

高次推論の原子力プラントへの適用性研究  
(研究報告書)

1999年3月

核燃料サイクル開発機構  
大洗工学センター

本資料の全部または一部を複写・複製・転載する場合は、下記にお問い合わせください。

〒319-1194 茨城県那珂郡東海村村松4番地49

核燃料サイクル開発機構

技術展開部 技術協力課

Inquiries about copyright and reproduction should be addressed to :

Technical Cooperation Section,

Technology Management Division,

Japan Nuclear Cycle Development Institute

4-49 Muramatsu, Tokai-mura, Naka-gun, Ibaraki-ken, 319-1194,

Japan

©核燃料サイクル開発機構 (Japan Nuclear Cycle Development Institute)

1999

## 高次推論の原子力プラントへの適用性研究 (研究報告書)

吉川 信治<sup>1</sup>、米川 強<sup>1</sup>、須田 一則<sup>1</sup>、長谷川 信<sup>1</sup>

### 要旨

原子力プラントという複雑なシステムでは、情報処理技術として異常の事象進展から遅延せずに結論を導く実時間性、結論が人間にとって納得できるための透明性、及び膨大な情報から除去することが困難な局所的な欠陥に対する頑健性が求められる。これらは単純なルール型の推論機構では実現不可能なものである。

このため、本報告ではこれらの要件の内、参照される情報に内在する局所的な欠陥に対する頑健性を研究対象とした。一方、これに対処するための手段として、最近進展の目覚ましい制御系に関して調査・検討した上で、情報処理系に求められる要件の抽出を行った。また、比較的単純な診断モジュール間の相互評価については、具体的な2つのアルゴリズムを実際にシステム化し、原子力プラントへの適用性評価を行った。

## Applicability Study of Non-linear Reasoning for Nuclear Plant Operation

Shinji Yoshikawa<sup>1</sup>, Tsuyoshi Yonekawa<sup>1</sup>

Kazunori Suda<sup>1</sup>, Makoto Hasegawa<sup>1</sup>

### Abstract

Achievement of improved operation safety of complex nuclear power plants by decision-making functions based on information processing technologies requires real time concluding to be in-time for plant state evolutions, transparency for human operators to understand the derived conclusion, and robustness against local defects inevitably hidden in huge amount of the referred information. These requirements can not be met by the simple rule-based reasoning. The study introduced in this report put a focus on the robustness of the three requirements. This study derives the required feature of robust decision making, after thinking of possible contribution of fast emerging measurement technologies. Finally, two concrete algorithm based on mutual evaluation of relatively simple diagnostic modules are implemented on computer systems and evaluated in views of applicability.

---

1.Beam Technology Development Gr, System Engineering Technology Division, OEC

目次

1. 緒言 .....	1
2. 人工知能技術としての高次推論の研究とその一般産業分野での応用.....	4
2.1 推論方法.....	4
2.2 応用例.....	6
3. 原子力プラントの運転における高次推論の位置付け .....	7
3.1 原子力プラントの運転員を支援する情報システムについての検討 .....	7
3.2 事例に見る運転員意思決定支援への要求 .....	9
3.3 高次推論で追求すべき意思決定機能 .....	16
4. 相互評価に基づいた推論機構.....	17
4.1 免疫型診断システム .....	17
4.2 分散物理モデルによるセンサー・アクチュエータバリデーションシステム .....	22
5. 今後の課題と考察 .....	43
5.1 免疫型診断システムに関する課題と考察 .....	43
5.2 適用性検討 .....	44
参考文献 .....	45
Appendix：免疫型診断システムのプログラムソースリスト .....	47

表目次

表3.1	2重漏洩事象の時間経過.....	11
表3.2	スリーマイルアイランド2号炉事故の経緯.....	13-14

図目次

図3.1	スリーマイルアイランド発電所2号機系統図.....	12
図4.1	パイロットソフトウェア動作例.....	22
図4.2	分散制御システムの構成.....	24
図4.3	システム構成.....	26
図4.4	センサーエージェント動作.....	27
図4.5	故障検知システム処理フロー.....	30
図4.6	試作対象範囲.....	32
図4.7	エージェント及び物理モデルの関係.....	32
図4.8	センサーエージェントメンバーシップ関数.....	36
図4.9	蒸発器出口蒸気温度センサー故障.....	38
図4.10	給水調節弁開度センサー故障.....	39
図4.11	システム表示画面（故障検知前）.....	40
図4.12	システム表示画面（故障検知後）.....	41

## 1. 緒言

原子力プラントは、大規模かつ複雑な巨大システムであり、安全性に対する要求が社会的に非常に高い点で他に例を見ないシステムである。従って、プラント内部での不具合に際しては、原因の同定や対処方法の判断を確実に、しかも過去の事例や経験に依存せずに行わなければならない場合が多い。現在、このような場合の原因同定や対処方法は全て運転員による意思決定に任されているため、意思決定に時間的制約のある場合には運転員は高い認知的な負荷が生じる。運転員のこのような負荷の軽減を目的として、かつ原子力プラントの安全性・信頼性の向上を図るために、高度な意思決定を計算機に行わせるための技術開発が求められている。

原子力の専門家が「安全性」という語を用いた場合、保守的な定義として多重故障や発生頻度が極めて低い異常に対して、プラント外部への放射能の漏れが問題とされることが多い。しかし、放射能の有意な環境放出だけが否定できれば安全であるとの認識が一般公衆の感覚から乖離したものであることは、近年の世論の反応からも明らかである。また、電力市場における競争力向上のためには、プラントの建設費のみならずその稼働率、すなわち経済性の向上が不可欠である。従って、安全保護系やプラントの特性評価により放射能の環境への放出がありえない場合でも、異常の発生に際して、その事象を迅速に同定し（診断）、適切な対応操作（制御）を行う必要性は存在する。このため、安全保護系の作動による原子炉停止に至る場合は、プラント機器への影響を緩和する必要がある。同様に、運転員が不具合を認知できるくらいにプラントの挙動変化が遅い場合でも、これを発見して対処する保守作業も同じく重要である。これは、高速炉のように定格運転時の原子炉の出入口温度差が大きいプラントでは特に重要である。

従って、運転員と協調して信頼性の高い意思決定を迅速に行う情報処理システム開発が要求されており、現在基礎的な研究が各研究機関で盛んに行われている。原子力プラントの運転における意思決定は、言うまでもなく異常発生時の対応が最も重要であり、その一般性のあるモデル化は容易ではない。特筆すべき課題は、過去に経験した、もしくは既に計算機による定量的な再現手法が得られている現象の範囲をはるかに超える広い事象に、迅速かつ確実に対処しなければならないことである。

原子力プラントのような大規模で複雑なシステムにおける異常事象の多くは、ある部品のささいな劣化に端を発し、それが進展してやがて主システムの挙動に影響を与える、という過程を経ることが多いと考えられる。

例えば、

(1)冷却材循環ポンプ駆動モータの潤滑系に不具合が発生



(2)モータ軸受けの発熱量増大



(3)軸受けの材質劣化



(4)モータ回転摩擦増大



(5)モータ回転数減少



(6)ポンプ設置配管系の流量低下



(7)主冷却系のヒートバランス変化

という過程で発生した異常を考えると、何らかの手段で(4)が解除されると(7)までの状態が回復するが、(1)だけが解除されても(3)が回復しないので、(7)までも回復しない。第一の原因が発生してからその影響が運転員に認識されるまでの過程は、このように局所的で不可逆な過程と、大局的で可逆な過程の組み合わせとして記述できる場合が多い。もちろん、配管の漏洩のように認識されるまでが不可逆な過程である場合や、放置されれば不可逆になる過程が点検や監視によりその初期段階で発見、回復される場合もある。原子力プラントの運転における意思決定は、不可逆な過程の進展を抑制し、その再発を防止するための情報処理でなければならない。

従来原子力分野で行われてきた大規模計算は、主に物理的な現象の解明を目的とし、境界条件を全て明確に与えて、対象領域を多数の要素に分割した上で微分方程式を解き、種々の物理量の時間的、空間的な分布を定量的に予測するための情報処理であった。

これに対して、上述の意思決定に求められる情報処理の機能は以下の点で大きく異なる。

- 1)最初に得られる情報は、境界条件ではなく、既に発生した現象の一部である。多くの場合、境界条件はむしろ得るべき結果であり、それを特定するために必要な情報は、情報処理の必要性を認識時点でも全て与えられていない。結果を得るためにさらに必要な情報は何か、ということをも導く機能が要求される。
- 2)得られる情報に定量的な精度が期待できない。また、数値のみで構成される出力では目的を達せず、工学的な意味を結果として出力する必要がある。
- 3)実験による検証が、数値解析システムについて用いられるような定義に照らせば、極めて困難である。従って、動作の各段階が人間から把握できる視認性、正解である可能性のあるものを列挙する機能が必要になる。

これらの機能を実現するために、方程式を解くのではなく、人間の日常的な認識の対象となる「知識」を計算機で処理する技術として研究されてきたのが、人工知能と称される情報処理の中の主たる技術である。原子力プラントのように、大規模でかつ安全性への要求が極端に高いシステムでの異常事象に対処するためには、更に以下のような特徴が要求される。

- ①プラントの異常状態の進展を止められるタイミングに判断が間に合わなければならない。つまり、情報処理に割ける時間を意思決定システム自身が行わなければならない。



②プラントのモデルや観測信号等の判断の基となる情報が膨大で、その全てが正しいことを前提条件とすることが現実的ではない。従って、これらの情報に内在する局所的な欠陥に対して柔軟な処理アルゴリズムが必要となる。

③意思決定システムの必要性が、対象とするプラントの複雑さに起因するため、意思決定システムは、信頼性を低下させるような複雑さを排除した構造を有している必要がある。

人工知能の分野では、②の特徴を有する推論システムが比較的早い時期から主要な研究テーマとして取り上げられており、用意した命題の数に対して導くことのできる結論の数が1次関数とならないことから高次推論と呼ばれている。本報告書の題名もこの呼称によっている。本報告書では、上述した3つの要件の内、②と③の特徴を有する意思決定システムについて動向調査、適用性調査及び試作システムを構築し、原子力プラントの異常診断への適用性について検討を行った。

## 2. 人工知能技術としての高次推論の研究とその一般産業分野での応用

基本的とも言うべき推論には、演繹的推論(A → B と A より B を導く型の推論)がある。演繹的推論は、そのメカニズムが大方解明され知識システムに利用されてきた。一方、完全な知識を対象にした従来の演繹的推論では、知識獲得の問題があるため、不完全な知識を扱い演繹的推論を超える知能のメカニズムを解明し、定式化する研究が行われている。このような推論は、前述したように一般的に高次推論と呼んでいる[2-1]。石塚の文献[2-2]では、完全な知識に対して、不完全な知識(常に正しいわけではないような知識)を扱う次世代知識ベースシステムを高次推論機能として捉えそのアプローチを示している。

高次推論の明確な定義はないが、古典的な分類に従えば、概ね以下のような推論方式を挙げることが出来る。

- ・ 事例ベース推論
- ・ 類推
- ・ 仮説推論

常に成立する知識に加え、他の知識と矛盾の可能性を持つ仮説という知識を扱い、与えられたゴールを説明するのに必要な仮説の集合を求めるものである。実用上は推論速度が問題となるためその高速化の研究[2-5]が行われている。

また、生体の免疫系の抗原認識機構を応用した自律分散診断アルゴリズムもある。免疫系は一つ一つの免疫細胞に注目してみると自律分散的に機能しているが、免疫系全体としては、システムレベルで抗原に対抗している。このアルゴリズムを診断応用に適用している例がある[2-3][2-4]。

### 2.1 推論方法

推論の技術動向として、事例ベース推論、ニューラルネット等について調査した。これらは、知識を規則で表現する従来のAIシステムに対し、どちらかといえば、類似事例の活用により推論を行うアプローチをとるやり方の推論方法である。

#### 2.1.1 事例ベース推論

「用意した知識要素の数と導くことのできる結論の数の関係が線形ではない」という意味では、人工知能分野で研究されている非常に広い範囲の手法が含まれているが、本報告書では、緒言に述べたように、「用意した知識ベース(データベースを含む)に含まれる局所的な欠陥によって結論が致命的な影響を受けない」情報処理を取り上げる。高次推論の定義や定式化に関する議論が盛んに行われていたのは、人工知能、中でも知識処理に対する期待が高かった時期である。この時期には、人間の思考を構成する知識は命題としてそのほとんどを記述できるという仮定が研究者の陰な認識としてあり、従って知識を数えることのできるものと捉える立場から高次推論の定義が導かれていた。現在は、人間が日常的な思考や意思決定で用いる情報のほとんどが命題として記述でき

るといふ仮定が余りにも楽観的でかつ素朴過ぎるとの認識が定着しつつあり、段落冒頭で掲げた意味での高次推論という手法の範疇そのものが意味を失ってきている。

### 2.1.2 ニューラルネット

ニューラルネットワークは事例ベース推論の一種とみなすことができる。これは、非常に広い産業分野で応用されている関数学習／発生機構であり、再現性のある入力／出力の例が十分な数を与えさえすれば、パラメータが多数である場合や、非線形性の強い関数に対しても一応の性能が示されることが明らかになりつつある。入力や出力の[0 or 1]を[偽 or 真]に置き換えて知識処理をすることも可能である。関数の入力／出力の例を用意された知識、得られる結合係数ベクトルを導かれた結論とみなせばこれも高次推論である。与える入力／出力の例に大きな誤差を含むものが少数混入しても、学習の段階でいわば統計処理が行われて学習結果には大きな影響は及ばない。そしてこれが広く応用されるに至った一因と考えられる。また、学習後のニューラルネットそのものによる信号処理についても、層毎のノードの数をあらかじめ考慮して学習を行うことにより、入力パラメータの内極く少数に誤差が混入しても正しい出力を得ることが可能であり、これも本報でいう所の高次推論の定義に整合する。但しこの手法については原子力の分野に限っても既に多くの開発、応用例の報告があるので、詳細は略する。

### 2.1.3 データマイニング

類推が進化した形と見ることが出来る。今日の商店における売上げのように、統一された形式に沿って電子的に収集された膨大な量の情報がある場合に、戦略上有益な規則をそこから見出すことは重要である。これを可能にする技術がデータマイニングである。帰納推論と類似点があるが、結論となり得る結論を予め規定せずに、言いかえれば人間の観察や注意では見出しにくい知識の発見を目的に情報処理を行う点が異なる。元となる情報量が膨大であることが前提となるので、言うまでも無く例外を例外と判断する機能を有しており、本報で言うところの高次推論に含まれる。しかし、現状では、基礎的な手法の開発が諸についたばかりである。

### 2.1.4 仮説推論

仮説推論は、結論の候補を予め数個限定して記述しておいて、基となる知識ベースと整合する（説明が生成できる）ものをその中から選択するものである。結論の候補群(A or B or C or ...)を、基となる知識と整合させて記述できれば、知識の処理手法として独立した範疇として扱う必要はない。

### 2.1.5 方式、情報の多様化

計算機による何らかの意思決定を行う時、人間が判断の材料とする情報を何らかの形式に変換する必要があるので、方式の多様化と情報の多様化は不可分の概念となる。こ

の多様化の概念が意思決定の信頼性向上に有効であることを示す初期の研究が北村らによって行われた。この研究では、原子力プラントのプロセスパラメータの挙動に基づく診断を対象にして、パラメータ間の相関、個々のパラメータの定性値、その時間微分等、抽出する特徴を多様化することで、対雑音性が著しく向上することが示されている。

## 2.2 応用例

大規模かつ複雑な原子力プラントなどにおいては、大量の知識や情報から”これだけは確実にいえる”という正確性（確実性）のある情報を迅速に運転員や経営者に導く技術に関する研究が必要であると考えられている。

また、用意された知識群に内在する局所的欠陥に対して頑健で、かつ結論に求められる精度や詳細さに妥協が認められる場合に、それに応じて計算資源を低減できるようなシステム化が求められている。

両者とも知識、情報の絞り込み、利用方法へのAI 技術の適応がキーポイントである。

### 3. 原子力プラントの運転における高次推論の位置付け

本章では、原子力プラントの運転及び保守に関する、信頼性の高い意思決定を計算機に行わせるために必要な情報処理技術と、現状における技術的見通しについて考察し、高次推論技術の位置付けを明らかにする。

#### 3.1 原子力プラントの運転員を支援する情報システムについての検討

##### 3.1.1 原子力プラントで発生する異常事象に対処するための意思決定

運転保守に関わるタスクは、通常、「異常検出、原因同定、対応（復旧）操作立案と実行、保守保全」のようなシーケンシャルタスクとして想定される。しかしこのような見方は現実を正しく反映してはいない。

