チャージャを用いて主帯電、転写の三機能を発現させることも可能である。このように、機能冗長機械は少ない部品数の増加で、多数の機能冗長系を構成することにより、多くの故障に対応可能である。

第三の段階は、故障発生時における自己修復機械による機能冗長系の動的な発見、実行を実現することである。このとき、同種の部品の共通化は機能冗長系の実現可能性を高めるという意味で重要である。このような機械自身による機能冗長系の動的生成を実現するとき、FBSダイアグラムの枠組、特に機能とその実現挙動を記述する枠組は機械のある環境下における目的や実現方法を記述するために重要な役割を果たすと考えられる。この問題は、ある機械系の範囲内に対象領域を限定した形の自動設計、すなわち、要求機能から実現挙動、状態を導出する問題と考えられる。この対象領域を限定したことによって、一種の自動設計問題を効率的に解く方法を提案することは今後の課題である。また、機能冗長系を発見できた場合にそれを自己修復機械が実行できるようにしなければならない。これは、一つには制御シーケンスを動的に生成することによって実現可能であり、小池の方法は動的な制御シーケンスの生成の可能性を提示している。さらにより柔軟な形としては、多数の同様の部品から構成されるような細胞型の機械が考えられる [Koike92b]。

これらの発展した形の機能冗長型自己修復機械を実現することにより、本研究で提案した「機能保全」、「機能冗長」という考え方の重要性がより高まることが期待される。

## 8.6 第8章のまとめ

本章は機能冗長型自己修復機械の設計方法、および、実験機を用いて機能冗長の有効性に関して議論した。第8.1節では、第4章で述べたFBSモデラーを用いて機能冗長機械の概念設計を支援する方法を述べた。第8.2節では、制御シーケンス・プログラムを入れ換えることにより機能冗長型修復を実現する方法を述べた。第8.3節では、本研究で構築した機能冗長型自己修復機械の設計について述べた。概念設計では様々な機能冗長系を導出可能であったが、ここでは、既存の機械の改造という方法を用いたので、概念設計された冗長系の実際の機械への適用、および、制御シーケンス・プログラムの作成に難しさがあることが明らかになった。さらに、第8.4節で本実験機を用いた修復実験の結果について述べ、第8.5節で実験結果、機能冗長設計方法、および、機能冗長機械の有効性について考察を行なった。本実験により制御型修復では行なえないような修復が実現できたなど、機能冗長系の基本的な有効性と知的判断部が有効に動作することが確認できた。さらに、より多数の機能冗長系の付加、複数の機能冗長系の利用、および、自己修復機械自身による動的な機能冗長系の発見、利用等の今後の発展により「機能保全」、「機能冗長」という本研究の考え方がより有効になると考えられる。







本章では、結論と本研究に残された課題を述べる。また最後に、今後の自己修復機械について展望する。

## 9.1 結論

保全作業の社会的重要性が高まってきており、また、特に保全作業の自動化の必要性が高まっていることを背景とし、これらの問題を解決するために自己修復機械の構築方法を明らかにすることが本研究の目的であった。

本研究ではこの目的を実現するために、自己修復機械の実現方法の提案、知的判断部の構築方法の提案、および、自己修復機械に適した機械部の設計方法の提案という三側面から議論を行なった。

自己修復機械の実現方法として、本研究では「機能保全」により機械の一部が故障しても動作する機械、すなわち、「柔らかく壊れる」機械という考え方を提案した。また、修復実行の方法として、アクチュエータの制御により修復を実現する制御型自己修復、および、既存の部品の潜在機能を利用し修復を実現する機能冗長型自己修復の二つの方法を提案した。

知的判断部の構築方法を提案するために、機械のモデリング、故障診断の自動化、および、修復計画の自動化という三点から議論を行なった。

- 機械のモデリングの方法として、人間の機械に対する意図や理解を表す機能と機械の物理的振舞い、性質を表す挙動、状態を関係付けて表現するFBSダイアグラムを提案した。このFBSダイアグラムに基づき機械を表現することで、設計や保全作業を有効に行なえることを明らかにした。
- 故障診断の自動化手法として、対象モデルに基づき定性物理を用いた手法を提案した。具体的には、構造変更を引き起こす故障を扱うことが可能で、かつ、因果的に関連する複数の故障発生の状況を診断可能な単一故障発生過程仮説に基づく故障診断手法を提案した。また、定性物理を現実の対象に適用する際の問題点を解決するためにファジィ定性物理とファジィ定性物理に基づく故障診断手法を提案した。実験的な故障診断システムを構築することにより、これらの手法がその目的を達成し、有効に動作することを確認した。
- 修復計画の自動化手法として、故障診断と同様に対象モデルに基づく定性物理を用いた制御型修復計画手法、および、機能冗長型修復計画手法を提案した。これら手法により、機能保全を目的として故障に対応して柔軟に修復計画を作成することが可能になった。



自己修復に適した機械部の設計方法として、制御型、および、機能冗長型修復を実現可能な機械の設計方法を提案した。制御型の設計に関しては、制御型修復実行機能、状態監視機能の実現のために、機械の因果的構造、アクチュエータ、センサの配置について議論した。また、機能冗長型機械部の設計方法に関しては、FBS モデラーを用いた機能冗長機械の概念設計支援方法を提案した。

以上に提案した手法の有効性を検証するために、本研究では制御型自己修復実験機、および、機能冗長型自己修復実験機を構築し、実際に実験を行なった。その結果、故障に応じた柔軟な修復を実行する自己修復機械の実現が可能であること、知的判断部は保全の特徴である不定性、非繰り返し性に対応し、有効に故障診断、修復計画を行なえること、および、機能冗長系が故障に対して有効に作用することを確認した。

