

インパス事例による矛盾解消法を用いた理論修正

佐々木 和雄 上原 邦昭 前川 禎男

神戸大学工学部情報知能工学科

近年、盛んに研究されている理論形成の分野において、不完全な領域理論をいかにして現実世界に沿うような形に修正していくかという問題が重要なテーマとなっている。従来から用いられてきた理論修正の手法は、一般化ルールにより演繹的に領域理論を修正していくものであった。しかしながら、このような一般化ルールの構築は非常に困難であり、完全な一般化ルールを構築するのは不可能である。本稿では、事例ベース推論のアプローチを採用し、理論の修正法を事例として記憶しながら学習を行ない、新たに理論を修正する際には過去の事例の修正法を利用するシステムを提案する。本システムは、類似した事例をメモリから検索する機構として、Falkenhainerらによって提案された構造写像エンジンを使用している。

The Theory Revision by Adaptation of Impasse Cases

Kazuo Sasaki Kuniaki Uehara Sadao Maekawa

Department of Computer Science and Systems Engineering

Kobe University

Nada, Kobe, 657 Japan

In the field of Scientific discovery, a newly formed theory will seldom be perfect, and its shortcomings will be exposed as familiarity with the domain increases. In order to deal with this deficiency, a number of approaches, called theory revision, have been proposed. Theory revision is to eliminate these shortcomings by augmenting or modifying the imperfect theory. Typically, these approaches make use of the procedural rule set so as to eliminate the impasse included in the imperfect theory. However, one of the problems associated with these approaches is that the construction of the reliable rules is very difficult.

This paper proposes the system which revises the impasse by using Case-Based Reasoning (CBR) approach. CBR is based on appealing to memory of past cases as a more efficient and reliable way to reason about new situations than relying on procedural rules. The basic operation of this system is to match the new situation to an analogous past case from memory and acquire the revision method for the impasse. This system retrieves a closely matching case from memory by means of structure mapping engine proposed by Falkenhainer, Forbus and Genter.

1 序論

近年、人工知能の分野において学習の1つとして理論形成に関する研究が盛んに行なわれている。理論形成に関する研究には、大きく分けて2つの流れが存在する。1つは、ある対象物に関する知識をルール化された領域知識として予め用意しておき、領域知識を用いて演繹推論による観測データの説明を試みる。もし、説明に失敗したならば、発想推論などを用いて領域知識の洗練化を行なうという手法である。もう1つの流れは、対象物の構造的な因果関係を理論として持ち、演繹推論、発想推論、類推に関して明確な区別を持つことなく、統一的なアプローチによってこれらの推論を実現する手法である。本研究は後者の流れに沿ったものである。

後者の代表的なシステムとして、Falkenhainerの提案したPhineas [3] がある。Phineasは、物理現象の観測データが与えられると、それに類似した過去の物理現象を検索し、過去の物理現象の説明に使われた理論を利用して、現在の物理現象を説明するための新たな理論を形成するシステムである。しかしながら、形成された理論は必ずしも観測データをうまく説明できるとは限らない。このため、観測データをうまく説明するために、理論を修正する必要がある場合が生じる。これを理論修正と呼ぶ。本研究では、このような問題を解決するために、過去の理論修正に用いられた修正案を事例として記憶しておき、新たに理論を修正する際には過去の事例からの類推によって修正案を獲得するというアプローチを採っている。

2 事例による領域理論の矛盾解消

一般に、不完全あるいは不正確な領域理論によって観測データ(訓練例)を説明する場合、説明結果とデータとの間に矛盾(インパス)が発生する場合がある。また、本研究で対象としている科学的発見(Scientific Discovery)の分野では、不完全な領

域理論にインパスが発生するようなデータを意図的に与え、インパスを解消するように領域理論を修正しながら理論の洗練化をはかるという手法が用いられることが多い。このため、インパスの解消は機械学習にとって重要な問題である。