- (1) 原子力プラントにおいては、その複雑さと巨大性のため、人間や計算機による診断や対応操作の信頼性を保証することや評価することが非常に困難である。このため、主用箇所での温度や流量等のプロセスパラメータがあらかじめ設定された範囲を超えると安全保護系が作動してプラントを緊急停止させる。このような安全保護系は、異常の検知から対応操作へ、その原因同定を経由せず、運転員の意思とは無関係に強制的にプラントを操作するシステムである。現状では、この必要性は否定できないため、あらゆる意思決定システムは安全保護系と共に用いられると想定せざるを得ない。安全保護系の動作条件は、プラントの物理的状態を基に、主要パラメータの代数式間の論理演算でこれを包含するように設定される。従って、原因同定に費やすことのできる時間は、プラントが現象論的に危険域に達する時間ではなく、安全保護系の起動時刻に制限される場合がある。
- (2) 原因が同定される以前に危険回避のための操作を行わざるを得ず、その操作の結果原因同定が困難になる場合もある。例えば、ある箇所の温度が危険な値に近づきつつある時に、原因の特定に先だって出力を下げる操作を行うことは十分に考えられる。この場合、個々の主要パラメータを定格値と比較しての原因の推定は著しく困難になる。
- (3) プラント状態の変化によって原因となる事象が解除される場合がある。弁の固着により炉心の熱を除熱するための冷却材流量が減少し、危険回避のために原子炉を停止する場合、急激なポンプ水頭の減少によって弁の固着が解除されることが考えられる。この場合、固着した弁を同定すること、また固着の状況を解明することは極めて困難になる。

このように、これらの意思決定は予見や操作結果のフィードバックも含んだ多重のシナリオが同時並行的に進行する総合的なものとして捉える必要がある。しかしこの課題を的確に支援できるような強力な技術は、現時点では程遠いのが現状である。従って当面は、上記課題の重要性は十分に認識した上で、事象の時間的な進展に余裕がある初期段階の故障(incipient failure)への対応に目標を限定して、その目的に有

効な知識ベースならびに計測系の満たすべき要件について考察すべきである。

### 3.1.2 プラントの運転/保守のための意思決定に必要な知識ベースと計測系の要件

緒言において、原子力プラントで発生した異常に対処するための知識処理システムの要件を整理したが、高度な推論手法の適用の前提条件としてプラントからの入力情報の情報量や信頼性の向上が必須である。また、大量のプラントパラメータをより良く運転員へ提示することにより、意思決定の信頼性や迅速性を向上させる方策も当然検討されるべきである。ここではこのような方策の検討を行い、本報告書の本題である高次推論について備える。

原子力プラントが運転されはじめた頃から、センサー、アクチュエータ、動的・静的機器等の故障の初期段階を対象とした検査や診断を支援するツールを開発するための研究開発が積極的に行われてきた。その結果、提案されたツールのいくつかは大規模になり、今でもより多くの機器の種々の故障モードを対象に加えるために増大を続けている。要素技術の急速な発展—新たなセンシング手法、信号/知識処理方法論—もまた新たなツールの開発を促した。これらの個々の手法が有益であること、また原子力プラントでの未対応異常の低減や予防保全の効果を改良するのに寄与することは疑いない。しかし、プラントの安全性を実際に高めるためには、使用するツールの単なる増大は常に有益であるとともに効果的であるとは言えない。いくつかの問題を注意深く調査して解決する必要がある。

人間工学的な観点から広く認められているように、大量の情報は貧弱な情報に決して勝つことはない。詳細な監視のためとして計器やグラフ表示の数を増やすことは運転員の認知的負荷を過剰に増大する。

異常が減多に発生しないことが運転員の作業にさらに悪影響を及ぼす、というのは数の増えた計器を監視するための新たな労力が通常は酬いられないからである。

運転/保守に関わる意思決定のための知識ベースと計測系を検討する際には、現存するプラント監視診断系ではなく、将来型の監視診断系を想定して考察を進めることが必要である。ここでは、情報の適時提供、観測の状況適応化、マンマシン系としての適切さなどの要件を備えた監視診断+保守情報提供系を提案、想定する。

このような監視診断系では分散型処理（知識ベースを含む）と能動（またはオンデマンド）計測系が主要な役割を担うことが考えられる。この知識ベースを構築するため、また保守に関わる問題解決を人間機械協調型の枠組みで支援するための技法として、「症候データベース構築」という課題が重要である。この症候データベースは基本的には正常異常を問わず一定周期ごとの時系列データと、異常が疑わしい状況発生時症候の生データとを別々に記憶し、後日のオンデマンド探索に資する。このような分散型症候データベースは、基本的にはプラントの供用期間全体にわたり運用されて情報や知識を蓄積できる成長型ライフサイクル監視系として機能するものである。この監視診断+保守情報提供系は将来において、センシング技術、寿命予測技術、知識

情報処理技術などの革新を柔軟に取り込み、目的指向の知識ベースや意思決定システムを構築するための基盤技術として想定される。しかしこのままの枠組みであってもすでに現場の運転員や保守担当者にとって有用な情報を提供できる支援システムとして機能し得るものであることに注意すべきである。

なお、この検討を通じて、以下のような示唆が得られたことを付け加えておく。

- (1) 運転と保守の統合の必要性をより強く認識するに至ったこと。
- (2) 知識ベースは機器構造や設計データ、目的機能なども格納されるべきではあるが、まずは症候候補だけを対象として、検討を進めることが現実的であること。
- (3) 計測系としては常設系だけを考えることは将来的には制約が大きすぎるので移動型等の随時・能動監視計装を積極的に採用すべきであること。

### 3.2 事例に見る運転員意思決定支援への要求

#### 3.2.1 高速炉の仮想的2重漏洩事象の概要

金属ナトリウム冷却高速炉で、1次主冷却系循環ポンプ出口配管でのナトリウム漏洩と、このループから2次主冷却系のナトリウムへ熱を伝えるための中間熱交換器の伝熱管の破断が同時に発生する。この中間熱交換器は放射化した1次系のナトリウムと水・蒸気系を隔離しながら熱的には結合するためのもので、1次側は2次側よりも圧力を低くして運転する。従って、伝熱管の破断口を通して2次系のナトリウムが1次系へ流入する。ここでの想定では、この2次系から1次系へのナトリウム流入量が1次主循環ポンプ出口での漏洩量を若干上回る。1次系のナトリウム漏洩は、配管からカバーガスへの流出であり、また炉心冷却能力に直接関わる事象であるため、検出装置が2系統設置されている。1つは絶縁体である保温材内で接近して設置された1対の電極が液体金属ナトリウムとの接触により導通することを用いた検出装置で、警報を発生させる。もう一方はカバーガス循環ループ内のフィルタ前後の圧力差を計測するもので、ナトリウム等の液体あるいは異物の付着による圧力損失の増大を検出することを目的としている。このフィルタは圧力損失の増加確認後、とりはずして異物の成分分析を行うことができる。

このシナリオでは、異常の発生に関して運転員に与えられる最初の情報は、1次主循環ポンプ出口でのナトリウム漏洩を知らせる警報である。

この漏洩の発生から予測される観測情報の変化は：

##### (1) 格納容器床下の雰囲気温度の上昇

格納容器床下温度は通常40℃付近であり、1次系コールドレグ配管内を流れる300℃以上のナトリウムが流入すれば温度を上昇させる要因となる。その程度はいうまでもなく床下空間の熱容量とナトリウムの流入流量とによって決まる。

##### (2) 格納容器内放射線線量の上昇

上述のように、1次主冷却系を循環するナトリウムは、燃料ピンから直接熱を除去するのであるから放射化しており、これが配管から漏洩すると、流出した領域の

線量は上昇する。ただし、正常時でも原子炉格納容器内にはバックグラウンド線量があるので、小規模漏洩の場合には線量増加を確認することは困難で、裏付け情報としては用いられない。

(3)原子炉容器内ナトリウム液位の降下

1次主冷却系バウンダリから有意な量のナトリウムが失われれば、原子炉容器のナトリウム液位の維持が行われなくなって液位が降下することが考えられる。この徴候による漏洩の裏付けの可否には、液位計の精度が大きく影響する。

(4)排気筒内の線量の上昇

格納容器内の放射線線量の増加と同じ事象だが、さらに下流の箇所でも測定される点異なる。

(5)主循環ポンプのポンプ内液位の降下

これも原因は原子炉容器内ナトリウム液位の降下と同じく、1次主冷却系バウンダリからの有意な量のナトリウム喪失である。

ナトリウムは熱膨張性を持つので、炉容器やポンプでのナトリウム液位を一定に保つためには、炉容器やポンプからある量のナトリウムを「あふれ」させて下方のオーバーフロータンクに落下させ、そこから専用のポンプでまた主冷却系へ汲み上げる必要がある。従って1次系からナトリウムが失われれば、このオーバーフロータンクの液位は下降する。

従って、警報が誤報か否かを確認するために、運転員は1次主冷却系のナトリウムオーバーフロータンクの液位を確認する。ところがこのシナリオでは、1次系からカバーガス空間への流出量よりも2次系からの流入量の方が大きいため、この液位は上昇する。ここで運転員は混乱し、上述のフィルタ付着物の成分分析をスタッフに依頼するが、この結果が得られるまでには時間がかかる。

運転員は関連すると思われる各兆候を繰り返し確認して、「1次主冷却系からナトリウムが流出している」ことと、しかし「1次主冷却系内のナトリウム量が増加している」ことに突き当たる。それから、1次主冷却系に外側からナトリウムが供給されていること、それは漏洩が発生したループに接続された2次主冷却ループ以外にありえないことから、中間熱交換器の伝熱管破断を結論する。これは、2次主冷却系のオーバーフロータンクの液位の下降によって確認される。

このように、運転員は観測徴候に対して、可能な場合は徴候に対する原因で観測可能なものを確認しようとし、あるいはその徴候を生じうる複数の原因の中で可能性が高いものに限られた他の徴候を確認しようとしながら、次第に原因を絞り込む。

運転員は、警報を含めて中央制御盤の計器から得られる信号をそれぞれ単独では確信しないように、訓練されており、同じ原因から異なる経路を伝わって現れる他の徴候が計器の表示と矛盾しないことを確認するまではそれを後の思考の材料としては用いない。



パラメータ観測の必要性の認識、パラメータの観測、故障仮説の変遷を時間経過と共に表3.1におおまかにまとめる。

表3.1 2重漏洩事象の時間経過

時刻	会話	意図	備考
17:17:04	一次主冷却系漏洩警報確認		仮説:[一次主冷却系漏洩]
17:17:14	炉容器液位の確認依頼	漏洩確認	
17:17:14	床下雰囲気温度の確認依頼	漏洩確認	
17:17:55	報告:炉容器液位変化無し		
17:17:58	報告:1次OFT液位変化無し		
17:19:08	報告:SID差圧上昇		
17:19:23	報告:DPD変化無し		
17:20:11	報告:床下温度変化無し		
17:20:19	報告:炉容器液位変化無し		
17:20:21	報告:格納容器内線量変化無し		
17:20:27	報告:1次OFT液位変化無し		
17:21:24	報告:排気筒モニタ変化無し		
17:23:36	「誤動作かも知れない」	漏洩否定に傾く	
17:24:40	報告:1次OFT液位変化微増		
17:24:47	「どういうこと?」	混乱	
17:25:30	報告:1次循環ポンプA液位微減		
17:25:46	「どういうメカニズム?」	ますます混乱	
17:26:19	炉容器液位確認依頼		
17:26:21	報告:炉容器液位変化無し		
17:26:27	報告:1次OFT液位変化微増		
17:26:29	「何で上がってるの?」	仮説見つからず	
17:32:51	報告:SID上昇継続		
17:32:53	報告:DPD外観変化無し		
17:33:29	DPD差圧確認依頼		
17:35:27	「どっからNa来てるの?」		バウンダリ内Na増加
17:38:43	「レベル計がおかしいのか?」		センサー故障
17:39:36	報告:ポンプBC吐出圧微増		
17:47:35	報告:2次OFT液位微減		
17:47:35	「あっ、チューブラブチャカ」	IHX伝熱管漏洩	
17:48:21	「圧力が1Kありますから」	新仮説支持	IHX伝熱管漏洩
17:48:47	報告:炉容器圧力微増	新仮説支持	

### 3.2.2 スリーマイルアイランド2号炉事故の概要

1979年3月28日午前4時頃(現地時間)、米国ペンシルベニア州にあるスリーマイルアイランド(Three Mile Island)原子力発電所(電気出力959MW/B&W社製加圧軽水型/1978.12運開)において発生した、いわゆるTMI事故[3-1,3-2,3-3]はその後の原子力開発及び安全研究に極めて有用な教訓を与えたことで広く知られている。

事故は復水系機器の故障が発端とり、これに加圧器逃し弁の故障が重なったために、

小破断冷却材喪失(LOCA)が発生し、さらにいくつかの運転操作上の誤判断、誤操作によって炉心損傷に至ったものである。この事故の初期の時系列は表3.2に示す通りである。また、TMI-2の主要系統は図3.1に示すものである。また、TMI事故がこのように重大化した要因は、その後の調査によって、以下の2種類に分類されている。

A.装置の誤動作

- (1)加圧器逃し弁閉不動作 ←B&W社150回の開放異常発生  
[内4回閉不動作異常]
- (2)加圧器水位=炉内水位 ←炉内水位計測器無し[設計上の問題]
- (3)格納容器隔離不実施 ←緊急炉心冷却装置作動時[設計上の問題]

B.運転員誤操作

- (1)補助給水ポンプ出口弁閉運転 ←[保安規定違反]
- (2)緊急炉心冷却装置の早期停止 ←加圧器水位計指示(炉内水位と誤判断)
- (3)原子炉内圧力維持不実施 ←加圧器逃し弁不作動認識せず
- (4)原子炉主冷却系ポンプ停止 ←ポンプ振動による破損を危惧
- (5)格納容器隔離不実施(水溜ポンプ作動) ←通常、緊急炉心冷却装置作動と同時に実施  
[TMI設計では格納容器内圧力が設定値(4psi)以上で自動隔離される]

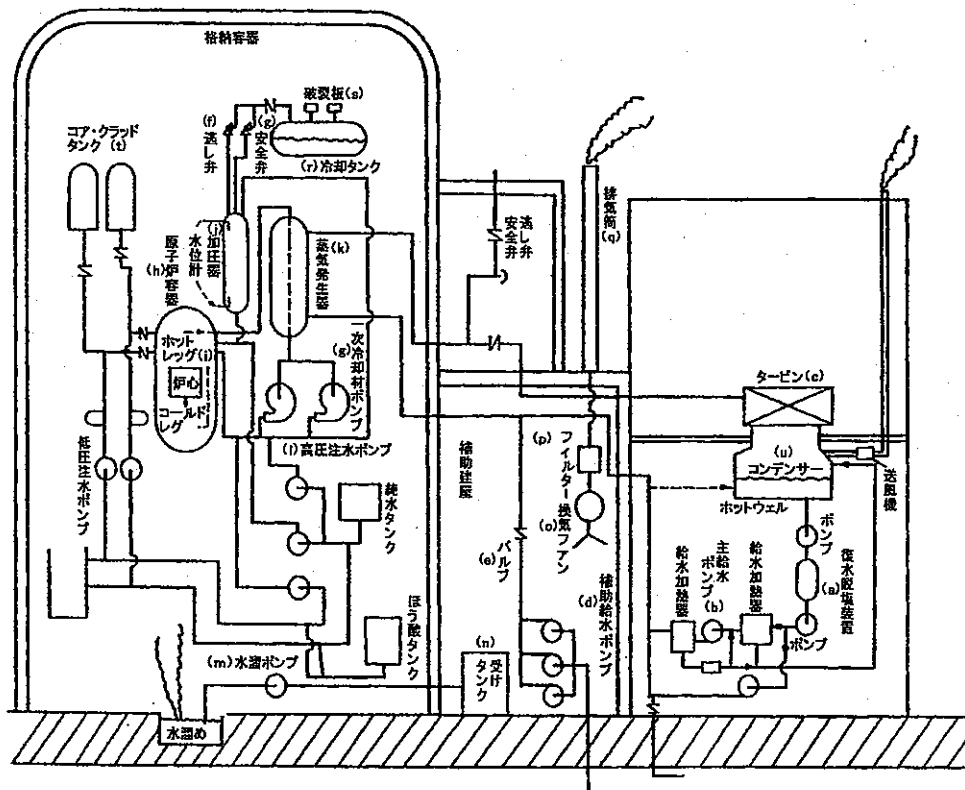


図3.1 スリーマイルアイランド発電所2号機系統図

表3.2 スリーマイルアイランド2号炉事故の経緯 (1/2)