以上の議論により、本研究は「自己修復機械」の実現という課題に対して、「機能保全」という考え方の実現方法、設計方法、および、運用方法を提案することで、自己修復機械の一設計方法論を提案することができた。

## 9.2 本研究の課題

本研究の今後の課題として以下の五点を挙げることができる。

保全という観点からの設計を考えた場合、設計時に故障や劣化情報を積極的に利用することが必要である。本研究ではこの点に関して十分に考察を行なうことができなかったが、本研究で提案した、機械のモデル化手法、故障発生過程のモデル、および、劣化現象のモデルは、故障を予測した設計のための知識記述の枠組として有効であると考えられる。

本研究の「機能保全」をより有効に作用させるためには、使用者による使用状況に応じた機能のトレードオフを行ない、修復の目的を動的に決定することが重要になってくる。この問題は機械の価値という問題を含んでおり自動化には適さないので、使用者に対し判断が容易な形での意思決定支援を行なう必要がある。

本研究の知的判断手法では、保全の特徴である多様性に対応する明確な方法を提示できなかった。この問題に対応するためには、大規模な知識ベースを構築、利用することが必要になる。本研究で述べた FBS モデラーや故障診断システムを有効に動作させる場合も知識の量が問題になる。逆に、多くの知識を収集することができれば、本研究で提案した FBS ダイアグラムや故障現象の枠組の有効性を検証することができる。このように、知識獲得の問題は本研究の今後の発展のために重要な課題となる。

本研究の自己修復機械の知的判断部は、劣化のモデルでいうところのショートタームの視点のみを扱った。しかし、修復計画の決定や機能トレードオフの判断材料として、劣化、寿命というロングタームの視点は重要である。このようなロングタームとショートタームの両方の視点を持つ、より総合的な知的判断部の構築が必要である。

本研究では、実験機により本手法による自己修復機械の実現性の検証を行なうことはできたが、実用化へ向けての課題は数多く残されている。上記の知識獲得の問題もその課題の一つであるが、その他、大規模問題に対応するための推論の効率化の問題、設計時における機能冗長系の評価の問題、計算法部、センサ部の故障に対する対処法、故障に対処可能な範囲を拡張するための方法等の問題が挙げられる。



### 9.3 今後の自己修復機械

本研究では、自己修復機械を実現する一手法を提案することができた。本手法に基づいて、今後実現可能であると考えられる自己修復機械の形態について述べることにする。

保全ロボット [Asama89] は、本研究における修復実行自動化手法として有効に自己修復機械に組み込むことができる。一つの考え方は、保全ロボットと予備部品の集まりを用意することで、修復実行方法としての自動部品交換を実現することが可能になり、修復可能範囲を広げることができる。また別の見方として、保全ロボットを万能な潜在機能を持つ部品として扱うことにより機能冗長系を構成することも考えられる。例えば、折れた支持部品の代替として保全ロボットにより当面の間必要箇所を支えることが考えられる。いずれの方法においても、保全ロボットを利用して、何を目的としてどのような方針により修復を行なうかという意味決定部分が重要であり、この意味で本研究の知的判断部の手法は有効であると考えられる。

より柔軟な構造変更による機能保全を考えると、機械の形態は数種類の部品が多数集まって一つの機械を構成するような細胞型の機械形態が有効であると考えられる [Koike91, Koike92a]。このような機械形態において、故障に応じた構造の再構成を行えば、非常に柔軟な修復が実現可能である。この方法は、本研究の機能冗長系の動的発見、制御シーケンスの動的変更という考え方の延長線上にあるものと捉えることができる。この方向性は、自律分散機械 (例えば、[Matsumoto89]) の研究と深く関わりあう。この場合においても、機械の目的、および、その物理的実現方法に関する知識表現形態としての FBS ダイアグラム、および、保全に関する知的判断のモデルとしての故障診断、修復計画の手法は有効である。

さらに一般に、周囲の状況、内部状況、使用者の要求等の変化にあわせて、適宜自己の状態を変化させつつ対応する機械システムを「知的機械システム」と呼ぶとすると、機械システムの今後の方向の一つとして知的機械システムという形態がますます重要になってくると考えられる。本研究は、故障という内部状況の変化を対象として知的機械システムの構築方法論を提案したものであると位置付けられ、この意味で、本研究は知的機械システム実現のための一方法論を提案することができたといえる。今後は、内部状況の変化のみならず、周囲の状況、使用者の要求等の変化にも柔軟に対応可能な機械システムの構築手法を考えていかなければならない。

### 謝辞

指導教官である吉川弘之教授には、修士課程、博士課程の五年間にわたり研究を指導していただきました。吉川教授には、「自己修復機械」という実現不可能とも思える遠大なテーマを与えていただきました。本研究は、吉川教授の理論的方向付けがあって初めて可能になったものです。また、教授の常に新しい視点からコメント、社会との関わりからのコメントは本研究を進める上でその目標や自分のやりたいことが何であるのかということをはっきりと明かにして下さいました。さらに、研究上の面のみならず、その優しい人柄、深い才能は研究者のあるべき姿、自分の進むべき目標となりました。深く感謝致します。

富山哲男助教授からは、研究の具体的な方向付け、目標を実現するためのアイディア等に関して親身にかつ厳しく議論して頂きました。富山助教授の強い信念と正しい判断に基づくアドバイスは本研究が袋小路に入った時に、常にブレイクスルーとなりました。また、富山助教授と常日頃接することにより、国内外の研究動向やその中で研究計画の立て方、研究者のスタンス等、研究者になるための方法論を身を持って教えて頂きました。深く感謝致します。