インパスを解消する方法として従来から用いられてきた手法として、手続き的な一般化ルールを利用するものがある。たとえば、Rajamoneyの提案したADEPT [8] がこの手法に属している。ADEPTは、インパスによって複数の修正案が提案された場合に、修正案を1つに絞り込むための実験方法を提案するシステムである。ADEPTでは、複数の修正案を導出するために手続き的な一般化ルールを利用している。しかしながら、ADEPTの一般化ルールは、発生するインパスを予測することによって構築されているために、一般化ルールを構築する時点では予測不可能なインパスについては対処できないという問題点を持っている。このため、一般化ルールを適用できないインパスが発生した場合には、その度に新たな一般化ルールを追加する、あるいは既存の一般化ルールを修正するという操作が必要となる。また、一般化ルールを追加するという操作を繰り返していくと、蓄積された一般化ルール間で互いに矛盾が発生する場合がある。このような問題を解決するためには、一般化ルール間の整合性を常に保つようなモジュール、すなわち一般化ルール間の矛盾を解消する機構を追加する必要がある。これは、矛盾解消機構の中に新たな矛盾解消機構を構築することになり、一般には収束しないという危険性を持っている。

これに対し、本研究では、インパスに対する修正案を事例(インパス事例と呼ぶ)として記憶しておき、過去の事例に従って新たなインパスを修正するという手法を採っている。本アプローチは、人間の理論修正メカニズムと類似していると考えられる。たとえば、コンピュータを管理しているスーパーユーザについて考えると、コンピュータに何らかのトラブルが発生した場合、人間は予め

定まったルールに従ってトラブルに対処するのではなく、蓄積された過去の経験から同様なトラブルを検索し、その時点で行なった対処法を現在のトラブルにも適用しようと試みることが多い。もし、トラブルが初めて遭遇するトラブルである場合は、専門書で対処法を調べて、それを新たな経験として蓄積している。このようにして、経験を積んだスーパーユーザは、経験の浅いスーパーユーザよりもより迅速にトラブルに対処することができるようになる。言い換えると、人間はインパスを解消する場合、過去の経験において類似したインパスに対して行なった修正案を、現在発生しているインパスにも適用しようとするということである。

以上のような考え方に基づいて、本研究では、理論修正メカニズムに、Shank の提案した SWALE [7] [9] の戦略を採用している。SWALE では、説明評価機構 (Explanation Evaluator) が事例集合 (Explanation Library: XP) の中の事例を適用して、与えられた問題を説明しようと試みる。説明に失敗すると、トゥイーカ (XP Tweaker) と呼ばれる修正機構が 3 つの修正戦略に従って事例集合内の事例を修正し、再び問題の説明を試みるようになっていく。

トゥイーカによる修正戦略には、事例を特殊化する特殊化戦略、事例を一般化する一般化戦略、ある内容を別の内容に置き換えて 2 つの説明に対応づける代入戦略がある。しかしながら、SWALE の事例修正戦略はヒューリスティックを用いて行なわれているために、過去の事例をうまく現在の問題に適用できるような形に修正できない可能性が生じる場合がある。このような場合には、SWALE のように過去の事例を一般化、あるいは特殊化することによって現在の問題に適用するよりも、過去の事例と現在の問題の対応関係を調べ、現在の問題には存在しない情報を過去の事例から類推するという手法が妥当であると考えられる。また、戦略の適用は無作為に行なわれるために、最適な

戦略に到達するまで思考錯誤しなければならない。このようにすれば、事例間の類似性に着目するのみで、現在欠落している情報、すなわち現在問題となっているインパスの解消方法を発見することができる。

このような考え方に基づいて、本研究では、過去のインパス事例と現在の事例の類似性に基づいて修正案を直接写像する機構として、Gentner の提案した構造写像エンジン [2] (Structure Mapping Engine, 以降では SME と呼ぶ) を使用している。SME を用いることにより、SWALE の修正戦略を省略することができ、ヒューリスティックに依存しないメカニズムを実現することが可能になる。以下では、まず SME の概要を説明する。