No	時刻	プラント状態	備考
1	(28日午前4時頃)	定格出力の98%で運転中	
2		復水脱塩装置(a)が故障	
3		主給水ポンプ(b)が停止	
4		約5秒後に主給水喪失	
5	T=0秒	タービン自動停止	(ここを時間の基点とする)
6		補助給水ポンプ(d)が自動起動	弁(e)が閉まったままのため、 補助給水は流れず
7		給水喪失のため原子炉冷却系の熱除去がなく、系の圧力上昇	運転員はポンプ作動したため、 給水されていると誤認
8	T=3~6秒	加圧器逃し弁(f)が作動"開"	圧力2255psi
9	T=9~12秒	原子炉(h)スクラム	圧力2355psi
10		加圧器安全弁(g)が作動"開"により、圧力低下開始	
11		1次冷却水、逃し弁から格納容器内冷却タンクへ流出	加圧器逃し弁(f)故障により 閉まらず
12	T=15秒	原子炉冷却系ホットレグ(i)温度=611° F、圧力2147psi	弁故障を運転員気付かず
13	T=30秒	3基の補助給水ポンプ運転継続	弁(c)閉のため、給水不能
14	T=1分	加圧器水位計(i)急上昇	
15		蒸気発生器(k)AとBの2次系水位低下	→数分後空
16	T=2分	ECCS(高圧注水系; e)作動	貫流型のため、U型に比べ保有 水少ない
17	T=4~11分	加圧器水位(j)がスケールを越える	圧力1600psi 運転員は、加圧器が水で充填 され、圧力制御不能になること
18		高圧注水系のポンプ(l)1台を4分30秒に手動停止	2台目のポンプ、10分30秒に 停止
19	T=6分	原子炉冷却系ホットレグ(i)温度=548° F、圧力1350psi	
20	T=7分30秒	格納容器内、水溜ポンプ(m)自動作動	貯留水が補助建屋の受入 タンク(n)へ送出される
21	T=8分	閉まっていた弁(e)を開くことにより、補助給水が流れはじめる	補助建屋通風筒を通して放射 能が外部環境へ放出される
22	T=8分18秒	蒸気発生器(k)Bの圧力最大	
23	T=8分21秒	蒸気発生器(k)Aの圧力が再生し始める	
24	T=11分	圧力が低下し、加圧器水位計(i)がもとに戻る	
25	T=11~12分	高圧注水系給水ポンプ(l)を再作動(運転員操作)	
26	T=15分	逃し弁(f)が開いたままのため、原子炉冷却水排水/冷却タンク(r)の破裂板(s)が破れる	再び、水が補助建屋へ流出
27	T=20~60分	冷却水は温度550° F、圧力1015psiの飽和状態で安定	冷却水量は減少していく
28	T=1時15分	冷却水ポンプ(q)がキャビテーション状態になったため、Bループポンプ	
29	T=1時40分	Aループポンプも停止	ポンプ破損防止のため運転 員により停止
30	T=1時45分~2時	ホットレグ(i)温度が620° Fに上昇。	ホットレグ部に生成された蒸 気が水の流れをせきとめ、 自然対流を阻害
31	T=2時30分	ホットレグ部蒸気泡をつぶすため、逃し弁を閉める	
32	T=3時	原子炉冷却系の圧力が上昇(2150psi)し、逃し弁開く	
33	T=3時15分	原子炉冷却系の排水タンクに圧力5psiのスパイク発生	
34	T=3時24分	高圧注水系を再作動	

表3.2 スリーマイルアイランド2号炉事故の経緯 (2/2)

35	T=3時36分	Ａループへの高圧注水を停止		
36	T=3時48分	原子炉冷却系の排水タンクに圧力11psiのスパイク発生	原子炉冷却系の蒸気とガスが放出され、格納容器内圧力上昇(1→3psig)	
37	T=5時	格納容器内圧力4.5psigに達す		
38	T=5時18分	逃し弁が閉じられ、高圧注水系が再起動する		
39	T=5～6時	原子炉冷却系圧力上昇(1250→2100psi)		
40	T=7時30分	原子炉冷却系を減圧するため、逃し弁を開く(運転員操作)	2050→480psigに低下	(余熱除去装置を作動させるための操作)
41	T=8～9時	炉心クラッドタンク(t)から注水		
42	T=10時	格納容器内で圧力28psigのスパイク発生し、格納容器内スプレー装置作動		
43	T=10時2分	格納容器内スプレー装置停止		
44	T=13時30分	蒸気泡をつぶすために逃し弁を閉め、冷却系ポンプ起動		
45	T=13時30分～16時	原子炉冷却系圧力上昇(650→2300psi)		
46	T=16時	Ａループの冷却ポンプ起動	ホットレグ温度は560° Fに低下、コールドレグ温度は400° Fに上昇	(Ａループ; 蒸気発生器を通じて冷却材が循環)
47	T=16時以降	Ａループ蒸気発生器の蒸気をコンデンサ(u)に流し、コンデンサの真空度を確立させる	冷却水温度290° F、圧力1000psiに保持	
48	4月4日	加圧器のベント弁を通して、水を少量ずつ供給し冷却を継続		

上記要因の中で、運転員的意思決定に関わるものは、起動した緊急炉心冷却装置を停止するという判断(下線部分)が最大のもと考えられる。この判断の基になった観測情報は加圧器水位(運転員は=炉内水位と等価であると認識)であるが、この情報によって炉内圧力が上昇し原子炉容器損傷を招くことを予見/推測したものと考えられ、それ自体の判断は必ずしも誤ったものとは言い難い。一方、TMIの運転員は、加圧器水位と炉内水位が等価であることはマニュアルで明記されていたものの、事故初期段階において、それに反する情報(原子炉冷却系圧力指示が低過ぎる)を得ることができたが、それを確信するまでには至らなかったようである。ここで示された意思決定過程において、矛盾した情報を相互評価することにより、センサー故障の可能性が提示された場合は、せっかく起動した緊急炉心冷却装置を開始直後に(2分30秒後)停止することを更に慎重になったものと考えられ、その後の(一部)炉心損傷を回避できた可能性が高い。

以下は、前述の要因に対し高次推論機構を用いた診断システムに対し期待する効用をQAとして列記したものである。ここでは、あまりにも具体的すぎて3-2の意志決定支援への要求という切り口にはそぐわないものも取り上げた。この内、Qに対するAは現段階で同定できる場合にのみ記した。後の考察の材料としてここに列挙する。

Q1: A-(1)、B-(3) [経過No.11に対して]、プラント状態、機器固有の情報(もし与えられる場合)からこの弁故障を検出することは可能か?

A1:

Q2: A-(2)、 に対して、炉内水位計自体が無いかあったとしても加圧器水位と炉内水位の指示が違っている場合には、どちらが真であるか、あるいはいずれも偽であるかを推測できるか?

A2:

Q3: B-(1)に対して[経過No.6]、補助給水ポンプ自動起動に際し、ポンプ起動したことにより給水されているとの誤った認識(思い込み)を排除することは可能か?

A3:

Q4: B-(4) [経過No.28-29]に対して、ポンプキャビテーション状態のままでも運転継続を推奨できるレベルの情報提供は可能か?

A4:

Q5: B-(5)に対して、運転員による手動隔離が可能であるとした場合に、設計条件より優先して隔離操作を行うことを推奨できるレベルの情報提供は可能か?

A5:

### 3.3 高次推論で追求すべき意思決定機能

今までの事例調査に見られるように、運転員は自分が陽に認識している知識には不完全性が含まれても、その不完全性そのものを認識して、必要に応じて動的に自分の置かれた状況に重要な情報を獲得、確認して意思決定を行う。今まで報告された原子力プラントの異常事象については、センサーバリデーションによって回避されたと思われる事態は列挙できるが、プラントに対する理解の不完全性を情報処理機構が支援しうる状況を明確に記述することは困難である。この章の第1節で述べたような、運転員が処理しなければならない情報を却って増加させるようなシステムは、たとえ最終的な出力結果の信頼性が十分であっても運転員の心的負荷を少なくとも導入後一定期間は増大させると思われる。情報処理技術による意思決定機能と人間の果たすべき機能との最適な協調のあり方に基づいてこのような研究が行われるべきであるが、これは長年の各分野における精力的な努力にも拘わらず、未だに衆知の認める指針を見出すに至っていない。また、緒言で述べた、プラントの事象推移に対して「間に合う」システムの構築の方法も、近年の急速な進展によっても未だ未解決な課題を多く抱えている。

このことを鑑みて、本研究では、「自動化の皮肉」を克服するのに有効であると期待されている、相互評価による分散協調診断アルゴリズムを2例取り上げて、3.1で紹介した原子力プラントでの異常事象への適用性を評価することとした。

## 4. 相互評価に基づいた推論機構

### 4.1 免疫型診断システム

多数の要素知識間の相互評価に基づいて、個々の知識要素毎の、他の大多数の知識との整合性を導く手法は、原子力の診断においては、故障仮説群から観測徴候と矛盾するものを排斥したり、何らかの形で「思いこんで」しまった誤った知識を抽出したりするのに有効であると考えられる。ここで困難なのは、原子力プラントでは同種の故障が多数繰り返されるということが極めてまれなので、異常事象の原因を同定する際に運転員が「思いこんで」いてしかも疑う必要のある知識の中に「機器や設備が壊れていない」という情報があることである。この数は大規模な工学系の代表である原子力プラントの場合は膨大な数になる。実際には現状の運転員は系統間の影響の伝播経路やプラントの各機能とそれを支持している機器設備との関係を用いて、次第に異常原因の発生箇所を狭めていくことで対処している。

しかし、スリーマイルアイランド事故のように、異常の発生によって、計測機器がその信頼性を維持できる範囲を超えた物理条件にさらされて運転員に誤った情報を提示し、さらにプラントの異常時の挙動に対する運転員の誤った理解がこれに加わった結果、重大な結果をもたらすことになる。

#### 4.1.1 アルゴリズムの詳細

この章で取り上げる免疫ネットワーク(イディオタイプネットワーク)は、N.Jerneによって提唱された、免疫系の抗原認識機構である[4-1]。それは、抗原の認識が、それに特異な単一の認識ユニットによって行われるのではなく、その認識の過程で生成された抗体もまた抗原性を有し、他のユニットとの認識対象となるという、相互認識連鎖のネットワークによる認識である。

このアイデアが提唱されてから、多くのモデルが提案され[4-2~6]、そのほとんどは生態モデルと類似のポピュレーションダイナミクスのモデルであり[4-7]、免疫現象を出生死滅の動的なモデルによって説明しようとする。

ここでは、相互認識ネットワークモデルに対する診断アルゴリズム[4-8,9]に、プロセス診断への応用のためにいくつかの改良を加える。それは、実際のプロセスでしばしば見られるように、対象が正確な等式ではなく、またセンサ値の数も十分でない場合にそれを反映した診断を行えるようにするためである。

次に、免疫型並列分散モデルについて述べる。診断理論では、自己診断モデルの診断アルゴリズムが研究されている[4-10]。これらは、主に分散システム、分散処理において耐故障性(fault tolerancy)をいかに実現するかの研究であるが、これらのモデルはまた並列分散モデルとしてみることもできる。したがって、自律分散診断では各々の自律ユニットが他の連結したユニットの意見を聞きながら自ら診断する。これらは、Hopfield Net Algorithm[4-11]に基づいたものである。まず次のような問題を考える。数人の被疑者がおり、それぞれの人がほかの人にたいして犯人である(ない)という証言をしたとき、その証言をもとにして犯人を捜す問題を考

える。各人がそれぞれほかの人に対しその人が犯人か否かの証言をする。証人が犯人でないときは正しい証言をするが、犯人であるときは嘘の証言をするかも知れない。この犯人捜しの問題においては、各人がテストされ、またテストする自律分散型である。これをテストする側とテストされる側に二分した二分論理では、次のようなパラドックスが起こる。

「あるシステムを高信頼化するため、そのシステムにその診断システムを付加する。するとその対象システムの故障に関しては信頼化できるが、診断システムの故障の可能性が残る。その診断システム自体を診断するための“診断システム”を付加すると、その付加した“診断システム”自体の故障の可能性は残る。このように、無限に診断システムの診断システムを付加し続けなければならない。」

この犯人捜しの問題を解くアルゴリズムは自己診断モデルの診断アルゴリズムとして多く提案されている[4-12,13]が、ここでは並列分散モデルによる方法[4-9]を考える。

一般に、自己診断モデルは、それぞれ他のユニットをテストする能力を持ち、また自らも他のユニットによりテストされ得るユニットの相互結合からなる。このモデルは有向グラフ $G(U, E)$ により表わされる。ただし頂点の集合 $U = \{u_i \mid i=1, \dots, n\}$ はユニットの集合に対応し、また有向枝の集合 $E = \{e_{ij} \mid i, j \in \{1, 2, \dots, n\}\}$ はそれらのユニット間の以下のような相互テスト関係を表わす。

- i)  $e_{ij} \in E$ ,  $u_i$ が $u_j$ をテストするとき
- ii)  $e_{ij} \notin E$ , それ以外るとき

上記の犯人捜しの問題は、この有向枝に次のように定義されるテスト値を付加したものと定式化される。

- i)  $u_i$ が犯人でなくかつ $u_j$ が犯人のとき  $\rightarrow T_{ij} = -1$
- ii)  $u_i, u_j$ とも犯人のとき  $\rightarrow T_{ij} = 1$
- iii) それ以外るとき  $\rightarrow T_{ij} = 1 \text{ or } T_{ij} = -1$

この自己診断モデルにはいくつかのバリエーションがある。これはほとんどの自己診断モデルについて再診断性や、診断アルゴリズムが研究されてきた。しかし、それは離散・静的モデルであるため本論文で扱っているような動的モデルには直接適用できない。またこれらの相互認識ネットワークモデルは、並列分散モデルとしてはまだ調べられていない。以下では、これらのモデルを並列分散モデルとみなし、教師なしの学習アルゴリズムにより診断が一種の自己組織化として行えることを示す。



## (自律分散診断アルゴリズム)

自律分散診断とは、各々のユニット自身に単純な情報処理機能を持たせ、それにより互いに他を同時に評価させ、中央処理を一切行わないものである。このときテスト結果と最も整合する信頼度は次の動的モデルにより計算される。

$$dr_i(t)/dt = \sum_j \{T_{ji}R_j(t)\} - \sum_{e_{ij} \in T} (T_{ij} + 1), \quad (1)$$

$$R_i(t) = 1/(1+\exp(-r_i(t)))$$

ここで $T_{ij}^* = T_{ij} + T_{ji}$ である。

また $T_i$ 、は $u_i$ によってなされるすべてのテストの集合である。

$r_i(t)$ は時刻 $t$ におけるユニット $u_i$ の信頼度を反映している。すなわち(1)式の右辺により、ユニット $u_i$ を信頼しうるとするユニットの加重評価が、信頼できないとするユニットのそれより大きいとき、は正となり増加する。

$R_i(t)$ は、この $r_i(t)$ を $[0, 1]$ の範囲に正規化したものである。

(1)式の第1項は、他のユニットのこのユニットに対する意見を総合したものであり、第2項はこのユニット自身の他のユニットに対する意見からくる寄与である。後者の効果を、反射効果と呼ぶことにする。この反射効果は、あるユニットが、信頼度の高いユニットに対し異常だということ、逆にそのユニット自身の信頼度が低く評価されてしまうことに対応している。

以上では、正常なユニットによってなされるテストは完全であると仮定した。すなわち、もし $R_i=1$ ならば $T_{ij}=1$ は $R_j=1$ を意味し、また $T_{ij}=-1$ は $R_j=-1$ を意味する。しかし実際のシステムでは、これは常には成り立たないかも知れない。つぎのセンサ故障診断への応用でも見るように、テストはそれがたとえ正常なユニットによってなされても、そのテストの感度の限界のため必ずしも完全ではない。別言すると、たとえ $R_i=1$ で $T_{ij}=1$ でも $R_j=0$ の場合がありうる。すなわちテストは、この場合、次のように定義される。