本論文の査読をして頂き、有益なコメントをしていただいた新井民夫教授 (東京大学工学部精密機械工学科)、木村文彦教授 (同)、矢川元基教授 (東京大学工学部原子力工学科) に感謝いたします。

三田工業株式会社の皆様には 1988 年に行なった故障診断実習以来、実験機の製作等多岐にわたり協力して頂きました。特に、無謀とも思える研究の主旨を理解して頂き、惜しみない援助をして頂いた三田順啓社長、高岸暎治氏には深く感謝致します。また、下村芳樹氏には、氏が吉川研究室研究員当時から自己修復機械の研究の立ち上げの際御指導いただきました。また、共同研究の担当者として我々の無理な要求を実現して頂き、先輩として様々な相談に乗って頂きました。氏なしでは本研究は中途



半端なものにならざるをえなかったと思われます。深く感謝致します。久島昌彦氏、加藤一郎氏、楠本弘氏、下村芳樹氏、石井喜文氏には、第7章に述べた制御型自己修復機械を製作して頂きました。AIシステム研究室のメンバーの久島昌彦氏、樫田正氏、宮川修宏氏、平田信行氏、森本一弘氏、梅澤秀夫氏、下村芳樹氏、竹本光利氏、堀和人氏、谷川貞夫氏、小川和博氏には、第8章に述べた機能冗長型自己修復機械を製作して頂き、また、常に具体的な経験、データに基づいた様々なヒントを頂きました。三田工業の諸氏の深い経験と実践技術に基づく様々な議論は、本研究を進める上での良き反省材料となりました。ここに、深く感謝致します。

吉川富山研究室の先輩、同輩、後輩の方々には、研究活動を始めとし公私共にお世話になりました。特に、保全グループとして研究と一緒に進めてきた、佐藤隆氏、小池雄一氏、井上真治氏、帖佐太一氏、黒木文明氏、茂木康男氏、和田吉樹氏、大道憲哉氏には深く感謝致します。特に、佐藤氏は第5章で述べた故障シミュレーション、第6章の修復計画の自動化に関して多大な貢献がありました。また、小池雄一氏は、第5章の単一故障発生過程仮説の理論の提案、および、故障診断 / 修復計画システムのインプリメンテーションに関して多大な貢献がありました。茂木康男氏は第5章で述べたファジィ定性物理とその故障診断手法に関して貢献がありました。保全グループの全てのメンバーに深く感謝致します。

第2章で述べた、佐藤による修復作業の実際に関する調査は、新日鉄株式会社エレクトロニクス研究所の田浦俊春氏の御厚意により、新日鉄の製鉄所を見学、調査することにより行なうことができました。深く感謝致します。

第4章に述べたFBSダイアグラムに関する議論は本研究室のSIG-Functionにおける自由な討論から生まれました。SIG-Functionのメンバーである安藤公一氏、田浦俊春氏、桐山孝司氏、武田英明氏、浜田進氏、山本文緒氏に感謝致します。

第4章に述べたFBSモデラーのインプリメンテーションにおいて、桐山孝司氏の御厚意により、氏の開発したメタモデルシステムを利用させて頂きました。感謝致します。

理化学研究所の浅間一氏には、「保全」に関する研究の面白さを教えて頂き、また、適切な批評を頂きました。感謝致します。

また、車谷浩一氏、石田慶樹氏、桐山孝司氏、武田英明氏との議論を通じて、様々なアイデアを得ることができました。感謝致します。

研究室での活動を支えてくれた、技官の小野里雅美（旧姓熊沢）さん、碓山みち子

さん、秘書の下村貴子（旧姓奈良）さん、荒武由香さん、岡庭美樹さん、高井正子さん、塩田恭子さん、野萱和子さん、新井富美子さんに感謝します。

最後に、筆者は日本学術振興会特別研究員として日本学術振興会からの援助を受け、また、本研究の一部は文部省科学研究費補助金（特別研究員奨励費）の援助を受けたことをここに明記します。



## 参考文献

- [Abu-Hanna88] A. Abu-Hanna and Y. Gold: An Integrated, Deep-shallow Expert System for Multi-level Diagnosis of Dynamic Systems, In J. S. Gero(ed.): *Artificial Intelligence in Engineering: Diagnosis and Learning*, pp. 75-94, Elsevier, Berlin (1988).
- [Akizuki91] 秋月影雄: 保全のための基礎的測定技術, 精密工学会誌, Vol. 57, No. 3, pp.422-425 (1991).
- [Amari89] 甘利俊一: ニューラルネットワーク研究の過去、現在、将来, 人工知能学会誌, Vol. 4, No. 2, pp.120-127 (1989).
- [Anonymous82] 機械システム振興協会(編): 機械工業における信頼性と保全に関する調査研究報告書, 国際科学振興財団 (1982).
- [Anonymous86] 発電設備技術検査協会(編): 軽水炉改良技術確証試験等に関する調査報告書(原子力発電所信頼性向上調査 資料編-II [原子力保全のトータルシステムに関する調査]), 発電設備技術検査協会 (1986).
- [Anonymous90] 日本ロボット学会(編): ロボット工学ハンドブック, コロナ社 (1990).
- [Arai87] 新井政彦, 本位田真一: 診断型エキスパート・システム, 情報処理, Vol. 28, No. 2, pp.177-186 (1987).
- [Asama89] 浅間一: 保全用マニピュレータ構成論, Ph.D. thesis, 東京大学大学院工学系研究科精密機械工学専攻 (1989).
- [Bobrow84] D. G. Bobrow and P. J. Hayes(eds.): *Artificial Intelligence*, volume 24 (1984).