3 構造写像エンジンの概要

SME は、「人間が心理学的にどのような規則によって類比や類似を解釈しているか」というテーマに焦点を当てた、構造写像理論 [6] を基にしたシミュレーションプログラムである。SME には、図 1 のように木構造で表現された基本領域 (ベース) と目標領域 (ターゲット) という記述が与えられる。SME は、ベースからターゲットへの写像を行ない、ターゲットには存在していない関係をベースから推論するものである。SME の処理過程は、局所的マッチング、大域的マッチング、推論候補の導出、評価スコアの計算の 4 つのステップからなる。以下では、この 4 つのステップを個別に説明する。

3.1 局所的マッチング

マッチング規則にしたがって、与えられた 2 つの記述からマッチング可能なすべてのアイテムの組合せを作り出す。なお、アイテムとは図 1 における木構造の各ノードのことである。局所的マッチングによって作られたマッチングの対をマッチング仮説と呼ぶ。以降では、2 つのアイテム b_i , t_j のマッチング仮説を $MH(b_i, t_j)$ と表すものと

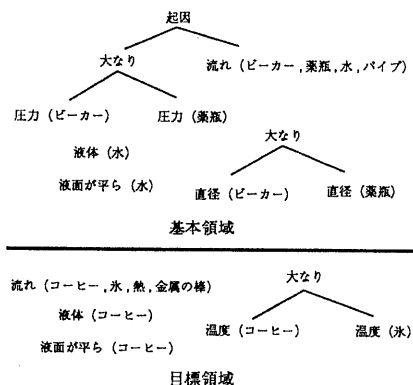


図 1: 構造の記述

する。

3.2 大域的マッチング

大域的マッチングでは、局所的マッチングによって作られたマッチング仮説を要素とする構造的に矛盾のない集合が作られる。作られた集合はマッチング集合 (Gmap) と呼ばれる。Gmap が構造的に矛盾がないとは、次の 2 つの制約を満たしている場合をいう。

- (1) 1 対 1: Gmap 内で、同一のベースアイテムが複数のターゲットアイテムとマッチングしないこと。同様に、同一のターゲットアイテムが複数のベースアイテムとマッチングしないこと。
- (2) サポート: ある Gmap 内のマッチング仮説 MH に対し、 MH のベースアイテムとターゲットアイテムの引数を対にするマッチング仮説も同一の Gmap 内に存在しなければならないこと。

3.3 推論候補の導出

大域的マッチングのステップで作られた Gmap において、集合内のベースの木構造がターゲットとマッチングされていないノードを持っているな

らば、そのノードに対応するターゲットノードの存在を推論することができる。このようにして推論されるターゲットノードを推論候補と呼ぶ。たとえば、ベース内で“起因 (A, B)”という関係があり、A と B はそれぞれターゲットの C と D にマッチングすることが分かっているならば、C と D にどんな関係も存在しない場合でも、もし両者が互いに類似しているならば、ターゲットにおいても“起因 (C, D)”という未知の関係が存在すると推論することができる。

3.4 評価スコアの計算

一般に、SME は複数の Gmap を作ることが多い。すなわち、類似の対応関係の解釈が複数存在することが多い。このため、それらの中から最も良い解釈をしているマッチング集合を選択する必要がある。SME では、それぞれの Gmap に対してスコアをつけ、最も高いスコアの Gmap を選択するという手法を採っている。評価スコアの計算は次の 2 つのステップからなる。1 つはそれぞれのマッチング仮説に重みをつけることであり、もう 1 つはその重みを計算して Gmap の全体的なスコアを計算することである。

(A) 局所的なスコアの計算

局所的なスコアの計算は、BMS [4] (Belief Maintenance System) によって行なわれる。BMS は TMS [1] (Truth Maintenance System) を改良したシステムである。従来の TMS では、信念 (Belief) と呼ばれるノードが真か偽の 2 値で表され、あるノードの値が変化すると、その値はネットワークを伝播して他のノードの値を変化させるというものであった。これに対し、BMS はノードの値とアークの値に正の実数値をとっていることに特徴がある。