- ①テスト側ユニット $u_i$ が正常で、
    - 被テスト側ユニット $u_j$ も正常なとき、 $T_{ij}=1$ となる。
    - 被テスト側ユニット $u_j$ は異常でテストがそれを検知したとき $T_{ij}=-1$ となる。
    - 被テスト側ユニット $u_j$ は異常でテストがそれを検知しなかったとき(テストが不完全なため) $T_{ij}=1$ となる。
  - ②テスト側ユニット $u_i$ が異常のときは、被テスト側ユニット $u_j$ の正常/異常にかかわらず、テスト結果 $T_{ij}$ は1にも-1にもなりうる。すなわちテスト側ユニットの異常のため、それによるテスト結果は全く信頼できないものとなる。
  - ③ユニット $u_i$ から $u_j$ へのテストがないとき、 $T_{ij}=0$ となる、
- まとめると、次のようになる。

- i) テストする側のユニット $u_i$ とテストされる側のユニット $u_j$ が正常なとき  $\rightarrow T_{ij} = 1$
- ii) テストする側のユニット $u_i$ かテストされる側のユニット $u_j$ が異常なとき  $\rightarrow T_{ij} = 1$  または  $-1$  のいずれか
- iii) ユニット $u_i$ からユニット $u_j$ へのテストがないとき  $\rightarrow T_{ij} = 0$ 。

ただし双方向のテストが1であるときは、そのテストを完全なものとする。すなわち $R_i=1$ で $T_{ji}=T_{ij}=1$ のときは、 $R_j=1$ と考える。しかし、 $R_i=1$ でかつ $T_{ij}=1$ でも、必ずしも $R_j=1$ とは限らないので、そのように(1)式の右辺を修正する必要がある。不完全なテストのもとでのモデルは、次のようになる。

$$dr_i(t)/dt = \sum T_{ji}^+ R_j(t) \quad (2)$$

$$R_i(t) = 1 / \{1 + \exp(-r_i(t))\}$$

ここで

- i)  $e_{ij}$ か $e_{ji}$ のいずれか一方が存在するとき  $\rightarrow T_{ji}^+ = T_{ji} + T_{ij} - 1$
- ii)  $e_{ij}$ も $e_{ji}$ も存在しないとき  $\rightarrow T_{ji}^+ = 0$

である。このリアプノフ関数は、次のようになる。

$$V(t) = -(\sum_{i,j=1}^n T_{ji}^+ R_i R_j) / 2$$

上記のモデルが整合診断に収束するのは、このリアプノフ関数の時間微分が次のようになる事からわかる。

$$dV/dt = -\sum_{i=1}^n R'_i(r_i) \{2 \sum_{j=1}^n T_{ji}^+ R_j\}^2 \leq 0$$

センサ診断への応用のため、上記の不完全なテストのもとでのモデルにさらに改訂を行う。上記のモデル(1)、(2)式は、いずれも白黒はっきりした診断を与えるように設計されている。すなわち信頼度は、早く0また1に収束する。しかし、実際の診断の状況では、あるユニットが正常か異常かはっきりと判定できない状況があり(それが正常であるという情報とそうでないという情報が拮抗している場合)、そのようなときには、それをむりに0/1に収束させるのではなく、そのあいまいな状態として診断結果を出力してやる方が良い場合がある、

次のモデル((3)、(4)式)は、完全、不完全テストのそれぞれについて、そのような

あいまいな状態をそのまま診断結果に反映するモデルである。

$$dr_i(t)/dt = \sum T_{ji}^* R_j(t) - \sum (T_{ij} + 1)/2 - r_i(t) \quad (3)$$

$$dr_i(t)/dt = \sum T_{ji}^+ R_j(t) - r_i(t) \quad (4)$$

(3), (4)式はそれぞれ(1), (2)式の右辺に $-r_i(t)$ の修正項を加えたものである。こうすることにより、信頼度はその時点の他のユニットの意見の加重平均を反映したものにとどまろうとする。

(1)~(4)式のモデルをそれぞれ完全テストモデル、不完全テストモデル、あいまい完全テストモデル、あいまい不完全テストモデルと呼ぶことにする。例えば(4)式のあいまい不完全テストモデルでは、平衡点では $r_i(t) = \sum T_{ji}^+ R_j(t)$ が成立する。すなわち $R_i$ は単調に $\sum T_{ji}^+ R_j(t)$ の値を反映する。したがって、 $\sum T_{ji}^+ R_j(t)$ が0に近いとき $R_i$ は0.5近くなる。

#### 4.1.2 免疫型診断手法の今後の課題

参考文献[4-9]においては、このデフォルトの初期設定として(1, ..., 1)が提案されている。これは正常ユニットのほうが異常ユニットより、多いという仮定のもとでは正診断にいつでも近く。この不完全なテストのもとでも良い方策である。しかし実際には、同時の多重故障というのはあまり発生せず。ある単一異常が、次々に他の異常を呼ぶ連鎖異常の方がよく起こる。よって、この初期値設定のもう一つの方策として、新しい異常が起こる直前のRの値を使うことが、この場合に推奨される。またそれぞれのユニットに対し、それらの故障の蓋然性が先験的にわかる場合は、それを初期値としておいてやると、より正確な診断が行える。

#### 4.1.3 パイロットソフトウェアの構築

このソフトウェアはInprise社(旧Borland社)のDelphiを用いて開発した。石田らの開発したオリジナルのアルゴリズムでは、ネットワーク全体における多数決が推論の根拠になっている。原子力プラントの運転等においては、現場に人員を派遣して確認した情報は真実として扱われるので、本研究においては、石田らのアルゴリズムを若干修正し、常に真で、他を評価はするが自身は他から評価を受けない知識ユニットを設定できるようにシステムを構築した。動作例を図4.1に示す。

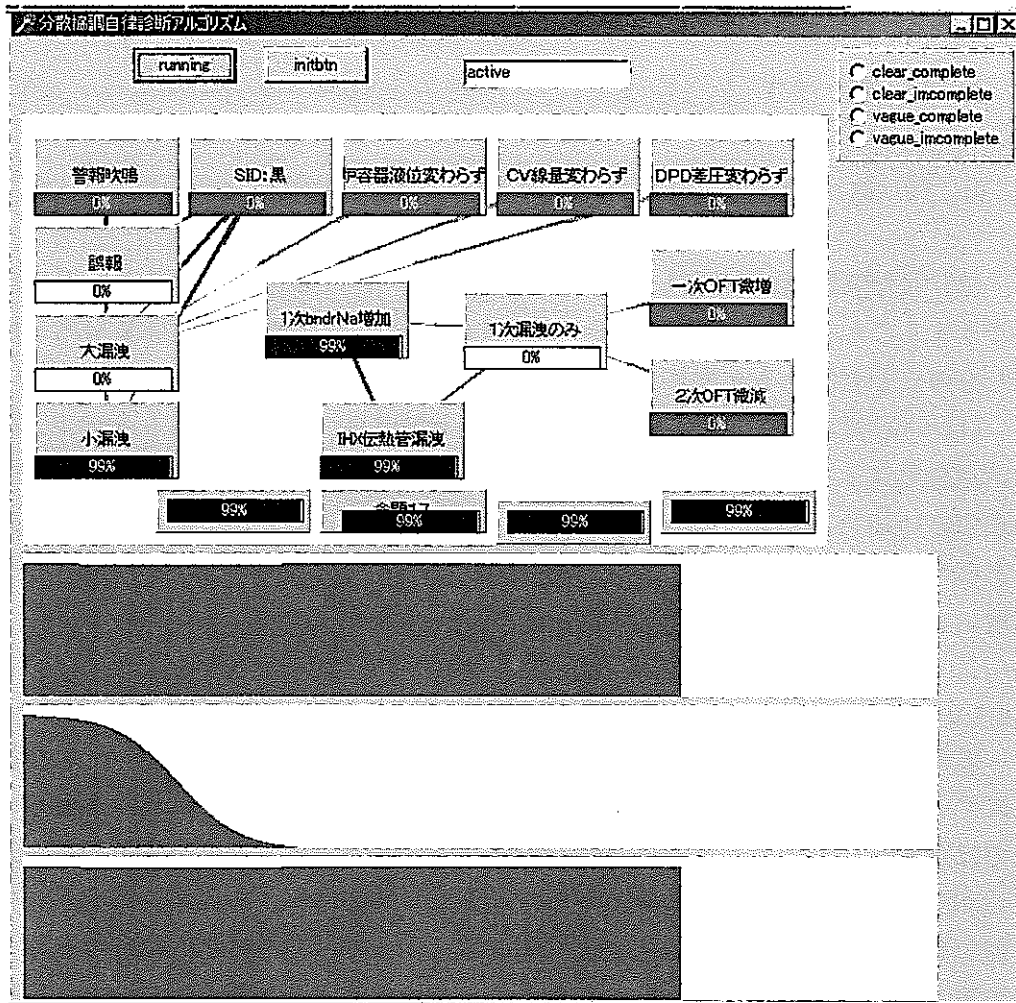


図4.1 パイロットソフトウェア動作例

## 4.2 分散物理モデルによるセンサー・アクチュエータバリデーションシステム

### 4.2.1 本節での研究の背景

本節では、機器の状態監視及び異常診断に必須であるセンサーの健全性に着目し、分散物理モデルに基づいてシステム構築を行い検証することとした。

自律的なシステム [4-14, 15] を構築するためには、系統毎に分散化されたシステムが有機的に結合しあってプラント診断・制御を行い、さらには運転・保守を実施する必要がある。分散協調型のシステムの利点としては、通常時における試験等により一時的に設置されるセンサー群を容易に取り入れられることが挙げられる。そのため、診断や制御の性能向上が図れると考えられる。一方、異常時においては、機器の異常及びセンサーの異常により使用できなくなったセンサーの代替センサーを周囲に取り付けることにより、最低限の運転が可能になると考えられる。また、従来の集中型システムに比べて、システムの改造のしやすさ、そのための作業時間の短

縮化、プラントに異常が発生した場合の原因追求のしやすさが挙げられる。従来の原子力プラントはある程度分散化が進んではいるが、そのデザインは人間によって系の外で集中的に行われている。また、系統毎の大きな枠組みで分散が行われており、その内部のポンプや弁の制御は出力指令により集中管理されている。すなわち、システムの分散は行われているが階層の異なるサブシステム間の上意下達的な協調であり、センサーレベルの協調までは至っていない。そこで、従来システムにおける自動化の向上、過酷な状況下における作業支援を目指すためには、従来のシステムではなく、分散協調によるシステム開発が必要である。

本節では、FBRプラントの最も複雑な系統である水・蒸気系の一部を対象として取り上げる。故障検知・診断評価システムに必要なエージェントの構成、センサーの健全性評価に必要な評価法検討、分散配置されたエージェントの協調に必要な通信形態の検討について試作システムにて検証を行う。

#### 4.2.2 故障検知及び故障診断評価システム検討

従来の原子力プラントにおいては、安全上重要なパラメータ、物理量の計測等の信頼性向上を目的として、過去には計装センサーの多重化などが実施されてきた。しかしながら、広範な計測対象についてセンサーを多重化することはコスト面及び配置上からも困難であると考えられる。そこで、本システムの実現においては、種々の計測量の物理的関係に着目し、異なる種類、箇所の計測量から他の計測量を機器の特性モデル（物理モデル）を用いてセンサー値を推定して、計測値との比較（センサーヴァリデーション）によりセンサーの健全性について評価を行う必要がある。さらに、システムの実装には人工知能分野で検討されている分散協調手法を採用してシステムの拡張性及びセンサーの物理的な多重化と同様な多重化をソフトウェア上で構築し、誤信号に対するプラント診断システムの信頼性の向上を図る。

一般的に、プラントにはシステム状態に冗長性が備わるように同一系統に複数のセンサーが設置され中央制御室等において運転員により監視されている。センサーの健全性を評価する手法の一つとして、全てのセンサーが正常であればそれぞれの計測値より推定したシステムの状態は真の状態に収束するはずであり、仮に一つのセンサーが故障した場合には、推定したシステム状態に誤差が生じ、他のセンサーの計測値から推定した状態と相互比較することにより、どのセンサーが故障したのかを推定することが可能である。また、状態推定に用いるアクチュエータの特性（弁のCv特性値）が変化した場合には、推定した一部の状態が計測値と誤差を生じることにより、どの特性に変化が生じたのか同定することが可能となる。そこで、センサー及びアクチュエータ毎に故障を検知・診断する機能を有するセンサーエージェント及びアクチュエータエージェント（以下、エージェントとする）を設け、エージェントがお互いに協調することにより、全体としてセンサー故障あるいは機器の特性変化が診断可能となる。しかしながら、この方法においても以下に示すような

幾つかの問題点が存在する。

- ・ 計算に用いられる計測値をもたらすセンサーが故障している場合、出力される推定値が真の値から離れ、原因同定が困難になる。
- ・ プラント自体に異常が生じ、計算に用いられているモデルが対応しない場合、センサー故障と誤診する可能性がある。
- ・ 計測ノイズや応答時間の遅れ等により計算誤差が生じ、センサー故障と誤診する可能性がある。
- ・ 物理モデル自体も十分な精度でかつ計算量の小さなものを準備しなければならない。

上述した問題を解決するため、以下では自律的に故障検知及び故障診断評価を行うための分散協調機構検討、エージェント構成、推定値の算出方法、非同期放送通信による自律動作の向上について検討を行う。

#### 4.2.2.1 分散協調機構検討

大規模複雑なプラントにおける保安全管理を自律的に行うシステムの機能、アーキテクチャー等の概念を確立することを主眼として、分散協調手法を用いてシステム開発を行っている。分散協調手法の特徴としては、信頼性、拡張性、操作性、応答性、経済性の向上という特徴を有しており、逆にこのことは従来型の集中制御方式の問題点として挙げられている。分散協調システムの問題点としては管理者がいない完全に平等な形態であるとの意見があるが、必要であれば階層的な組織を作り制御を集中することを妨げるものではないと考える。また、階層的な組織をエージェント群が自ら作り、自ら変えられることができることが重要な点である。すなわち、分散協調システムはもともと独立に存在するエージェント群を陽に意識し、必要に応じて組織し、統合し、調整し、協力できる機能を有するものである。

分散化には大きく分けて二通りあり、図4.2に示すように従来プラントのシステム構成である垂直型の分散、一般的には階層型分散またはヒエラルキー型分散と呼ぶ。もう一つは水平型分散またはネットワーク型分散と呼ばれるものがある。

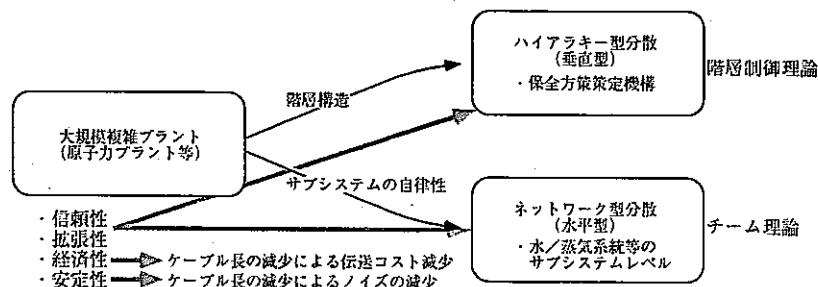


図4.2 分散制御システムの構成[4-16]

知的保全管理システムは故障検知・故障診断評価システム、保全方策策定システム及び過去の履歴を蓄積するデータベースにより構成する。故障検知・故障診断評価システムの役割は、プラントの機能毎に設けられたネットワーク上でセンサー、アクチュエータ、蒸発器・過熱器等の静的機器等に割り付けられた自己判定機能（エージェント）の局所的な情報交換により、故障検知及び故障診断評価を行うことである。保全方策策定システムの役割は、故障検知・故障診断評価結果と予め規定されているプラント運用上の制約事項及び定型的な保全スケジュールデータを記述した保全管理計画に基づき作業工程、作業手順を策定することである。知的保全管理システムの分散化されたシステム構成については、系統毎に管理する多種多数のエージェントとそれらの協調による故障検知・故障診断評価を行うことから水平型の分散システムが最適と考えられる。一方、全体情報と必要とする保全方策策定エージェントと故障検知・故障診断評価結果との関係は垂直型の分散システムが最適と考えられる。よって、知的保全管理システムの分散システム構成は各機能毎のサブシステムレベルにおいては水平分散型のネットワーク構成とする。保全管理システムの全体構成については、運転制御システムと同様に複雑で大規模なシステムをより単純なシステムの結合と捉えることにより、図4.3に示す通りシステム全体の見通しを良くし、システムの設計・製作・保守においても容易になる特徴をもつ階層型分散との複合型の分散システム構成とした。また、サブシステムレベルではエージェントの数が増大することにより、エージェント間の通信コストの増大が考えられる。従って、ここではエージェントは個々に独立とし、従来の共有メモリによる同期を取りながらのデータ通信は行わず、放送通信により必要な時に必要な情報だけを送受信することが可能な非同期放送通信方式[4-17]の採用する。また、上位と下位の階層間の通信には、必要不可欠なデータ以外を通信しない機能を有するゲートウェイを配置し階層型分散における通信のボトルネックを解消する。