- [Bradshaw91] J. A. Bradshaw and R. M. Young: Evaluating Design Using Knowledge of Purpose and Knowledge of Structure, *IEEE Expert*, Vol. 6, No. 2, pp.33-40 (1991).
- [Bylander85] T. Bylander and B. Chandrasekaran: Understanding Behavior Using Consolidation, In *Proceedings of IJCAI-85*, pp. 450-454 (1985).
- [Crow91] J. Crow and J. Rushby: Model-based Reconfiguration: Toward an Integration with Diagnosis, In *Proceedings of AAAI-91*, pp. 836-841 (1991).
- [Dague87] P. Dague, O. Raiman, and P. Deves: Troubleshooting: When Modeling Is the Trouble, In *Proceedings of AAAI-87*, pp. 600-605 (1987).
- [de Kleer84a] J. de Kleer: How Circuits Work, *Artificial Intelligence*, Vol. 24, pp.205-280 (1984).
- [de Kleer84b] J. de Kleer and J. Brown: A Qualitative Physics Based on Confluences, *Artificial Intelligence*, Vol. 24, pp.7-83 (1984).
- [de Kleer86] J. de Kleer: An Assumption-based TMS, *Artificial Intelligence*, Vol. 28, No. 2, pp.127-162 (1986).
- [de Kleer87] J. de Kleer and B. C. Williams: Diagnosing Multiple Faults, *Artificial Intelligence*, Vol. 32, No. 1, pp.97-130 (1987).
- [de Kleer89] J. de Kleer and B. C. Williams: Diagnosis with Behavioral Modes, In *Proceedings of IJCAI-89*, pp. 1324-1330 (1989).
- [Faltings87] B. Faltings: Qualitative Kinematics in Mechanisms, In *Proceedings of IJCAI-87*, pp. 436-442 (1987).
- [Forbus84] K. Forbus: Qualitative Process Theory, *Artificial Intelligence*, Vol. 24, No. 3, pp.85-168 (1984).
- [Forbus87] K. Forbus, P. Nielsen, and B. Faltings: Qualitative Kinematics: A Framework, In *Proceedings of IJCAI-87*, pp. 430-435 (1987).
- [Forbus89a] K. Forbus: Introducing Actions into Qualitative Simulation, In *Proceedings of IJCAI-89*, pp. 1273-1278 (1989).

- [Forbus89b] K. D. Forbus: A Perspective on Qualitative Physics, *人工知能学会誌*, Vol. 4, No. 5, pp.497-508 (1989).
- [Franke91] D. W. Franke: Deriving and Using Descriptions of Purpose, *IEEE Expert*, Vol. 6, No. 2, pp.41-47 (1991).
- [Fukuda87] 福田敏男: セル構造を有する自己組織化ロボット、-その基礎概念とセル間の粗接近制御および形態決定方法について、*電気学会論文誌*, Vol. 107-C, No. 11, pp.1377 (1987).
- [Genesereth84] M. Genesereth: The Use of Design Descriptions in Automated Diagnosis, *Artificial Intelligence*, Vol. 24, No. 1-3, pp.411-436 (1984).
- [Gero89] J. S. Gero and M. A. Rosenman: A Conceptual Framework for Knowledge-Based Design Research at Sydney University's Design Computing Unit, In J. S. Gero(ed.): *Artificial Intelligence in Design*, pp. 363-382, Springer-Verlag, Berlin (1989).
- [Gershenson91] J. Gershenson and K. Ishii: Life-cycle Serviceability Design, In L. A. Stauffer(ed.): *Proceedings of Design Theory and Methodology - DTM'91*, pp. 127-134, ASME (1991).
- [Goldberg87] A. Goldberg and D. Robson: *Smalltalk-80: The Language*, Addison-Wesley, Reading, MA (1987)  
相磯秀夫監訳, Smalltalk-80 - 言語詳解 -, オーム社, (1987).
- [Hamilton88] T. P. Hamilton: HELIX: A Helicopter Diagnostic System Based on Qualitative Physics, *Artificial Intelligence in Engineering*, Vol. 3, No. 3, pp.141-150 (1988).
- [Hayes85a] P. Hayes: Naive Physics Manifesto I: Ontology for Liquids, In J. Hobbs and R. Moore(eds.): *Formal Theories of the Commonsense World*, pp. 71-107, Ablex Publishing (1985).
- [Hayes85b] P. Hayes: The Second Naive Physics Manifesto, In J. Hobbs and R. Moore(eds.): *Formal Theories of the Commonsense World*, pp. 1-36, Ablex Publishing (1985).
- [Honiden91] 本位田真一: 協調型アーキテクチャによるソフトウェアの自動生成, *人工知能学会誌*, Vol. 6, No. 2, pp.184-186 (1991).