(B) マッチング集合のスコアの計算

それぞれのノードの信念の値が BMS によって計算されると、Gmap 内のマッチング仮説の値が

加算され、Gmap 全体のスコアが計算される。最終的に、スコアの最も高い Gmap が最も良い写像として採用される。

SME が出力する結果を図 2 に示す。

Gmap #1	
液面が平ら (水) → 液面が平ら (コーヒー)	
液体 (水) → 液体 (コーヒー)	
水 → コーヒー	
評価スコア : 2.4445	
推論候補 : なし	
Gmap #2	
大なり (圧力 (ビーカー), 圧力 (薬瓶)) → 大なり (温度 (コーヒー), 温度 (氷))	
流れ (ビーカー, 薬瓶, 水, パイプ) → 流れ (コーヒー, 氷, 熱, 金属の棒)	
圧力 (ビーカー) → 温度 (コーヒー)	
圧力 (薬瓶) → 温度 (薬瓶)	
パイプ → 金属の棒	
ビーカー → コーヒー	
薬瓶 → 氷	
水 → 熱	
評価スコア : 5.9916	
推論候補 : 起因 (大なり (温度 (コーヒー), 温度 (氷)), 流れ (コーヒー, 氷, 熱, 金属の棒))	
Gmap #3	
大なり (直径 (ビーカー), 直径 (薬瓶)) → 大なり (温度 (コーヒー), 温度 (氷))	
直径 (ビーカー) → 温度 (コーヒー)	
直径 (薬瓶) → 温度 (薬瓶)	
ビーカー → コーヒー	
薬瓶 → 氷	
評価スコア : 3.9376	
推論候補 : なし	

図 2: SME の出力結果

図 2 は図 1 を入力とした結果である。図 1 は、「水の入ったビーカーと薬瓶がパイプでつながれていると、水はビーカーから薬瓶に流れる」という現象をベースとしており、「コーヒーと氷が金属の棒でつながれていると、熱はコーヒーから氷に流れる」という現象をターゲットとしている。また、ターゲットにおいては「コーヒーと氷の温度の違い」と、「熱の流れ」との因果関係は分かっているものとされている。一方、図 2 では、推論候補が導出されているマッピング集合は Gmap #2 であること、「コーヒーの温度が氷の温度よりも高いことが、コーヒーから氷への熱の移動を引き起こしている」という因果関係が推論されていることが示されている。また、Gmap #2 は 3 つのマッ

チング集合で最も評価スコアが高いために、最良の写像として採用されることになる。

4 理論修正システム

本研究で提案する理論修正システムは、物理現象に関する領域理論を修正するシステムである。本システムのメカニズムを図 3 に示す。

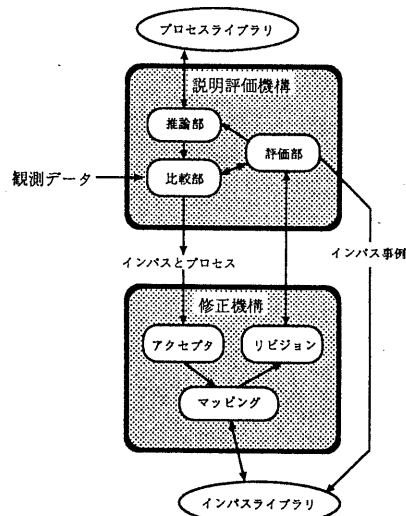


図 3: 理論修正システムのメカニズム

本システムに入力される観測データは定性的な値であり、観測データの説明には定性推論を用いている。また、定性推論を行なう機構として定性プロセスエンジン [5] (Qualitative Process Engine, 以降では QPE と呼ぶ) を用いている。