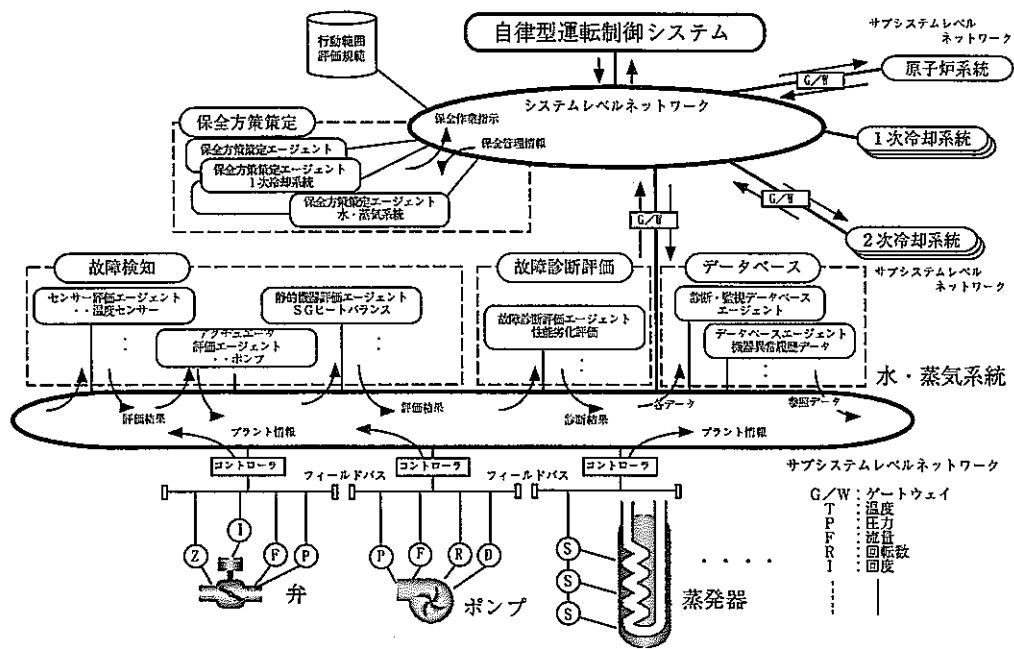


図4.3 システム構成

4.2.2.2 エージェント構成

故障検知・故障診断評価システム開発のケーススタディーとして、異常診断等の情報源となるプラント情報の入力値の正確さを検査するセンサーレベルでのヴァリデーション機能を取り上げ、FBRプラントの蒸発器回りの水・蒸気系統を対象として検討する。本システムは、各種センサーから構成されるデバイス系エージェント群、データの管理を実施する情報系エージェント群より構成される。デバイス系エージェントについては、対象とする系統内にある物理デバイス、すなわち各種センサー／アクチュエータ毎にエージェントを設定した。それらの各エージェントは、自センサー信号もしくは自アクチュエータの特性に限定した範囲でプロセス特性を理解することができるように設計する。情報系エージェントは、異常診断や保全ロボット群とのインタフェース等を司る機構として設計した。本システムでは、上述したように従来のソフトウェアのように階層的なシステム構成ではない。よって、エージェントを管理するような機構は設けず、各エージェントはそれぞれが対等で、自由にデータ交換し合うことにより、各エージェント間を結合する。これらの分散協調の機能により、故障検知・故障診断評価システムは以下の5種類のエージェントを用いて開発した。

- ・ デバイス系エージェント：各種センサーエージェント
- ・ 各種アクチュエータエージェント
- ・ 情報系エージェント       ：異常診断エージェント
- ・ ロボット動作管理エージェント



・共通データ管理エージェント

デバイス系エージェントは、自身が観測するデータ（温度、圧力値など）を周期的に取り入れて評価するとともに、自身が有する複数の物理モデルに基づく評価式を用いて評価を行う。観測データと評価式に必要なデータは、情報系エージェントである共通データ管理エージェントから周期的に取得する。共通データ管理の役割は、各種センサーやアクチュエータが自身の評価式に必要なデータを取り込む状況をエミュレートすることであり、各エージェント独自のタイミング（周期）で共通データ管理エージェントに必要情報を送信するよう依頼する。

情報系エージェントは、デバイス系エージェントからの情報をもとにプラントの異常診断を行ったり、上記のデバイス動作状況をエミュレートしたり、保全ロボット群への保全指令をエミュレートする。従って、これら情報系エージェント機能は、完全な分散協調手法の性質からいえば、各デバイス系エージェントに分散配置されているべきものだが、人間がシステムの状態を総覧できることが望ましいので情報系エージェントを設けることとした。

以下に各エージェントの機能概要を示す。

・センサーエージェント

計測値と、他エージェントが出力した計測値／推定値を用いて算出した自センサーの推定値を比較し、一致度をファジイロジックに従って評価して、推定値グレードを算出するエージェントである。本エージェントは、一定時間間隔で共通データ管理エージェントに対してデータ収集要求を行い、必要データの送付を受ける。また、入手したデータを使用して推定値計算と独立／従属判定データ計算を行う。さらに、算出した推定値と計測値を比較して計測値の評価を行い、計測値整合グレードの算出を行う。最後に、エージェント内で更新したデータについて、共通データ管理エージェントに更新を要求しデータを送付する（図4.4）。

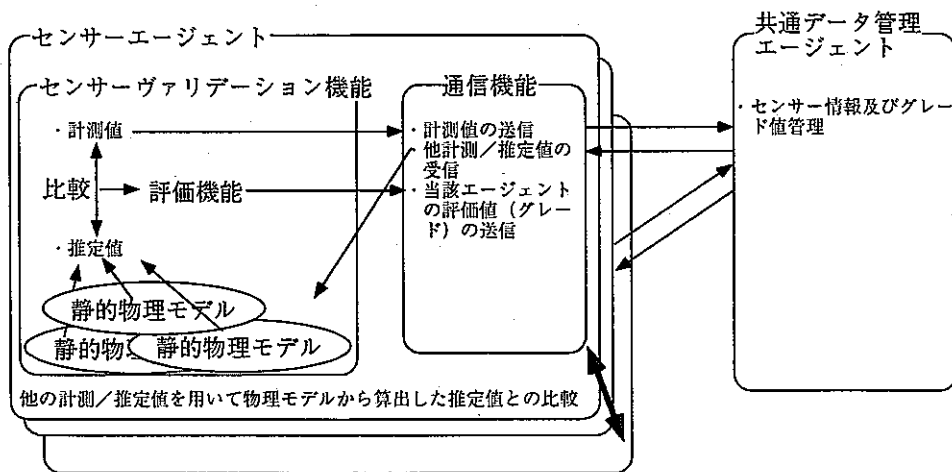


図4.4 センサーエージェント動作

- ・アクチュエータエージェント

機器特性としてのアクチュエータ（弁のCv特性値）の診断を行うエージェントである。アクチュエータ特性値を使用して計算される各センサー値の推定値とセンサー値の計測値をそれぞれ比較することで、機器特性の変化を判断し、推定値グレードを算出する。本エージェントは、一定時間間隔で共通データ管理エージェントに対してデータ収集要求を行い、必要データの送付を受ける。また、入手したデータを使用して、アクチュエータ特性値を使用して計算されるセンサー値についての推定値計算を行う。さらに、算出した推定値と当該センサーの計測値を比較して計測値の整合グレードを算出する。最後に、エージェント内で更新したデータについて、共通データ管理エージェントに更新を要求しデータを送付する。

- ・異常診断エージェント

センサー及びアクチュエータエージェントとは非同期／独立に動作して計測値同定グレード及びアクチュエータ特性値同定グレードを監視し、異常を検知した場合にはロボット動作管理エージェントに対して指示を与えるエージェントである。本エージェントは、共通データ管理エージェントに対して、定期的に計測値同定グレードの収集要求を出し、グレードデータを入手する。また、グレードが一定期間にわたって異常判定用の閾値を下回り続けた場合、当該センサーに異常が発生したものと判断する。さらに、異常発生を検知した場合、ロボット動作管理エージェントに対して異常と判断したセンサー名を通知する。

- ・ロボット動作管理エージェント

故障が発生した時の保守員の代わりとなる自律ロボットへのインタフェースを受け持つエージェントである。本システムでは、異常診断エージェントによる異常センサー名が通知された場合、異常となったセンサー名等を他の計算機にデータを転送し、ロボットへデータを通知する。

- ・共通データ管理エージェント

各センサー信号を知的保全管理システムに取り込むためのインタフェースエージェントである。本システムでは、センサー値の収集及び伝送を行う。本エージェントは、一定時間間隔で計測値データファイルの内容を読み込み、計測値データを更新するとともに、次章で述べるセンサー等の推定値算出数の各種変数の初期化を行う。また、各センサーエージェント、アクチュエータエージェント、異常診断エージェントからのデータ収集要求を受け、データを要求したエージェントに送付する。さらに、各センサーエージェント、アクチュエータエージェントからのデータ更新要求を受け、当該データを各エージェントから受け取って更新する役割を持つ。

#### 4.2.2.3 推定値算出手法検討

各センサーエージェントでは、他センサーの計測値／推定値を使用して自センサーの推定値の算定を行っている。推定値はモデルを使用して算出されたものであるが、1つのモデルが複数のエージェントで使用されている。同一のモデルに属する互いに従属な関数形を重複して用いても独立な推定値は算出できないため、推定値の独立／従属関係を判定することが必要となる。

例えば、3種類の物理モデルによって、温度：T、圧力：P、流量：G、弁開度：Zの各物理量の間以下に示す計9個の関数形が存在する場合について考える。

モデル1： $T = F11(P, G)$ 、 $P = F12(T, G)$ 、 $G = F13(T, P)$

モデル2： $P = F21(G, Z)$ 、 $G = F22(P, Z)$ 、 $Z = F23(P, G)$

モデル3： $T = F31(G, Z)$ 、 $G = F32(T, Z)$ 、 $Z = F33(T, G)$

$F11 \sim F33$ は関数形であり、 $F1i$  ( $i=1 \sim 3$ )、 $F2i$  ( $i=1 \sim 3$ )、 $F3i$  ( $i=1 \sim 3$ )はそれぞれ式を変形しただけなので互いに従属である。

この9個の関数形を使用して温度Tの推定値： $T_j$  ( $j=1 \sim$ )を算出する。この場合は以下の5種類の推定値計算が考えられる。

$T_1 = F11(P, G)$

$T_2 = F31(G, Z)$

$T_3 = F11(F21(G, Z), G)$

$T_4 = F11(P, F32(T, Z))$

$T_5 = F11(F21(G, Z), F32(T, Z))$

この5種類だけが独立であり、他は全てこの5種類のどれかに従属する。

例えば、 $T_6 = F11(P, F22(P, Z))$ というものを考えると、これは「計測値」と「モデル1」と「モデル2」により算出した推定値であるが、この3者の組み合わせの計算は既に $T_3$ で行われている。ゆえに、 $T_3$ が既に存在する場合に新たに $T_6$ の計算を行わないとする。

本システムは分散協調機構を採用するため、各エージェントは独立かつ非同期に動作する。そのため、計算順序やデータ更新順序は不定である。上記の例でいえば、 $T_3$ の計算よりも $T_6$ の計算が先に行われている場合がありうるが、その場合は逆に $T_3$ の計算を行わないとする。

さらに、この例ではモデル3つを仮定したが、実際にはエージェントやモデルの追加によって、独立な推定値計算の数は変化する。本システムでは、組み合わせや個数を規定しない自由度のある設計とした。また、本節では分散協調システムの成立性の検討を主目的とし、使用するモデルは静的なモデルとする。プラント機器の状態推定に蒸発器等、あるいは弁の静的定量モデルを扱うものとして、センサーヴァリデーション及びアクチュエータ特性診断を行った。この場合、定量的に計算されたシステム状態が正常か否か判定する際の閾値に曖昧性が入る余地がある。特に、計測値に雑音がある場合には計測値自体に曖昧性があり、計測値の関係も曖昧関係

として与えられる。そこで、定量的に計算された状態の判定は、それらの曖昧性を考慮できるよう、ファジィアルゴリズムを用いて行う。詳細は後述する推定値グレード算出法にて述べる。センサー故障及びアクチュエータの特性変化の診断を目的とした静的定量モデルを用いる本診断の基本処理フローは図4.5に示す通りである。

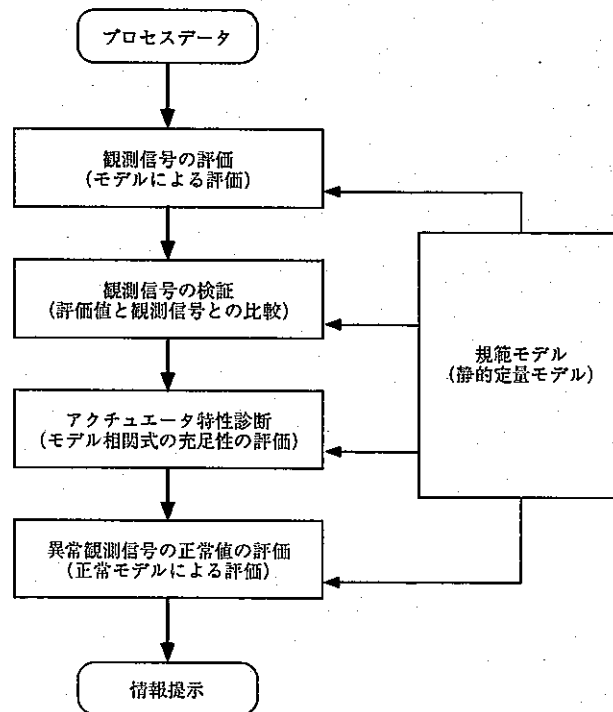


図4.5 故障検知システム処理フロー

#### 4.2.2.4非同期放送通信

各エージェントは他エージェントからの観測データや評価結果を随時受信し、それら情報を基に所有する物理モデルなどにより自分が観測した値の検証を行う。検証のための具体的な処理機能は各エージェント毎に定義され、必要に応じて起動される。すなわち、各エージェントは他のエージェントの動作を直接意識することなく、エージェントの推定値算出に必要なデータがそろえば自律的に処理を開始する。また、センサーエージェントは複数の物理モデルを有することができ、データがそろえば該当するモデルを処理する機能が自律的に起動すると同時に、自身のモデルによる推定値を全エージェントに放送する。このように、各エージェントは他のエージェントの存在を意識することなく、自身が得た情報（計測値データや推定値）を不特定多数のエージェント群に放送すると同時に、自身の物理モデルに必要なデータがそろった時点で処理を行うことで、間接的に他エージェントと協調的に相互作用する。

本システムにおいては、分散配置された各センサー／アクチュエータエージェン

トは独自のタイミングで互いにデータを送信することにより協調する。すなわち、分散協調手法の特徴を最大限に活用できる非同期放送通信方式を用いた上で、システムの自律性について検討する。非同期通信におけるシステムの成立性検討に主眼を置き、本節では静的物理モデルによる試験を行うに留める。

#### 4.2.3 故障検知及び故障診断評価システム試作

故障検知・故障診断評価システムの問題に対し検討を行った。センサー故障に対して検討した点は物理モデルの多重化、非同期放送通信の採用、ファジリアルゴリズムによるノイズによる誤診断の回避が挙げられる。機器の故障に対して検討した点は、機器の状態と物理モデルが乖離した場合に異常箇所の範囲をある程度特定することである。なぜなら、詳細な診断は不可能ではあるが、機器及びセンサーの正常部分の確認は可能であるからである。また、機器の故障診断においては、一つの診断システムによって結論を出すのではなく、他の診断システムとの融合、すなわち診断する対象システムの範囲や出力する診断結果の詳細度の違う診断システムから合意形成を行うことにより信頼性・安全性の向上を図ることが考えられる。ここでは、センサー故障における異常検知について試作システムを構築する。

本システム試作においては、診断情報対象とする各センサー・アクチュエータ毎にエージェントを設定し、そらの各エージェントは、自センサー信号もしくは自アクチュエータの特性に限定した範囲でプロセス特性を理解することができるようにする。全エージェントを管理するような機構は存在せず、各エージェントはそれぞれが対等で、データ交換を非同期放送通信を用いて行う。また、システムの成立性を試作システムにて検証するため、対象範囲をFBRプラントの水・蒸気系の一部を対象として実施し、プロセス特性を理解するための物理モデルとして静的モデルを採用した。