- [Ichida68] 市田嵩: 保全性工学入門, 日科技連 (1968).
- [Iri80] M. Iri, K. Aoki, E. Ooshima, and H. Matsuyama: A Graphical Approach to the Problem of Locating the Origin of the System Failure, *Journal of the Operations Research Society of Japan*, Vol. 23, No. 4, pp.295-311 (1980).
- [Ito90a] 伊藤公俊: 知的 CAD のための属性モデリング方法論, 人工知能学会研究会資料 SIG-KBS-9002, pp. 97-105 (1990).
- [Ito90b] 伊藤修, 松林成彰: 定性推論に基づく学習制御 (第3報) - 定性モデルの自己修正 -, 1990 年度人工知能学会全国大会 (第4回) 論文集, pp. 227-230 (1990).
- [JIS87] 日本規格協会 (編): JIS 工業用語大辞典 (第2版), 日本規格協会 (1987).
- [Joskowicz88] L. Joskowicz and S. Addanki: From Kinematics to Shape: An Approach to Innovative Design, In *Proceedings of AAAI-88*, pp. 347-352 (1988).
- [Joskowicz89] L. Joskowicz: Simplification and Abstraction of Kinematic Behaviors, In *Proceedings of IJCAI-89*, pp. 1337-1342 (1989).
- [Keuneke91] A. M. Keuneke: Device Representation The Significance of Functional Knowledge, *IEEE Expert*, Vol. 6, No. 2, pp.22-25 (1991).
- [Kiriya91] 桐山孝司: 定性物理を用いた設計対象表現論, Ph.D. thesis, 東京大学大学院工学系研究科 (1991).
- [Kise91] 吉瀬隆, 村田真人: モデルに基づく故障診断システム, 情報処理学会人工知能研究会資料 78-3, pp. 1-10 (1991).
- [Kobayashi90] 小林康弘, 宇多村元昭, 佐藤隆雄, 片柳大二郎, 今野隆博, 吉川和明: プラント機器の予防保全計画支援システム, 人工知能学会誌, Vol. 5, No. 5, pp.558-568 (1990).
- [Koike91] 小池雄一, 梅田靖, 富山哲男, 吉川弘之: 細胞型機械の構築 (第一報) - 概念と基本構成 -, 1991 年度精密工学会春季大会学術講演会講演論文集, pp. 553-554 (1991).

- [Koike92a] 小池雄一, 梅田靖, 下村芳樹, 富山哲男, 吉川弘之: 自己修復機械の構築 (第4報) - 知識の分散化による論理構造の再構成 -, 1992 年度精密工学会春季大会学術講演会講演論文集 (1992) (印刷中).
- [Koike92b] 小池雄一: 分散型知識処理を用いた自己修復機械の研究, 東京大学工学部精密機械工学科修士論文 (1992) (印刷中).
- [Kuipers86] B. Kuipers: Qualitative Simulation, *Artificial Intelligence*, Vol. 29, No. 3, pp.289-338 (1986).
- [Kuipers88] B. Kuipers and D. Berleant: Using Incomplete Quantitative Knowledge in Qualitative Reasoning, In *Proceedings of AAAI-88*, pp. 324-329 (1988).
- [Kumano90] 熊野彰司, 坪内元孝, 合田忠弘, 竹垣盛一, 京本寿美恵: 変電所運転支援エキスパートシステムの開発, 人工知能学会研究会資料 SIG-KBS-9003, pp. 81-90 (1990).
- [Kurumatani89] 車谷浩一: 概念設計支援のための機械の挙動の表現と推論, Ph.D. thesis, 東京大学大学院工学系研究科 (1989).
- [Mäntylä88] M. Mäntylä: *An Introduction to Solid Modeling*, volume 13 of *Principles of Computer Science Series*, Computer Science Press, Rockville, Maryland (1988).
- [Matsumoto83] 松本啓之亮: 知識ベースに基づく電力系統復旧方式の決定法, 電気学会論文誌, Vol. 103-B, No. 3, pp.25-32 (1983).
- [Matsumoto89] 松元明弘, 浅間一, 石田慶樹, 田中澄人: 自律分散型ロボットシステム ACTRESS の開発研究 (第1報) - システムの構想 -, 1989 年度精密工学会春季大会学術講演会講演論文集, pp. 115-116 (1989).
- [Matsumoto91] 松本幹雄, 川合稔, 中田秀基, 山本文緒, 富山哲男, 吉川弘之: 設計向き大規模知識ベースの研究, 1991 年度人工知能学会全国大会 (第5回) 論文集, pp. 717-720 (1991).
- [Mavrovouniotis87] M. Mavrovouniotis and G. Stephanopoulos: Reasoning with Orders of Magnitude and Approximate Relations, In *Proceedings of AAAI-87*, pp. 626-630 (1987).



- [Miles72] L. D. Miles: *Techniques of Value Analysis and Engineering*, McGraw-Hill, New York (1972).
- [Mizukami90] 水上雄一, 片桐充, 長田政和, 尾上晴紀, 山田邦臣, 高岡良行, 山下昌利: 一次変電所運転支援システム, 人工知能学会研究会資料 SIG-KBS-9003, pp. 91-100 (1990).
- [Mizumoto88] 水本雅晴: ファジィ理論とその応用, サイエンス社 (1988).
- [Nayak91] P. P. Nayak, L. Joskowicz, and S. Addanki: Automated Model Selection Using Context-dependent Behaviors, In B. Kuipers(ed.): *Proceedings of 5th International Workshop on Qualitative Reasoning about Physical Systems*, pp. 10-24 (1991).
- [Nishida87] 西田豊明, 堂下修司: 簡単なバルス回路における不連続変化の定性的解析法, 人工知能学会誌, Vol. 2, No. 4, pp.501-510 (1987).
- [Nishida88] 西田豊明: 定性推論に関する最近の研究動向 (I) 基礎技術の進歩, 情報処理, Vol. 29, No. 9, pp.1009-1022 (1988).
- [Nishida91] 西田豊明: 定性推論の考え方とその知的問題解決への応用, 情報処理, Vol. 32, No. 2, pp.105-117 (1991).
- [Nishizawa90] 西沢靖雄, 長岡幸夫: プラント保守作業計画支援システム, 人工知能学会研究会資料 SIG-KBS-9003, pp. 139-144 (1990).
- [Nonaka90] 野中保雄, 島岡淳: 冗長系 理論と実際, 日科技連 (1990).
- [Ohsuga91] 大須賀節雄, 李春野: ソフトウェア設計自動化に向けて, 人工知能学会誌, Vol. 6, No. 2, pp.159-166 (1991).
- [Oomichi92] 大道憲哉: 大規模問題を対象とした故障診断システムの構築, 平成三年度東京大学工学部精密機械工学科卒業論文 (1992).
- [Ooshima91] 大島栄次: 設備管理と異常診断, 精密工学会誌, Vol. 57, No. 3, pp.413-417 (1991).
- [Price88] C. Price and M. Lee: Applications of Deep Knowledge, *Artificial Intelligence in Engineering*, Vol. 3, No. 1, pp.12-17 (1988).
- [Raiman86] O. Raiman: Order of Magnitude Reasoning, In *Proceedings of AAAI-86*, pp. 100-104 (1986).
- [Reiter87] R. Reiter: A Theory of Diagnosis from First Principles, *Artificial Intelligence*, Vol. 32, No. 1, pp.57-95 (1987).