QPE では、力学の差分方程式に当たる定性モデルをプロセスと呼んでいる。プロセスにおけるパラメータ間の影響の及ぼし方は、直接影響と間接影響の 2 つに分類される。直接影響は、パラメータ値の増減の比率がプロセスによって直接決定される場合であり、間接影響は、ある変数が定性的比例関係にある別の変数の影響を受ける場合である。QPE では、物理的な系の変化はすべてこの 2 種類の影響によってのみ決定されると仮定してい

る。プロセスは以下の 5 要素によって記述される。

1. [個体:] プロセスを構成する個体名とその属性
2. [前提条件:] プロセスが適用されるべき制約条件
3. [定量条件:] プロセスの存在条件 (不等式表現)
4. [関係:] 個体の物理パラメータ間の関係
5. [作用:] プロセスによる個体の物理パラメータへの影響

図 3 において、プロセスライブラリには領域理論となるプロセスが記憶されており、インパスライブラリにはインパス事例が記憶されている。インパス事例は図 4 のように、インパスの内容、インパスの修正案、背景知識から構成されている。

インパス事例

```
インパス  impasse (ds (熱量 (コーヒー), inc), ds (熱量 (コーヒー), dec))
背景知識  I + [熱量 (コーヒー), a [heat_flow_rate]]
修正案    rep (I + [熱量 (コーヒー), a [heat_flow_rate]],
              I - [熱量 (コーヒー), a [heat_flow_rate]])
```

図 4: インパス事例

インパスは、プロセスのパラメータの値が観測データと推論結果の間で異なっている場合に発生する。このため、両者間での値の異なり方がインパスの内容として記述されている。たとえば、図 4 では“熱量 (コーヒー)”というパラメータの定性微分値が推論結果“inc” (増加) と観測データ“dec” (減少) で異なっているためにインパスが発生している。

インパスの修正案は、“rep” (置換) のようにシステムに定義されているコマンドによって表されている。この例の修正案では、“I + [熱量 (コーヒー), a[heat_flow_rate]]”という記述を“I - [熱量 (コーヒー), a[heat_flow_rate]]”という記述に置換することが示されている。ここで“I + [熱量 (コーヒー), a[heat_flow_rate]]”は、「コーヒーの熱量が

“heat_flow_rate”の割合で増加する」という関係の意味し、“I - [熱量 (コーヒー), a[heat_flow_rate]]”は「コーヒーの熱量が“heat_flow_rate”の割合で減少する」という関係を意味している。背景知識は、修正案を適用するためにプロセス内で成立しているべき関係を表している。この例では、修正案を適用するためには“I + [熱量 (コーヒー), a[heat_flow_rate]]”という関係がプロセス内に存在している必要があることが示されている。

4.1 説明評価機構

説明評価機構は、QPE による定性推論部、推論結果と観測データの比較を行なう比較部、インパスの解消度の評価を行う評価部の 3 つから構成されている。入力された観測データは、ユーザによって指定されたプロセスライブラリのプロセスによって説明される。その後、比較部で説明結果と観測データの比較が行なわれ、インパスが発生している場合は、インパスの内容が修正機構に送られる。

評価部には、修正機構によって修正されたプロセスが送られてくる。修正されたプロセスは、修正機構のアクセプタで最初に解消するように決定されたインパスがうまく解消されているかどうかを検査するために、推論部と比較部に送られる。最初に解消するように決定されたインパスがうまく解消されている場合は、評価部はそのプロセスが正しく修正されたものと判断する。インパスを解消していないプロセスは棄却される。

正しく修正されていると判断されたプロセスが複数存在する場合は、どのプロセスを最終的に採用するか決定しなければならない。そこで、評価部は、比較部において発生したインパスの数が最小であるプロセスを最終的に採用する。これは、最も多くのインパスを解消している修正案を採用することによって、インパスの効率的な解消を目指すためである。