##### 4.2.3.1 試作対象範囲

試作システムは、代表的なFBRプラントを参照プラントとして、過熱器バイパス運転時のセンサー値の正常／異常の診断及びアクチュエータ特性値の変化判定を行えるように実施する。なお、対象とする機器は図4.6に示すように、「蒸発器入口／出口→気水分離器入口／出口→フラッシュタンク」という水・蒸気系の機器とそれに関連するセンサー及びアクチュエータとする。

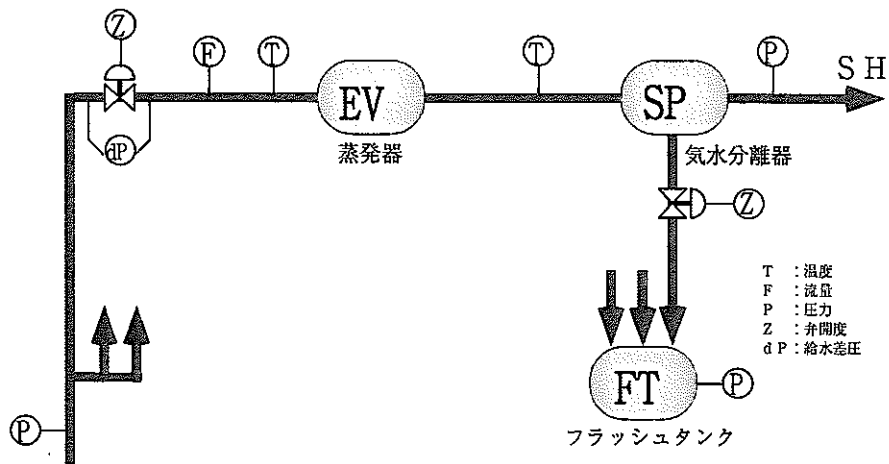


図4.6 試作対象範囲

4.2.3.2 試作モデル概要

「蒸発器入口/出口→気水分離器入口/出口→フラッシュタンク」の系統におけるセンサーヴァリデーシオンに必要なセンサー及びアクチュエータは11個であり、エージェントが使用する物理モデルの関係式は5種類/計8個である。

これらのモデル及びエージェントの関係を図4.7に示し、例として、気水分離器ドレン弁開度センサーエージェントの故障検知及び気水分離器ドレン弁アクチュエータ（弁のCv特性値）の機器特性診断について説明する。

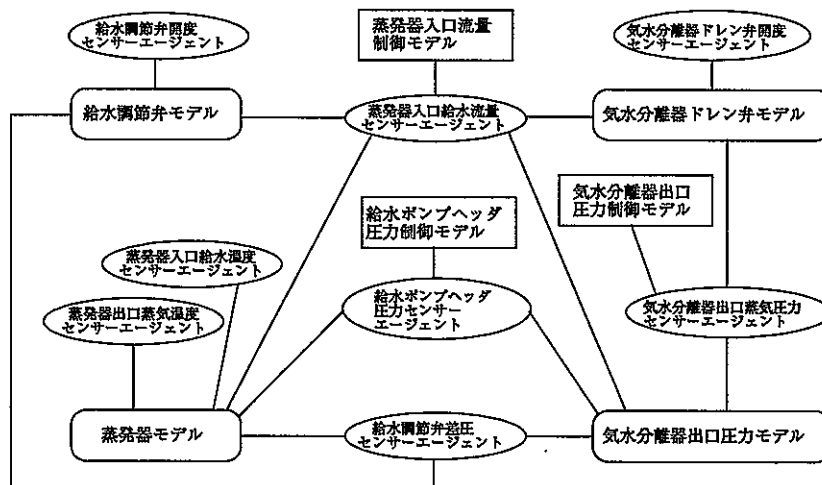


図4.7 エージェント及び物理モデルの関係

・気水分離器ドレン弁開度センサーエージェント

気水分離器ドレン弁モデルを使用し、自センサーの計測値と他センサーエージェント (Gfd, Psp) の計測値または推定値を用いて算出した自センサー値の推定値を比較する。

・気水分離器ドレン弁アクチュエータエージェント

気水分離器ドレン弁モデルを使用し、自センサーの計測値とアクチュエータ特性値を使用して計算される他センサーエージェント (Gfd, Psp, Zsp) の推定値と比較することで機器特性の変化を判断する。

以下に試作するセンサーエージェント、アクチュエータエージェント及び物理モデルを示す。

・センサー及びアクチュエータエージェント

- 給水ポンプ出口ヘッダ圧力センサーエージェント : P h d
- 給水調節弁差圧センサーエージェント : D p
- 給水調節弁開度センサーエージェント : Z f d
- 給水調節弁アクチュエータエージェント : V f d
- 蒸発器給水流量センサーエージェント : G f d
- 蒸発器給水温度センサーエージェント : T f d
- 蒸発器出口蒸気温度センサーエージェント : T s t
- 気水分離器出口蒸気圧力センサーエージェント : P s p
- 気水分離器ドレン弁開度センサーエージェント : Z s p
- 気水分離器ドレン弁アクチュエータエージェント : V s p
- フラッシュタンク圧力センサーエージェント : P f t

・物理モデル

- 給水調節弁モデル
- 気水分離器ドレン弁モデル
- 蒸発器モデル
- 気水分離器出口圧力モデル
- 制御モデル：給水ポンプ出口ヘッダ圧力制御モデル
- 蒸発器給水流量制御モデル
- 気水分離器出口蒸気圧力制御モデル
- フラッシュタンク圧力制御モデル

4.2.3.3 推定値算出法

4.2.3.3.1センサーエージェント推定値算出法

前記した推定値算出法を踏まえ試作システムでは推定値計算の独立/従属判定を以下のようにして実施する。試作システムでは多数のセンサー、アクチュエータとそれに対応した物理モデルが存在する。各モデルに固有のID番号を割当てることによりシステム化を行う。例えば、モデル1に1、モデル2に2、モデル3には3を割当てる。

計測値と推定値を格納する変数を配列で定義する。この場合は、 $T(i)$ とする。 $i=1$ には計測値を格納し、 $i \geq 2$ には推定値を格納する。

$T(i)$ と同時に、 $LT(i)$ 、 $KT(i)$ 、 $GT(i)$ 、という3種類の配列変数と、 $MT$ という1種類のスカラー変数を定義する。

$LT(i)$ は、センサーの種別を格納する変数である。試作システムでは、各計測値に対してセンサーの所属する冷却ループを示すID番号を添付する。具体的には、1次冷却ループのセンサーには1、2次冷却ループのセンサーには2、そして水・蒸気ループのセンサーには3をそれぞれ設定する。 $LT(i)$ には、当該推定値計算に用いたセンサー種別を格納する。

$KT(i)$ は、モデルの種別を格納する変数である。 $KT(i)$ には、当該推定値計算に用いたモデル種別を格納する。

$GT(i)$ は、グレード値を格納する変数である。計測値 ( $i=1$ ) は、推定値との比較を行うことによって、後述する推定値グレードの算出方法でグレード値が計算される。推定値 ( $i \geq 2$ ) は、計算に使用した計測値のグレードのうち最も小さい値をグレード値として持つ。 $GT(i)$ には、計算に使用した計測値/推定値のグレードの中の最小値が格納される。

$MT$ は、推定値の計算個数を格納する変数である。新たに計算した推定値が既存のものと独立であると判断された場合、 $MT$ はカウントアップされ、上記の各配列変数の  $i = MT$  番目に新規に計算したデータが格納される。

新たに推定値計算を行う場合、センサー種別とモデル種別の計算を行い、得られた結果がこれまでに行われた推定値計算の結果にも従属しない場合は新規データとして登録・格納を行い、もしも既存の推定値に従属するものである場合は結果を破棄する。この判定によって、従属する推定値計算を排除することができる。

#### 4.2.3.3.2 センサー値の推定値算出計算の打ち切り精度

各センサーエージェントにおいて、推定値を算出するための使用モデル数が多い場合は、独立な計測値/推定値の組み合わせ数も多くなるので、推定値算出計算回数が膨大なものとなる可能性がある。将来的に計測値が定期的に更新されることを仮定した場合、更新間隔内で推定値算出計算が終了していることが望ましい。

また、モデルには蒸気表の参照部分や簡易近似計算部分があるため、多くのモデルを使用した場合には計算精度が悪化する場合があるものと推測される。

以上の理由から、試作システムではモデルの使用回数に上限を設け、それを越える推定値計算は行わないこととする。このことによって、推定値算出計算回数が制限され、計測値データ更新間隔内で推定値算出計算が終了すると予想される。



#### 4.2.3.3.3 アクチュエータ特性値の推定値算出法

アクチュエータ特性値の推定値は、センサー値と異なり以下のような方法で算出する。

以下のような関数形を持ったアクチュエータモデルを仮定する。

$$G = F41 (P, Z), P = F42 (G, Z), Z = F43 (G, P)$$

ここで、G：流量、P：圧力、Z：弁開度とする。

この時、以下のような計測値と計測値を用いて算出した推定値の比較を実施する。

$$G(1) \doteq F41 (P(1), Z(1))$$

$$P(1) \doteq F42 (G(1), Z(1))$$

$$Z(1) \doteq F43 (G(1), P(1))$$

アクチュエータ値が正常であるならば、つまり機器特性が変化していないならば、比較した両辺はほぼ等しくなるはずである。もしもアクチュエータ値が変化しているならば、比較結果はある有意な差を生じるはずである。

差と計測値の関係から各比較結果についてグレード値を算出し、そのグレード値の中の最小値をアクチュエータ特性値の推定値グレード変数に格納する。

#### 4.2.3.4 推定値グレード算出法

##### 4.2.3.4.1 センサー値の推定値グレード算出法

センサー故障等により計測値が異常となった場合、その計測値と推定値とは偶然の一致以外、本来一致することはない。よって、推定値のいずれかが計測値と一致した場合、その観測信号は正常とみなすことができる。しかしながら、一般的にセンサー信号にはノイズ信号を含むため、正常か異常かを一義的に判定することは難しく、一致の程度を考慮した判定を行う必要がある。そのため、判定のロジックにはファジィロジックを用いる。具体的には、以下のように行うものとする。

- ・ 1次推定値（1つの物理モデル式を用いて評価した値）による判定

ここでは、「1次推定値が計測値に等しい」という状態をメンバーシップ関数により定義し、メンバーシップ関数により定まるグレードで一致しているとみなす。例えば、「T1はTに等しい」という状態のメンバーシップ関数が図4.8のように定義され、T1とTとの偏差が+5であったとする。

この場合、推定値T1は、グレード0.5で観測信号Tに一致しているとみなす。

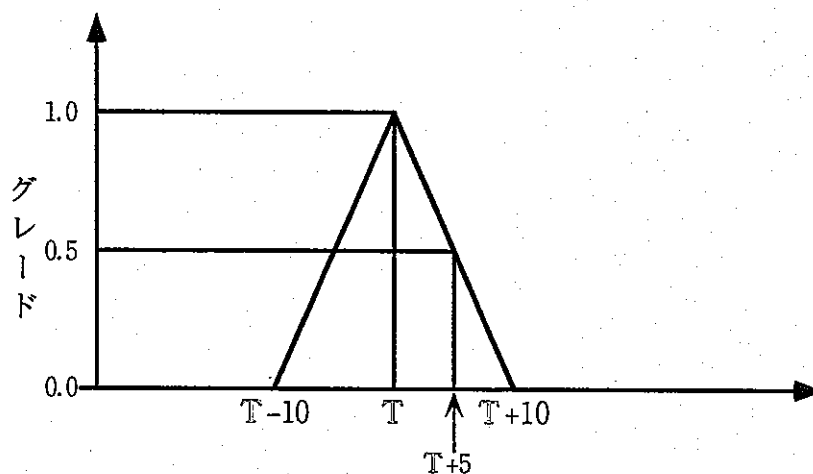


図4.8 センサーエージェントメンバーシップ関数

- ・ N次推定値 (N個の物理モデル式を用いて評価した値) による判定 (N=2以上)  
N次推定値で用いるN次未満の推定値の判定結果をも含め、ファジイロジックにより以下のように判定する。

「N次推定値で用いる全てのN次未満の推定値が観測信号に等しく」かつ「N次推定値が観測信号に等しい」場合「N次推定値は観測信号に一致」。なお、「N次推定値が観測信号に等しい」という状態は、「1次推定値が観測信号に等しい」という状態を表すメンバーシップ関数と同様に定義される。例えば、2次推定値T2が0.6のグレードで観測信号Tに等しく、T2で用いている1次推定値f21(N, F) (=P2)が0.7のグレードで観測信号Pに等しい場合、それらのグレードのMIN値をとり、推定値T2は0.6のグレードで観測信号Tに一致しているとみなす。

- ・ 観測信号の検証

全ての推定値の判定結果に基づき、以下に示すファジイロジックにより観測信号が正常か異常かを判定する。

「推定値のいずれかが観測信号に等しい」場合「観測信号は正常」

つまり、T1が0.5のグレードで観測信号Tに一致し、T2が0.6のグレードで観測信号Tに一致し、T3が0.4のグレードで観測信号Tに一致し、T4が0.7のグレードで観測信号Tに一致し、T5が0.45のグレードで観測信号Tに一致する場合、それらのグレードのMAX値をとり、観測信号Tはグレード0.7で正常であるとする。

#### 4.2.3.4.2 アクチュエータ特性値の推定値グレード算出法

物理モデル式に記述されたアクチュエータの特性が変化していなければ、正常な観測信号を用いて評価した物理モデル式の関係は満足する筈である。よって、正常と判定された観測信号を用いて評価した物理モデル式が、その関係を満足していない場合、その物理モデル式に記述されたアクチュエータの特性は変化しているとみなせる。具体的には、以下のファジィロジックによりアクチュエータの特性変化を判定するものとする。

「ある物理モデル式に用いられている全ての観測信号が正常」かつ「その物理モデル式の関係が満足しない」場合「その物理モデル式に記述されたアクチュエータの特性は変化している」

なお、「物理モデル式関係を満足しない」という状態も、メンバーシップ関数にて定義されるものとする。

また、アクチュエータの特性が変化していないことは、以下のファジィロジックにより判定できる。

「ある物理モデル式に用いられている全ての観測信号が正常」かつ「その物理モデル式関係が満足している」場合「その物理モデル式に記述されたアクチュエータの特性は変化していない」

なお、「物理モデル式関係を満足している」という状態も、メンバーシップ関数にて定義されるものとする。

#### 4.2.3.5 試作システム

本システムは、FBRプラントの水・蒸気系統の一部を対象とし、またプラント状態としては過熱器バイパス運転時を想定して故障検知・故障診断評価が行えるシステムとして開発した。その機能検証試験として、蒸発器出口温度センサー故障と、給水調節弁開度センサー故障について実施した。

その結果、蒸発器出口温度センサー故障を模擬した場合は、他エージェント間の協調により、変更を加えた蒸発器出口温度センサーエージェントだけが正常の判定グレードが0、よって蒸発器出口温度センサーの故障を確認した(図4.9)。蒸発器出口温度センサーの推定値は他の健全センサー値と多重化された物理モデルにより、求めることが可能である。また、故障の度合いをファジィアルゴリズムを用いて1から0の間で正常のグレードを判定できるように開発し、故障による各エージェントの出力するデータの信憑性を確かめることを可能とした。故障判定後は、故障診断エージェントが起動し、故障を自律ロボットへの通知する。

故障診断		
OBJNAME tmht CYCLE 10		
給水調節弁差圧センサー (Dp)		0.992068
蒸発器給水流量センサー (Gfd)		1.000000
原子炉給水ポンプヘッダ圧力センサー (Phd)		1.000000
気水分離器出口蒸気圧力センサー (Psp)		1.000000
フラッシュタンク器内圧力センサー (Pft)		1.000000
蒸発器入口給水温度センサー (Tfd)		0.999999
蒸発器出口蒸気温度センサー (Tst)		0.000000
給水調節弁開度センサー (Zfd)		0.992075
気水分離器ドレン弁開度センサー (Zsp)		0.998328
給水調節弁アクチュエータ (Vfd)		0.992068
気水分離器ドレン弁アクチュエータ (Vsp)		0.996657

図4.9 蒸発器出口蒸気温度センサー故障

給水調節弁開度センサー故障を模擬した場合は、給水流量調節弁開度センサーエージェント以外に給水流量調節弁アクチュエータエージェント及び給水調節弁差圧センサーエージェントの正常グレードが0となる結果となる (図4.10)。これは、給水調節弁の特性 (弁のC V特性) 及び給水調節弁差圧の評価にあたっては、本試作システムでは必ず給水調節弁開度を用いているためであり、センサー信号あるいは機器特性間を記述する物理モデルが冗長化されていないことに起因する。このように、適切な診断がなされていない場合があるが、これは、分散協調機能に起因するものではなく、各エージェント間の情報交換機能、分散化された評価機能については良好な結果を得ることができた。