- [Rodenacker71] W. Rodenacker: *Methodisches Konstruieren*, Springer, Berlin (1971).
- [Roth82] K. Roth: *Konstruieren mit Konstruktions Katalogen*, Springer-Verlag, Berlin (1982).
- [Roversi91] A. Roversi and M. Conforti: Maintenance in Europe, 精密工学会誌, Vol. 57, No. 3, pp.408-412 (1991).
- [Saeki91] 佐伯元司: これまでの反省と今後の課題, 人工知能学会誌, Vol. 6, No. 2, pp.167-169 (1991).
- [Sato90] 佐藤隆: 対象モデル情報を用いた修復計画, 東京大学工学部精密機械工学科修士論文 (1990).
- [Sato91] 里佳史, 富山哲男, 吉川弘之: 多重オントロジーを用いた対象表現 (第一報), 1991 年度精密工学会秋季大会学術講演会講演論文集, pp. 483-484 (1991).
- [Sayama81] 佐山隼敏, 井上絃一: フォールト・ツリー解析 - その基礎と応用, 計測と制御, Vol. 20, No. 2, pp.32-41 (1981).
- [Shimomura90] 下村芳樹, 梅田靖, 富山哲男, 吉川弘之: 事例推論の修復戦略への適用, 1990 年度精密工学会春季学術講演会講演論文集, pp. 1231-1232 (1990).
- [Shimomura92] 下村芳樹, 谷川貞夫, 梅田靖, 富山哲男, 吉川弘之: 自己修復機械の構築 (第2報) - 仮想事例を用いた実用化へのアプローチ -, 1992 年度精密工学会春季学術講演会講演論文集 (1992) (印刷中).
- [Shinohara88] 篠原専二: 電車応急処置支援エキスパートシステム, 昭和 63 年度人工知能学会全国大会 (第二回) 論文集, pp. 281-284 (1988).
- [Shiomi68] 塩見弘: 信頼性入門, 日科技連 (1968).
- [Shiomi70] 塩見弘: 故障物理入門, 日科技連 (1970).
- [Shiomi91] 塩見弘: 故障メカニズムと解析法, 精密工学会誌, Vol. 57, No. 3, pp.418-421 (1991).
- [Struss89] P. Struss and O. Dressler: "Physical Negation" - Integrating Fault Models into the General Diagnostic Engine, In *Proceedings of IJCAI-89*, pp. 1320-1323 (1989).



- [Takada77] 高田祥三: 機械の故障診断に関する研究, Ph.D. thesis, 東京大学大学院工学系研究科 (1977).
- [Takatsuto90] 高津戸祐司, 八木昭, 照沼誠一, 吉田恵: 「常陽」異常時運転支援システムの開発, 人工知能学会研究会資料 SIG-KBS-9003, pp. 129-138 (1990).
- [Takemori89] 竹森利和, L.Wong, M. W.Siegel: 適応型知的匂いセンサー, 1989年度人工知能学会全国大会 (第3回) 論文集, pp. 201-204 (1989).
- [Tsuge84] 柘植義文, 潮崎淳一, 松山久義, 大島栄次, 井口友二, 淵上正勝, 松下元彦: 化学プラント用異常診断システムの実用化実験, 化学工学論文集, Vol. 10, No. 2, pp.240-246 (1984).
- [Ulrich88] K. T. Ulrich and W. P. Seering: Function Sharing in Mechanical Design, In *Proceedings of AAAI-88*, pp. 342-346 (1988).
- [Umeda89a] Y. Umeda, T. Tomiyama, and H. Yoshikawa: Model Based Diagnosis Using Qualitative Reasoning, In F. Kimura and A. Rolstadås(eds.): *Computer Applications in Production and Engineering CAPE'89*, pp. 443-450, Elsevier, Amsterdam (1989).
- [Umeda89b] 梅田靖: 対象モデルに基づく定性物理を用いた故障診断, 東京大学工学部精密機械工学科修士論文 (1989).
- [Umeda90] 梅田靖: 幾何推論に関する文献調査, 東京大学大学院工学系研究科精密機械工学特別研究IV 報告書 (1990).
- [Wada91a] 和田吉樹, 小池雄一, 梅田靖, 富山哲男, 吉川弘之: 統合化保全システムの研究, 1991年度精密工学会春季大会学術講演会講演論文集, pp. 937-938 (1991).
- [Wada91b] 和田吉樹: 統合化保全システムの研究, 平成二年度東京大学工学部精密機械工学科卒業論文 (1991).
- [Washio87] 鷺尾隆, 北村正晴, 相山一典: 定性推論のファジィ化の試み, 第5回知識工学シンポジウム講演資料, pp. 147-152 (1987).
- [Watanabe86] 渡辺栄一(編): フォールト・トレラント・システム, マグロウヒル (1986).