以上のようにしても、発生するインパスの数が

最小となるプロセスが複数存在している場合は、システムはどのプロセスを採用すればよいか判断できないために、ユーザにどのプロセスを採用するか尋ねる。採用されたプロセスが新たなインパスを発生しなければ、プロセスライブラリ内のプロセスは採用された新たなプロセスに置き換えられる。プロセスが別の新たなインパスを発生している場合は、再び修正機構に処理が移される。

4.2 修正機構

説明評価機構からインパスの内容が渡されると、修正機構はそのインパスの解消を試みる。修正機構は、アクセプタ、マッピング、リビジョンの3つから構成されている。

4.2.1 アクセプタ

説明評価機構において発生したインパスとプロセスの対は、まず初めにアクセプタに渡され、理論修正を行なうための前処理が行なわれる。通常、インパスは複数発生するため、いずれのインパスから解消していくかという問題が生じる。アクセプタでは、複数発生したインパスの解消順序を決定し、さらに最初に解消すべきインパスを決定する。これがアクセプタの前処理である。

インパスの解消順序は、インパスの発生しているパラメータの種類にしたがって決定される。パラメータの種類は次の3つに分類される。

1. プロセスの直接影響によって決定されるパラメータ
2. 1. のパラメータとの定性的比例関係によって値が決定されるパラメータ
3. 2. のパラメータとの定性的比例関係によって値が決定されるパラメータ

インパスの解消順序は、1. のパラメータを最高とし、以下 2., 3. という順位で行なわれる。同一種類のパラメータが複数存在する場合は、プロセ

ス中でのパラメータの記述順序に従うものとする。たとえば、図 5 の水流プロセスにおいて、“圧力 (薬瓶)” と “圧力 (ピーカー)” というパラメータにインパスが発生している場合、両パラメータはともに 1. に属するパラメータであるが、プロセスにおける記述順序に従って “圧力 (ピーカー)”, “圧力 (薬瓶)” という順序でインパスが解消される。

```

プロセス：水流
個体
  水 1
  ピーカー
  水 2
  薬瓶
  パス
前提条件
  fluid_aligned (パス)
定数条件
  greater_than (圧力 (ピーカー), 圧力 (薬瓶))
  greater_than (量 (水 1), zero)
関係
  Q = [flow-rate, 圧力 (ピーカー) - 圧力 (薬瓶)]
  qprop [圧力 (ピーカー), 量 (水 1)]
  qprop [圧力 (薬瓶), 量 (水 2)]
作用
  1- [量 (水 1), A [flow-rate]]
  1+ [量 (水 2), A [flow-rate]]

```

図 5: 水流プロセス

定性推論では、仮に 1. のパラメータにインパスが発生した場合、このパラメータと定性的比例関係にある 2. のパラメータにもインパスが伝播するため、1. のパラメータのインパスを初めに解消すれば、2. のパラメータのインパスは自然に解消されることがわかる。このように、インパスの解消順序をつけることによって、解消すべきインパスの数を減少させることができる。

4.2.2 マッピング

マッピングでは、図 6 のように、SME を用いて現在の事例とインパスライブラリ内のインパス事例の写像が行なわれる。写像は、インパス事例をベースとして、また、現在の事例をターゲットとして行なわれる。ベースとターゲットで、インパスの内容と背景知識の2つとも写像に成功した場合、

推論候補という形で修正案が導出される。本システムがSWALEにおける修正戦略ではなくSMEを使用した最大の理由は、インパスの内容と背景知識に関して事例間の類似性を直接検出し、それに基づいて修正案を導出することにある。これによって、修正戦略のヒューリスティックを排除することができる。