故障診断	
給水調節弁差圧センサー (Dp)	0.000000
蒸発器給水流量センサー (Gfd)	1.000000
原子炉給水ポンプヘッド圧力センサー (Phd)	1.000000
気水分離器出口蒸気圧力センサー (Psp)	1.000000
フラッシュタンク器内圧力センサー (Pft)	1.000000
蒸発器入口給水温度センサー (Tfd)	0.999999
蒸発器出口蒸気温度センサー (Tst)	0.999988
給水調節弁開度センサー (Zfd)	0.000000
気水分離器ドレン弁開度センサー (Zsp)	0.998328
給水調節弁アクチュエータ (Vfd)	0.000000
気水分離器ドレン弁アクチュエータ (Vsp)	0.996657

図4.10 給水調節弁開度センサー故障

給水調節弁開度センサー故障動作の表示例として図4.11に故障直後の故障診断システムの評価中表示画面、図4.12に故障診断評価後のシステム表示画面を示す。図中では、共通エリア、故障診断、ロボット通知、MMI、センサー及びアクチュエータエージェント（ここでは、給水調節弁開度センサー）のウィンドウから構成され、故障検知・故障診断評価を行うとする。上述した各表示画面について説明すると、以下のように示すことができる。

故障診断ウィンドウでは共通エリアウィンドウにおいて送受信されるデータにより、周期的に診断評価を行う。あるセンサーに故障が起きた場合には、故障診断中の正常のグレード値が低くなり、0の値を数回繰り返して出力した場合に故障と判定し、当該エージェントの故障ウィンドウを表示し（図4.2.10.2、中央）、結果をロボット通知ウィンドウに表示する。なお、故障発生、故障診断後における当該エージェントの削除及びシステムの起動・終了はMMIウィンドウにて実施する。

**共通エリア**

データ受信 Pft  
 データ送信 Zfd データ数 3  
 データ受信 Zfd  
 データ送信 Ysp データ数 4  
 データ受信 Ysp  
 データ送信 Zsp データ数 3  
 データ受信 Zsp  
 データ送信 Tfd データ数 9  
 データ受信 Tfd  
 データ送信 fhnt データ数 11  
 データ送信 Psp データ数 5  
 データ受信 Psp

**故障診断**

\*\*\*\*\*  
 蒸発器給水流量センサー (Gfd) 1.000000  
 原子炉給水ポンプヘッド圧力センサー (Phd) 1.000000  
 \*\*\*\*\*  
 フラッシュタンク器内圧力センサー (Pft) 1.000000  
 \*\*\*\*\*  
 \*\*\*\*\*  
 給水調節弁開度センサー (Zfd) 0.000000  
 \*\*\*\*\*  
 給水調節弁アクチュエータ (Vfd) 0.000000  
 気水分離器ドレン弁アクチュエータ (Vsp) 0.996657

**ロボット通知**

**給水調節弁開度センサー (Zfd)**

グレード値 0.992075  
 使用エージェント  
 給水調節弁開度センサー (Zfd)  
 診断時刻 97/05/09 12:04:45  
 観測値 0.70  
 グレード値 0.000000  
 使用エージェント  
 給水調節弁開度センサー (Zfd)

**MMI**

1. Initialize Environment
2. Object Group Load
3. Object Load
4. Object Delete
5. Send to Method
6. Add to Connect Tree
7. Remove Connect Tree
8. Connect Tree Presentation
9. Object Presentation
10. MMI Exit
11. Monitor Switch
12. Debugger Start
13. Object Copy
14. Object Move

Input No. ? ▲



図4.11 システム表示画面 (故障検知前)

<p><b>共通エリア</b></p> <p>データ送信 Tfd データ数 9          データ受信 Gfd          データ送信 Tfd          データ受信 fhht データ数 11          データ送信 Tst          データ受信 Ysp データ数 4          データ送信 Ysp          データ受信 Zfd データ数 3          データ送信 Zfd          データ受信 Dp データ数 3          データ送信 Dp          データ受信 fhht データ数 11</p>	<p><b>故障診断</b></p> <p>OB_NAME fhht CYCLE 10</p> <p>給水調節弁差圧センサー (Dp) 0.000000          蒸発器給水流速センサー (Gfd) 1.000000          原子炉給水ドレンヘッダ圧力センサー (Phd) 1.000000          気水分離器出口蒸気圧力センサー (Psp) 1.000000          フラッシュタンク器内圧力センサー (Pft) 1.000000          蒸発器入口給水温度センサー (Tfd) 0.999899          蒸発器出口蒸気温度センサー (Tst) 0.999888          給水調節弁開度センサー (Zfd) 0.000000          気水分離器ドレン弁開度センサー (Zsp) 0.998328          給水調節弁アクチュエータ (Vfd) 0.000000          気水分離器ドレン弁アクチュエータ (Vsp) 0.996657</p>	<p><b>ロボット通知</b></p> <p>異常検出 給水調節弁差圧センサー (Dp)          異常検出 給水調節弁開度センサー (Zfd)          異常検出 給水調節弁アクチュエータ (Vfd)          異常検出 給水調節弁差圧センサー (Dp)          異常検出 給水調節弁開度センサー (Zfd)          異常検出 給水調節弁アクチュエータ (Vfd)          異常検出 給水調節弁差圧センサー (Dp)          異常検出 給水調節弁開度センサー (Zfd)          異常検出 給水調節弁アクチュエータ (Vfd)          異常検出 給水調節弁差圧センサー (Dp)          異常検出 給水調節弁開度センサー (Zfd)          異常検出 給水調節弁アクチュエータ (Vfd)</p>
---	---	--

**給水調節弁開度センサー (Zfd)**

グレード値 0.000000  
 使用エージェント  
 給水調節弁開度センサー (Zfd)  
 診断時刻 97/05/08 17:31:48  
 観測値 0.70  
 グレード値 0.000000  
 使用エージェント  
 給水調節弁開度センサー (Zfd)

**給水調節弁アクチュエータ (Vfd)**

診断時刻 97/05/08 17:31:34  
 観測値 0.00  
 グレード値 0.000000  
 使用エージェント:

**MMI**

1. Initialize Environment
2. Object Group Load
3. Object Load
4. Object Delete
5. Send to Method
6. Add to Connect Tree
7. Remove Connect Tree
8. Connect Tree Presentation
9. Object Presentation
10. MMI Exit
11. Monitor Switch
12. Debugger Start
13. Object Copy
14. Object Move

Input No. ?

**給水調節弁差圧センサー (Dp)**

診断時刻 : 97/05/08 17:31:48  
 観測値 : 4.80  
 グレード値 : 0.000000  
 使用エージェント:  
 給水調節弁差圧センサー (Dp)

**給水調節弁開度センサー (Zfd)**

診断時刻 : 97/05/08 17:31:48  
 観測値 : 0.70  
 グレード値 : 0.000000  
 使用エージェント:  
 給水調節弁開度センサー (Zfd)



図4.12 システム表示画面 (故障検知後)

#### 4.2.4 本節の結び

本節では、知的保安全管理システムの概念検討を基に、知的保安全管理システムの分散協調手法による故障検知・故障診断評価システムの検討及び試作システムを用いて性能を評価した。

センサー故障やプラント異常における現状の問題点を2. に述べたように整理し、その解決のため、プラント機器に設置された各センサー値からソフトウェアを用いて相互評価させる手法を採用した。この手法に基づくシステムにより、従来よりも頑健な診断システムを構築できる見通しを得た。本システムでは、各センサーは自センサー値と他センサーからの信号値をもとに計算された推定値を随時比較することにより、センサー故障、弁等の動作機器の劣化状況を診断する。緊急かつ情報が不完全な状況下において、運転員に異常原因を詳細かつ信頼性高く求めることは非常に困難である。従って、情報が不完全な状況下においては、プラントの運転に重要なパラメータや物理量の値が推定値ではあるが計算機によってもたらされることは非常に有用である。さらに、センサー値、推定値の相対的な関係をファジィアルゴリズムを用いて表示し、センサー故障の状態量を示して情報の可視化を行い、運転員における異常時の支援、故障の早期検出、さらには、プラントの信頼性と制御システムの安全性を向上させる見通しを得た。

今後は、動的モデルを採用した分散協調機構を検討するとともに、プラント機器が故障した場合の保全方策を分散協調手法で実現し、プラントの配管、機器等に配置したエージェント及びその上位または下位に配置したエージェントとの情報交換により、プラントを停止せずに補修が行えるかどうかを自律的に判断する保全方策策定機構について検討する必要がある。



## 5. 今後の課題と考察

### 5.1 免疫型診断システムに関する課題と考察

#### 5.1.1 3項以上の論理式の扱い

3.2で取り上げた二重漏洩シナリオで取り上げた運転員による診断のプロセスで決定的なのは、「他に異常が無ければ、S I D (Sodium Ion Detector : ナトリウム漏洩検出器)で1次漏洩が検出され、かつ1次オーバーフロータンク液位が上昇することは、あり得ない」という命題である。この命題は、「他の異常がある」、「S I Dで1次漏洩が検出されている」、「1次オーバーフロータンク液位が上昇する」という命題個々をそれぞれ1個のユニットで表した場合、この手法では扱えないため、改良を施す必要がある。

P, Q, Rの3つの命題があり、「Pであれば、QとRが共に真ではありえない」という場合、これを論理式に表すと、

$$\begin{aligned}
 &P \rightarrow \text{not } (Q \wedge R) \\
 &(\text{not } P) \vee (\text{not } (Q \wedge R)) \\
 &(\text{not } P) \vee ((\text{not } Q) \vee (\text{not } R)) \\
 &(\text{not } (P \wedge Q \wedge R))
 \end{aligned}$$

このような論理式を扱えるように変更することが必要である。

#### 5.1.2 故障仮説の記述

例題として取り上げた二重漏洩シナリオでは、観測徴候全てに適合する原因が、診断を始めた初期段階に用意されていた故障仮説群には含まれない、という運転員の認識が決定的に重要である。従って、予め用意した命題群の間のみで相互評価を行う手法でこのような場合における運転員の意思決定を支援するためには、考察の範囲に含まれる機器全てについて、それが破損しているという命題を用意しなければならないため、現実的ではない。対象範囲の機器やシステムについて、その破損モードを必要に応じて動的に真偽評価の対象にすることができるような、機能やプロセスの記述手法を用いてプラントをモデル化する手法が必要となる。

#### 5.1.3 推論速度の向上方策

本研究で取り上げたアルゴリズムは、互いに直接矛盾の有無を判断できる知識要素相互の評価に基づくものであって、全体を統括するモジュールは存在しない。この機構を実現するために、局所的な相互評価を知識ネットワーク全体に渡って反復し、個々の知識に割り当てられた尤度を動的に評価する手法に基づいている。従って、論理的に距離のある、つまり一方に基づいてもう一方の真偽を評価するまでに演繹の回数を必要とするような知識から知識への尤度評価値の反映には時間がかかる。この問題に対しては上述の運転員による診断のように相互評価機構を階層化す

ることが有効と考えられる。この階層化は下位の層で分散化が可能なことを意味するので、推論速度の向上にさらに有益であろう。

#### 5.1.4 知識要素間の独立性の確保と粒度の整合

本報告書で取り上げた手法は、局所的な相互評価に基づいた動作で局所的な多数決にならないことが利点であるが、知識要素の記述のしかたによってはこの利点が失われる。それは、論理的に似通った知識を多数の個別ノードとして設定すると、それが知識ノード群の中で多数派となる場合があり、論理的には孤立しているにも関わらず肯定される可能性が否定できないからである。従って、知識ノードと知識ノードの間の論理的な独立性を保つ必要がある。あるいは、ある知識ノードと他の知識ノードとの間に等価な部分を見出した場合にはこれを変換して、一方からは真に独立な部分のみを抽出するような機構が必要と考えられる。

### 5.2 適用性検討

#### 5.2.1 設計への応用

複雑なシステムの設計は、その診断と同じく人間に高い負荷を生じる意思決定問題である。

- ・ 時間制約が軽い
- ・ 評価手法が確立されている因子に関しては思考錯誤が許される
- ・ 仕様の微小な変位についてまで、大域的な最適化が要求される

やはり仕様間の直接的なトレードオフのみを記述したネットワークに基づいて大域的な最適化を導くアルゴリズムの基礎となることが考えられる。

#### 5.2.2 核不拡散対策

政情の不安定な地域での核物質の生産や保管については、入手した情報のあるものについては為政者の思惑に起因する加工が施されている可能性があり、核不拡散対策上の対処方法を決定する上に支障を来す。個々の情報要素間の整合／不整合から、膨大な数の情報の内何を信頼すべきか、どのように総合的な評価をするかという意思決定の支援に、本研究で取り上げた手法を有効に活用できる可能性がある。

参考文献

- [2-1]石塚満、“特集「高次人工知能化へのパラダイム」について”、人工知能学会誌、Vol.2, No.1, pp4-5, 1987.
- [2-2]石塚満、“不完全な知識の操作による次世代知識ベース・システムへのアプローチ”、人工知能学会誌、Vol.3, No.5, pp552-562, 1988
- [2-3]石田好輝，“免疫型並列分散モデルによるプラントのセンサ自律診断”，システム制御情報学会論文誌，Vol 7, No.1, pp1-8, 1994.
- [2-4]Yoshiteru Ishida, "The Immune System as a Self-Identification Process: a survey and a proposal" International Workshop on the Immunity-Based Systems 1996(IMBS' 96), 1996.
- [2-5]松尾豊、二田丈之、石塚満，“SL法：線形・非線形計画法の併用によるコストに基づく仮説推論の準最適解計算”，人工知能学会誌，Vol.13, No.6, pp953-961, 1998.
- [3-1]尾崎 禎彦 他：「大規模・実時間知識ベースシステムの開発(1)-基本概念と機能-」, PNC TN9410 96-101
- [3-2]山田 太三郎：「TMI2号炉事故その発端から気泡との闘いまで」, 原子力工業、Vol.25, No.6(1979)
- [3-3]高橋 博：「TMI原発事故-技術面からの考察-」, 原子力工業, Vol.25, No.8(1979)
- [4-1]N.K.Jerne:The Immune System;Sci. Am. Vol.229, No.1, pp.52-60(1973)
- [4-2]H. P.Shwefel and R. Manner (eds.):Parallel Problem Solving from Nature, Springer-Verlag(1991)
- [4-3]A. S. Perelson et al. (eds.):Theoretical Immunology, Part 1, 2, Addison Wesley (1988)
- [4-4]A. Goldberger et al. (eds.) Cell To Cell Signaling :From Experiments to Theoretical Models, Academic Press (1988)
- [4-5]H. Atlan and I. R. Cohen (eds.):Theories of Immune Networks, Springer-Verlag (1978)
- [4-6]R.R. Mohler, C. Bruni and A. Gandolfi: A Systems Approach to Immunology ; Proc. Of the IEEE, Vol. 68, No. 8, pp.964-990 (1980)
- [4-7]S. Levin et al. (eds.):System Theories in Immunology: lecture Notes in Biomathematics, Springer-Verlag(1978)
- [4-8]石田:自己診断モデルの学習型アルゴリズム-免疫ネットワークモデル;信学技報, Vol. PRU-89, No.18, pp.31-38 (1989)
- [4-9]Y. Ishida: Fully Distributed Diagnosis by PDP Learning Algorithm: Towards Immune Network PDP Model; Proc. Of IJCNN 90, pp. 777-782(1990)

- [4-10] M. Malek and K. Y. Liu: Graph Theory Models in Fault Diagnosis and Fault Tolerance, Des. Autom. & Fault Tolerant Comp., 3, Issue 3/4, pp.15-169(1980)
- [4-11] J. J. Hopfield: Neural Network and Physical Systems with Emergent Collective Computational Abilities; Proc. Natl. Acad. Sci. USA, Vol.81, pp.3088-3092 (1982)
- [4-12] Y. Ishida, N. Adachi and H. Tokumaru: Syndrome Decoding Algorithms for Static Diagnosis Models; Int. J. Systems Science, Vol. 18, No.7, pp.1291-1304 (1987)
- [4-13] Y. Ishida, N. Adachi and H. Tokumaru: An Analysis of Self-Diagnosis Model by Conditional Fault Set; Int. J. of Computer and Information Sciences, Vol. 14, No.5, pp.243-259 (1986)
- [4-14] 小澤健二、"知的運転制御システムの開発"、原子力工業、Vol.42-No.5(1996)
- [4-15] 須田一則、米川強、吉川信治、小澤健二、"知的保全管理システムの概念構築"、PNC TN9410 97-049(1997)
- [4-16] 成田誠之助、"分散制御システムの構成"、電子工業月報、Vol.22, No.10, pp.56-81(1980)
- [4-17] 関俊文他、"オブジェクト指向分散システムにおける放送待機冗長処理方式"、T. IEE Japan, Vol. 114-D, No. 3(1994)