- [Watanabe89] 渡辺兼秀, 玉山清志, 橋高宏政: ニューラルネットワークを用いた時系列信号のパターン識別方法の開発, 1989年度人工知能学会全国大会 (第3回) 論文集, pp. 197-200 (1989).
- [Weld89] D. Weld and J. de Kleer(eds.): *Readings in Qualitative Reasoning about Physical Systems*, Morgan-Kaufmann, San Mateo, CA (1989).
- [Williams90] B. C. Williams: Interaction-based Invention: Designing Novel Devices from First Principles, In *Proceedings of the 4th International Workshop on Qualitative Physics*, pp. 161-168 (1990).
- [Williams91] B. C. Williams: Critical Abstraction: Generating Simplest Models to Causal Explanation, In B. Kuipers(ed.): *Proceedings of 5th International Workshop on Qualitative Reasoning about Physical Systems*, pp. 77-92 (1991).
- [Yamaguchi88] 山口高平, 溝口理一郎, 角所収: エキスパートシステムにおける深い推論機構の統一的枠組み, 人工知能学会研究会資料 SIG-KBS-8801, pp. 23-31 (1988).
- [Yamasaki89] 山崎弘郎: 知能化センシング技術, 精密工学会誌, Vol. 55, No. 9, pp.1562-1567 (1989).
- [Yoshikawa72] 吉川弘之: 機械のトポロジ, 精密機械, Vol. 38, No. 12, pp.30-35 (1972).
- [Yoshikawa79a] 吉川弘之: 一般設計学序説, 精密機械, Vol. 45, No. 8, pp.906-912 (1979).
- [Yoshikawa79b] 吉川弘之: 信頼性工学, コロナ社 (1979).
- [Yoshikawa83] 吉川弘之: 保全技術の必要性と可能性, 精密機械, Vol. 49, No. 1, pp.78-83 (1983).



## 発表論文

## 投稿論文他（査読付）

- [1] 梅田靖, 小池雄一, 佐藤隆, 富山哲男, 吉川弘之: 対象モデルに基づく定性物理を用いた故障診断. 精密工学会誌. (in printing).
- [2] Y. Umeda, T. Tomiyama, and H. Yoshikawa: Model Based Diagnosis Using Qualitative Reasoning. In F. Kimura and A. Rolstadås, editors, *Computer Applications in Production and Engineering CAPE'89*, pp. 443-450, North-Holland, Amsterdam, 1989.
- [3] Y. Umeda, H. Takeda, T. Tomiyama, and H. Yoshikawa: Function, Behaviour, and Structure. In J.S. Gero, editor, *Applications of Artificial Intelligence in Engineering V*, volume 1, pp. 177-194, Springer-Verlag, Berlin, 1990.
- [4] Y. Umeda, T. Tomiyama, and H. Yoshikawa: A Design Methodology for a Self-Maintenance Machine In *Design Theory and Methodology (DTM '91)*, pp. 143-150, The American Society of Mechanical Engineers (ASME), 1991.

## 講演論文他

- [1] 梅田靖, 下村芳樹, 佐藤隆, 富山哲男, 吉川弘之: 対象モデルに基づく定性物理を用いた故障シミュレータ. 昭和63年度精密工学会春季大会講演論文集, pp. 95-96, 明治大, 神奈川, 1988.
- [2] 下村芳樹, 浅間一, 佐藤隆, 梅田靖, 吉川弘之: 故障診断の為の機械対象表現の知識工学的手法. 昭和63年度精密工学会春季大会講演論文集, pp. 107-108, 明治大, 神奈川, 1988.



- [3] 梅田靖, 佐藤隆, 富山哲男, 吉川弘之: CAD からの情報を用いた故障診断. 第6回設計自動化工学講演会講演論文集, pp. 10-12, 東工大, 東京, 1988.
- [4] 梅田靖, 佐藤隆, 車谷浩一, 富山哲男, 吉川弘之: 対象モデルに基づく定性物理を用いた故障診断. 昭和63年度精密工学会秋季大会講演論文集, pp. 119-120, 山形大, 山形, 1988.
- [5] 佐藤隆, 井上真治, 梅田靖, 富山哲男, 吉川弘之: 定性物理を用いた故障シミュレーションの詳細化. 1989年度精密工学会春季大会講演論文集, pp. 853-854, 千葉工大, 千葉, 1988.
- [6] 梅田靖, 帖佐太一, 富山哲男, 吉川弘之: 定性プロセス理論を用いた故障シミュレーション. 1989年度精密工学会春季大会講演論文集, pp. 855-856, 千葉工大, 千葉, 1989.
- [7] 梅田靖, 佐藤隆, 富山哲男, 吉川弘之: 対象モデルに基づく定性物理を用いた故障診断. 1989年度人工知能学会全国大会(第3回)講演論文集, pp. 263-266, 学習院大, 東京, 1989.
- [8] 梅田靖, 富山哲男, 吉川弘之: 対象モデルに基づく定性物理を用いた故障診断. 東京大学工学部総合試験所年報, Volume 48, pp. 55-60, 1989.
- [9] 佐藤隆, 梅田靖, 富山哲男, 吉川弘之: 機械の構造と挙動の知識を用いた修復計画の生成. 1989年度精密工学会秋季大会講演論文集, pp. 573-574, 大阪大, 大阪, 1989.
- [10] 富山哲男, 桐山孝司, 梅田靖, 山本文緒, 吉川弘之: 定性物理の設計・保全への応用. 東京大学工学部紀要 A-27, pp. 24-25, 1989.
- [11] 梅田靖, 武田英明, 富山哲男, 吉川弘之: 機能, 挙動, および, 構造の関係について. 1990年度精密工学会春季大会講演論文集, pp. 307-308, 東工大, 東京, 1990.
- [12] 黒木文明, 梅田靖, 富山哲男, 吉川弘之: 幾何情報を用いた故障診断. 1990年度精密工学会春季大会講演論文集, pp. 677-678, 東工大, 東京, 1990.
- [13] 小池雄一, 佐藤隆, 梅田靖, 富山哲男, 吉川弘之: 故障発生メカニズムの定性物理による表現. 1990年度精密工学会春季大会講演論文集, pp. 679-680, 東工大, 東京, 1990.
- [14] 梅田靖, 佐藤隆, 富山哲男, 吉川弘之: 単一故障発生過程仮説に基づく故障診断. 1990年度人工知能学会全国大会(第4回)講演論文集, pp. 483-486, 学習院大, 東京, 1990.