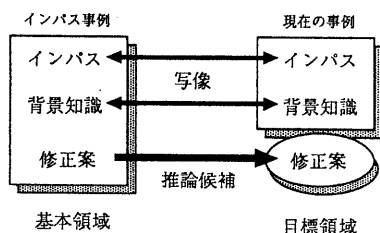


図 6: 事例の写像

4.2.3 リビジョン

リビジョンでは、提案された修正案をプロセスに適用して実際に理論の修正が行なわれる。修正案は、システムに定義されているコマンドによって表現されている。現在、本システムに定義されているコマンドには、rm (削除), add (加入), rep (置換) の3つがある。マッピングによって提案された修正案が複数存在する場合は、修正案をそれぞれ適用した複数のプロセスが作り出される。

リビジョンで修正されたプロセスは、説明評価機構の評価部に渡され、プロセスの評価が行なわれる。評価部ですべてのプロセスが棄却された場合、もしくはマッピングにおいてすべてのインパス事例との写像に失敗して修正案が全く得られない場合は、ユーザにプロセスの修正を求める。ユーザは、前述したシステム定義コマンドを使って理論の修正を行なう。その後、ユーザの提案した修正案は、実際にインパスを解消するか説明評価機構で検査される。最終的に、ユーザの修正案が実際にインパスを解消していると判断された場合、ユー

ザの修正案は新たなインパス事例としてインパスライブラリに貯蓄される。

5 理論修正の例

理論修正の具体的な例として、電池につながったコンデンサに電荷が溜る現象を考える。システムに入力される観測データを、図7に示す。

```
observation ( [ds (量 (電荷 1), std),
               ds (量 (電荷 2), inc),
               ds (電圧 (電池), std),
               ds (電圧 (コンデンサ), inc),
               greater_than (電圧 (電池), 電圧 (コンデンサ)),
               greater_than (量 (電荷 1)), zero) ],
              [ds (量 (電荷 1), std),
               ds (量 (電荷 2), std),
               ds (電圧 (電池), std),
               ds (電圧 (コンデンサ), std),
               equal_to (電圧 (電池), 電圧 (コンデンサ)),
               greater_than (量 (電荷 1), zero) ] )
```

図 7: 観測データ

“observation(list-1, list-2)” は “list-1” から “list-2” へ状態変化することを表している。また、“list-1” における “ds(量 (電荷 1), std)” は、“量 (電荷 1)” というパラメータの定性微分値が 0 であることを表している。この観測データの “list-1” は、「コンデンサの電圧より電池の電圧が高い状態で、コンデンサの電荷の量と電圧が増加し、電池の電荷の量と電圧は一定である」ことを表している。また、“list-2” では、「コンデンサの電圧が電池の電圧と等しくなったために、すべてのパラメータの値の変化が停止した」ことを表している。図7の観測データの説明には、図8の帯電プロセスが使われるものとする。

システムに観測データが与えられると、説明評価機構の推論部が図8の帯電プロセスを使って定性推論を行なう。推論された結果は図9のようになる。推論結果は “envisioning(list-a, list-b)” という表現形式で表されており、これは、“list-a” から “list-b” へ状態変化することを表している。“list-a” は「電池の電圧がコンデンサの電圧より高い状態で、コンデンサの電荷の量と電圧が増加し、電

```

プロセス：帯電
個体
  電池
  電荷 1
  コンデンサ
  電荷 2
  導線
定置条件
  greater_than (電圧 (電池), 電圧 (コンデンサ))
  greater_than (量 (電荷 1), zero)
関係
  Q = [flow-rate, 電圧 (電池) - 電圧 (コンデンサ)]
  qprop [電圧 (電池), 量 (電荷 1)]
  qprop [電圧 (コンデンサ), 量 (電荷 2)]
作用
  I- [量 (電荷 1), A [flow-rate]]
  I+ [量 (電荷 2), A [flow-rate]]