Appendix : 免疫型診断システムのプログラムソースリスト

免疫型診断システム動作環境

計算機本体	Gateway2000 G6-300
CPU	Pentium II 300MHz
メモリ	128MB
OS	windows95
プログラミング言語	オブジェクトパスカル (Delphi)

```
unit k_nodes;
```

```
interface
```

```
uses
```

```
Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,  
Gauges, StdCtrls, ExtCtrls;
```

```
type
```

```
TAuto_Diag = class(TForm)
  Panel1: TPanel;
  Panel2: TPanel;
  Panel3: TPanel;
  Gauge1: TGauge;
  Gauge2: TGauge;
  Gauge3: TGauge;
  Panel4: TPanel;
  Panel5: TPanel;
  Panel6: TPanel;
  Gauge4: TGauge;
  Gauge5: TGauge;
  Gauge6: TGauge;
  Panel7: TPanel;
  Panel8: TPanel;
  Panel9: TPanel;
  Gauge7: TGauge;
  Gauge8: TGauge;
  Gauge9: TGauge;
  Timer1: TTimer;
  modebtn: TButton;
  initbtn: TButton;
  Image1: TImage;
  Edit1: TEdit;
  Panel10: TPanel;
  Gauge10: TGauge;
  Panel11: TPanel;
  Gauge11: TGauge;
  Panel12: TPanel;
  Gauge12: TGauge;
  Panel13: TPanel;
  Gauge13: TGauge;
  Panel14: TPanel;
  Gauge14: TGauge;
  Panel15: TPanel;
  Gauge15: TGauge;
  Panel16: TPanel;
  Gauge16: TGauge;
  Panel17: TPanel;
  Gauge17: TGauge;
  Panel18: TPanel;
  RadioButton1: TRadioButton;
  RadioButton2: TRadioButton;
  RadioButton3: TRadioButton;
  RadioButton4: TRadioButton;
  procedure initialize;
  procedure modebtnClick(Sender: TObject);
  procedure FormCreate(Sender: TObject);
  procedure setfrom(i: integer);
```

```

procedure setto(i:integer);
procedure panelClick(Sender: TObject);
procedure proceed(ia:integer);
procedure TimerTimer(Sender: TObject);
procedure InitbtnClick(Sender: TObject);
procedure hypothesize(i:integer);
procedure PanelDb1Click(Sender: TObject);
procedure RadioButtonClick(Sender: TObject);

private
  { Private 宣言 }
public
  { Public 宣言 }
end;

var
  Auto_Diag: TAuto_Diag;
  reputation:array[1..17] of double;
  hypothesized:array[1..17] of boolean;
  setting:boolean;
  connection:array[1..17,1..17,0..1] of boolean;
  population:integer;
  fromto:boolean;
  ifrom, ito:integer;
  posagent:array[1..17,1..2] of integer;
  xfrom,yfrom,xto,yto:integer;
  complete_test:boolean;
  clear_test:boolean;
implementation

  {$R *.DFM}

procedure TAuto_Diag.modebtnClick(Sender: TObject);
begin
  case setting of
    true:begin
      modebtn.caption:='running';
      timer1.interval:=1;
      setting:=false;
    end;
    false:begin
      modebtn.caption:='setting';
      timer1.interval:=0;
      setting:=true;
    end;
  end;
end;

procedure TAuto_Diag.initialize;
var i:integer;
begin
  for i:=1 to population do
    reputation[i]:=5;
  end;
end;

procedure TAuto_Diag.FormCreate(Sender: TObject);
var i,j,k:integer;
    bitmap:tbitmap;
begin

```

```

modebtn.caption:='setting';
setting:=true;
population:=17;
bitmap:=tbitmap.create;
bitmap.height:=450;
bitmap.width:=800;
imager.canvas.rectangle(0,0,450,800);
imager.picture.graphic:=bitmap;
imager.canvas.pen.color:=rgb(0,0,0);
timer1.interval:=0;
complete_test:=true;
clear_test:=true;
initialize;
for i:=1 to population do
begin
  hypothesized[i]:=true;
  for j:=1 to population do
  begin
    connection[i,j,0]:=false;
    connection[i,j,1]:=true;
  end;
end;
gauge1.progress:=0;gauge2.progress:=0;gauge3.progress:=0;
gauge4.progress:=0;gauge5.progress:=0;gauge6.progress:=0;
gauge7.progress:=0;gauge8.progress:=0;gauge9.progress:=0;
gauge10.progress:=0;gauge11.progress:=0;gauge12.progress:=0;
gauge13.progress:=0;gauge14.progress:=0;gauge15.progress:=0;
gauge16.progress:=0;gauge17.progress:=0;
fromto:=false;
posagent[1,1]:=trunc(panel1.left+panel1.width/2);
posagent[1,2]:=trunc(panel1.top+panel1.height/2);
posagent[2,1]:=trunc(panel2.left+panel2.width/2);
posagent[2,2]:=trunc(panel2.top+panel2.height/2);
posagent[3,1]:=trunc(panel3.left+panel3.width/2);
posagent[3,2]:=trunc(panel3.top+panel3.height/2);
posagent[4,1]:=trunc(panel4.left+panel4.width/2);
posagent[4,2]:=trunc(panel4.top+panel4.height/2);
posagent[5,1]:=trunc(panel5.left+panel5.width/2);
posagent[5,2]:=trunc(panel5.top+panel5.height/2);
posagent[6,1]:=trunc(panel6.left+panel6.width/2);
posagent[6,2]:=trunc(panel6.top+panel6.height/2);
posagent[7,1]:=trunc(panel7.left+panel7.width/2);
posagent[7,2]:=trunc(panel7.top+panel7.height/2);
posagent[8,1]:=trunc(panel8.left+panel8.width/2);
posagent[8,2]:=trunc(panel8.top+panel8.height/2);
posagent[9,1]:=trunc(panel9.left+panel9.width/2);
posagent[9,2]:=trunc(panel9.top+panel9.height/2);
posagent[10,1]:=trunc(panel10.left+panel10.width/2);
posagent[10,2]:=trunc(panel10.top+panel10.height/2);
posagent[11,1]:=trunc(panel11.left+panel11.width/2);
posagent[11,2]:=trunc(panel11.top+panel11.height/2);
posagent[12,1]:=trunc(panel12.left+panel12.width/2);
posagent[12,2]:=trunc(panel12.top+panel12.height/2);
posagent[13,1]:=trunc(panel13.left+panel13.width/2);
posagent[13,2]:=trunc(panel13.top+panel13.height/2);
posagent[14,1]:=trunc(panel14.left+panel14.width/2);
posagent[14,2]:=trunc(panel14.top+panel14.height/2);
posagent[15,1]:=trunc(panel15.left+panel15.width/2);
posagent[15,2]:=trunc(panel15.top+panel15.height/2);

```



```

posagent[16,1]:=trunc(panell6.left+panell6.width/2);
posagent[16,2]:=trunc(panell6.top+panell6.height/2);
posagent[17,1]:=trunc(panell7.left+panell7.width/2);
posagent[17,2]:=trunc(panell7.top+panell7.height/2);

end;

procedure TAuto_Diag.setfrom(i:integer);
begin
  if setting then
    begin
      fromto:=true;ifrom:=i;
      xfrom:=posagent[i,1]-imager.left;
      yfrom:=posagent[i,2]-imager.top;
      Imager.canvas.moveto(posagent[i,1]-imager.left,posagent[i,2]-imager.top);
    end;
end;
procedure TAuto_Diag.setto(i:integer);
begin
  fromto:=false;
  if i<>ifrom then
    begin
      if connection[ifrom,i,0] then
        begin
          connection[ifrom,i,1]:=not connection[ifrom,i,1];
          if connection[ifrom,i,1] then
            imager.canvas.pen.color:=rgb(255,0,0)
          else
            imager.canvas.pen.color:=rgb(0,255,0);
          xto:=posagent[i,1]-imager.left;
          yto:=posagent[i,2]-imager.top;
          imager.canvas.Pen.width:=3;
          imager.canvas.lineto(trunc((xfrom*3+xto)/4),trunc((yfrom*3+yto)/4));
          imager.canvas.Pen.width:=2;
          imager.canvas.lineto(trunc((xfrom+xto)/2),trunc((yfrom+yto)/2));
          imager.canvas.Pen.width:=1;
          imager.canvas.lineto(xto,yto);
        end
      else
        begin
          connection[ifrom,i,0]:=true;
          imager.canvas.pen.color:=rgb(255,0,0);
          xto:=posagent[i,1]-imager.left;
          yto:=posagent[i,2]-imager.top;
          imager.canvas.Pen.width:=3;
          imager.canvas.lineto(trunc((xfrom*3+xto)/4),trunc((yfrom*3+yto)/4));
          imager.canvas.Pen.width:=2;
          imager.canvas.lineto(trunc((xfrom+xto)/2),trunc((yfrom+yto)/2));
          imager.canvas.Pen.width:=1;
          imager.canvas.lineto(xto,yto);
          imager.canvas.lineto(posagent[i,1]-imager.left,posagent[i,2]-imager.top);
        end;
      end;
    end;
end;

procedure TAuto_Diag.panelClick(Sender:TObject);
begin
  if sender=panell then
    begin case fromto of false:setfrom(1);true:setto(1); end;end;
end;

```

```

if sender=panel2 then
begin case fromto of false:setfrom(2);true:setto(2); end;end;
if sender=panel3 then
begin case fromto of false:setfrom(3);true:setto(3); end;end;
if sender=panel4 then
begin case fromto of false:setfrom(4);true:setto(4); end;end;
if sender=panel5 then
begin case fromto of false:setfrom(5);true:setto(5); end;end;
if sender=panel6 then
begin case fromto of false:setfrom(6);true:setto(6); end;end;
if sender=panel7 then
begin case fromto of false:setfrom(7);true:setto(7); end;end;
if sender=panel8 then
begin case fromto of false:setfrom(8);true:setto(8); end;end;
if sender=panel9 then
begin case fromto of false:setfrom(9);true:setto(9); end;end;
if sender=panell0 then
begin case fromto of false:setfrom(10);true:setto(10); end;end;
if sender=panell1 then
begin case fromto of false:setfrom(11);true:setto(11); end;end;
if sender=panell2 then
begin case fromto of false:setfrom(12);true:setto(12); end;end;
if sender=panell3 then
begin case fromto of false:setfrom(13);true:setto(13); end;end;
if sender=panell4 then
begin case fromto of false:setfrom(14);true:setto(14); end;end;
if sender=panell5 then
begin case fromto of false:setfrom(15);true:setto(15); end;end;
if sender=panell6 then
begin case fromto of false:setfrom(16);true:setto(16); end;end;
if sender=panell7 then
begin case fromto of false:setfrom(17);true:setto(17); end;end;
end;

```

```

function accord(a,b:boolean):real;
begin
  if a=b then accord:=1 else accord:=-1;
end;

```

```

procedure TAuto_Diag.proceed(ia:integer);
var i:integer;
    term1,term2:real;
    epnl:tpanel;
begin
  case ia of
    1:epnl:=panell1;
    2:epnl:=panel2;
    3:epnl:=panel3;
    4:epnl:=panel4;
    5:epnl:=panel5;
    6:epnl:=panel6;
    7:epnl:=panel7;
    8:epnl:=panel8;
    9:epnl:=panel9;
    10:epnl:=panell10;
    11:epnl:=panell11;
    12:epnl:=panell12;
    13:epnl:=panell13;
    14:epnl:=panell14;

```

```

15:epnl:=panell5;
16:epnl:=panell6;
17:epnl:=panell7;
end;
{epnl.caption:='' ;}
term1:=0;term2:=0;
for i:=1 to population do
if connection[i, ia, 0] then
begin
{epnl.caption:=epnl.caption+inttostr(i);}
term1:=term1
+accord(connection[i, ia, 1], true)*
1/(1+exp(-reputation[i]));
term2:=term2
+2*accord(connection[i, ia, 1], true)+1;
end;
if hypothesized[ia] then
begin
reputation[ia]:=reputation[ia]+(term1*2-term2/2)/100;
if abs(reputation[ia])>5 then
reputation[ia]:=5*reputation[ia]/abs(reputation[ia]);
case ia of
1:gaugel1.progress:=trunc(100/(1+exp(-reputation[ia])));
2:gaugel2.progress:=trunc(100/(1+exp(-reputation[ia])));
3:gaugel3.progress:=trunc(100/(1+exp(-reputation[ia])));
4:gaugel4.progress:=trunc(100/(1+exp(-reputation[ia])));
5:gaugel5.progress:=trunc(100/(1+exp(-reputation[ia])));
6:gaugel6.progress:=trunc(100/(1+exp(-reputation[ia])));
7:gaugel7.progress:=trunc(100/(1+exp(-reputation[ia])));
8:gaugel8.progress:=trunc(100/(1+exp(-reputation[ia])));
9:gaugel9.progress:=trunc(100/(1+exp(-reputation[ia])));
10:gaugel10.progress:=trunc(100/(1+exp(-reputation[ia])));
11:gaugel11.progress:=trunc(100/(1+exp(-reputation[ia])));
12:gaugel12.progress:=trunc(100/(1+exp(-reputation[ia])));
13:gaugel13.progress:=trunc(100/(1+exp(-reputation[ia])));
14:gaugel14.progress:=trunc(100/(1+exp(-reputation[ia])));
15:gaugel15.progress:=trunc(100/(1+exp(-reputation[ia])));
16:gaugel16.progress:=trunc(100/(1+exp(-reputation[ia])));
17:gaugel17.progress:=trunc(100/(1+exp(-reputation[ia])));
end;
end;
edit1.text:=' active' ; {floattostr(reputation[1]);}
end;

procedure TAuto_Diag.Timer1Timer(Sender: TObject);
var i:integer;
begin
for i:=1 to population do
proceed(i);
end;

procedure TAuto_Diag.InitbtnClick(Sender: TObject);
begin
initialize;
end;

procedure TAuto_Diag.hypothesize(i:integer);
var gcol:tcolor;
begin

```

```

hypothesized[i]:=not hypothesized[i];
if hypothesized[i] then gcol:=clWhite
    else gcol:=clRed;

case i of
1: gauge1.BackColor:=gcol;
2: gauge2.BackColor:=gcol;
3: gauge3.BackColor:=gcol;
4: gauge4.BackColor:=gcol;
5: gauge5.BackColor:=gcol;
6: gauge6.BackColor:=gcol;
7: gauge7.BackColor:=gcol;
8: gauge8.BackColor:=gcol;
9: gauge9.BackColor:=gcol;
10: gauge10.BackColor:=gcol;
11: gauge11.BackColor:=gcol;
12: gauge12.BackColor:=gcol;
13: gauge13.BackColor:=gcol;
14: gauge14.BackColor:=gcol;
15: gauge15.BackColor:=gcol;
16: gauge16.BackColor:=gcol;
17: gauge17.BackColor:=gcol;
end;
end;

procedure TAuto_Diag.PanelDb1Click(Sender: TObject);
begin
if sender=panel1 then hypothesize(1);
if sender=panel2 then hypothesize(2);
if sender=panel3 then hypothesize(3);
if sender=panel4 then hypothesize(4);
if sender=panel5 then hypothesize(5);
if sender=panel6 then hypothesize(6);
if sender=panel7 then hypothesize(7);
if sender=panel8 then hypothesize(8);
if sender=panel9 then hypothesize(9);
if sender=panel10 then hypothesize(10);
if sender=panel11 then hypothesize(11);
if sender=panel12 then hypothesize(12);
if sender=panel13 then hypothesize(13);
if sender=panel14 then hypothesize(14);
if sender=panel15 then hypothesize(15);
if sender=panel16 then hypothesize(16);
if sender=panel17 then hypothesize(17);
end;

procedure TAuto_Diag.RadioButtonClick(Sender: TObject);
begin
if sender=RadioButton1 then
begin clear_test:=true;complete_test:=true;end;
if sender=RadioButton1 then
begin clear_test:=true;complete_test:=false;end;
if sender=RadioButton1 then
begin clear_test:=false;complete_test:=true;end;
if sender=RadioButton1 then
begin clear_test:=false;complete_test:=false;end;
end;

end.

```