- [15] 梅田靖, 薛德意, 桐山孝司, 富山哲男, 吉川弘之: Smalltalk-80による故障診断システム - 吉川研究室における Smalltalk-80 の利用. 第1回 Smalltalk-80 シンポジウム論文集, pp. 40-51, 1990.
- [16] 梅田靖, 下村芳樹, 富山哲男, 吉川弘之: 自己修復機械の構築(第1報) - 定性物理による対象モデルを用いた制御的アプローチ. 1990年度精密工学会秋季大会講演論文集, pp. 1237-1238, 北海道大, 北海道, 1990.
- [17] 梅田靖, 富山哲男, 吉川弘之: アプリケーションから見たオブジェクト指向 - Smalltalk-80 を中心として -. 情報処理学会研究報告 90-FI-20, Vol. 90, No. 103, pp. 1-8, 情報処理学会, 1990.
- [18] 小池雄一, 梅田靖, 富山哲男, 吉川弘之: 対象モデルに基づく定性物理を用いた統合化保全エキスパートシステムの構築. 人工知能研究会資料 SIG-KBS-9004, pp. 49-56, 人工知能学会, 1991.
- [19] 石井理貴, 梅田靖, 桐山孝司, 富山哲男, 吉川弘之: 概念設計のための機能のモデリング. 1991年度精密工学会春季大会講演論文集, pp. 31-32, 早稲田大, 東京, 1991.
- [20] 小池雄一, 梅田靖, 富山哲男, 吉川弘之: 細胞型機械の構築(第一報) - 概念と基本構成 -. 1991年度精密工学会春季大会講演論文集, pp. 553-554, 早稲田大, 東京, 1991.
- [21] 梅田靖, 富山哲男, 吉川弘之: 機能冗長を用いた高信頼性機械システム構築法. 1991年度精密工学会春季大会講演論文集, pp. 767-768, 早稲田大, 東京, 1991.
- [22] 和田吉樹, 小池雄一, 梅田靖, 富山哲男, 吉川弘之: 統合化保全システムの構築. 1991年度精密工学会春季大会講演論文集, pp. 937-938, 早稲田大, 東京, 1991.
- [23] 茂木康男, 小池雄一, 梅田靖, 富山哲男, 吉川弘之: 故障診断のための定性推論へのファジィ理論の導入. 1991年度精密工学会春季大会講演論文集, pp. 939-940, 早稲田大, 東京, 1991.
- [24] 梅田靖, 茂木康男, 佐藤隆, 富山哲男, 吉川弘之: 故障診断のための定性推論へのファジィ理論の導入. 1991年度人工知能学会全国大会(第5回)講演論文集, pp. 273-276, 学習院大, 東京, 1991.
- [25] 下村芳樹, 谷川貞夫, 梅田靖, 富山哲男, 吉川弘之: 事例推論の修復戦略への適用(第2報) - 事例分類の詳細化 -. 1991年度精密工学会秋季大会講演論文集, pp. 487-488, 静岡大, 静岡, 1991.



- [26] 梅田靖, 小池雄一, 下村芳樹, 富山哲男, 吉川弘之: 自己修復機械設計方法論の提言. 第1回設計工学システム部門講演会講演論文集, pp. 149-154, 日本機械学会, 川崎, 1992.
- [27] 下村芳樹, 谷川貞夫, 小池雄一, 梅田靖, 富山哲男, 吉川弘之: 自己修復機械の構築 (第2報) - 仮想事例を用いた実用化へのアプローチ -. 1992年度精密工学会春季大会講演論文集, 1992, (in printing).
- [28] 梅田靖, 小池雄一, 下村芳樹, 富山哲男, 吉川弘之: 自己修復機械の構築 (第3報) - 機能保全を目的とした機能冗長の実現 -. 1992年度精密工学会春季大会講演論文集, 1992, (in printing).
- [29] 小池雄一, 梅田靖, 下村芳樹, 富山哲男, 吉川弘之: 自己修復機械の構築 (第4報) - 知識の分散化による論理構造の再構成 -. 1992年度精密工学会春季大会講演論文集, 1992, (in printing).
- [30] 下村芳樹, 梅田靖, 小池雄一, 富山哲男, 吉川弘之: 自己修復型複写機の開発. 日本機械学会第69回通常総会講演会講演論文集, 1992, (in printing).



卷之四 雜著 方注