```

図 8: 帯電プロセス

池の電荷の量と電圧が減少すること」を表している。また、「list-b」は「コンデンサの電圧が電池の電圧と等しくなったために、すべてのパラメータの値の変化が停止した」ことを表している。

```

envisioning ( [ds (量 (電荷 1), dec),
               ds (量 (電荷 2), inc),
               ds (電圧 (電池), dec),
               ds (電圧 (コンデンサ), inc),
               greater_than (電圧 (電池), 電圧 (コンデンサ)),
               greater_than (量 (電荷 1), zero)],
              [ds (量 (電荷 1), std),
               ds (量 (電荷 2), std),
               ds (電圧 (電池), std),
               ds (電圧 (コンデンサ), std),
               equal_to (電圧 (電池), 電圧 (コンデンサ)),
               greater_than (量 (電荷 1), zero)])

```

図 9: 推論結果

次に、説明評価機構の比較部によって、「list-1」と「list-a」、「list-2」と「list-b」がそれぞれ比較される。この場合、「量 (電荷 1)」と「電圧 (電池)」の 2 つのパラメータの定性微分値にインパスが発生していることが分かる。この結果、2 つのインパスは、「impasse(ds(量 (電荷 1), dec), ds(量 (電荷 1), std))」、「impasse(ds(電圧 (電池), dec), ds(電圧 (電池), std))」と表現され、修正機構のアクセプタに送られる。

アクセプタでは、2 つのインパスの解消順序が決

定される。「量 (電荷 1)」というパラメータは、帯電プロセスにおいて直接影響のパラメータであるのに対し、「電圧 (電池)」というパラメータは、間接影響のパラメータである。このため、「電圧 (電池)」よりも「量 (電荷 1)」におけるインパスの解消順位が高くなる。したがって、「impasse(ds(量 (電荷 1), dec), ds(量 (電荷 1), std))」を最初に解消することが決定され、このインパスがマッピングに送られる。

マッピングでは、図 10 のように現在の事例とインパス事例との写像が行なわれ、修正案「rm(I-[量 (電荷 1), A[flow-rate]])」が推論される。これは、「熱源の熱量が、「heat.flow.rate」の割合で減少する」ということを表している。

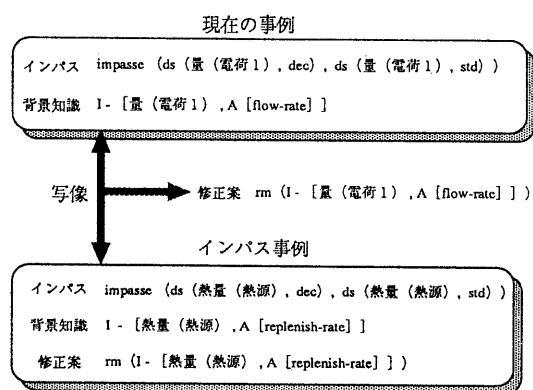


図 10: 事例の写像

リビジョンでは、マッピングで得られた修正案を実際にプロセスに適用し、帯電プロセスから「I-[量 (電荷 1), A[flow-rate]]」という作用を削除する。修正された帯電プロセスは、説明評価機構の評価部に送られて評価される。推論部で再び定性推論を行なった結果は図 11 のようになり、「impasse(ds(量 (電荷 1), dec), ds(量 (電荷 1), std))」というインパスが解消されていることが分かる。また、「impasse(ds(電圧 (電池), dec), ds(電圧 (電池), std))」というインパスも同時に解消されてい

るため、評価部はこのプロセスを正しく修正されたプロセスとして新たにプロセスライブラリに記憶する。

```
envisioning ( [ds (量 (電荷 1), std),
               ds (量 (電荷 2), inc),
               ds (電圧 (電池), std),
               ds (電圧 (コンデンサ), inc),
               greater_than (電圧 (電池), 電圧 (コンデンサ)),
               greater_than (量 (電荷 1)), zero) ],
              [ds (量 (電荷 1), std),
               ds (量 (電荷 2), std),
               ds (電圧 (電池), std),
               ds (電圧 (コンデンサ), std),
               equal_to (電圧 (電池), 電圧 (コンデンサ)),
               greater_than (量 (電荷 1), zero) ] )
```

図 11: 修正されたプロセスによる推論結果

6 考察

本研究では、事例を利用した理論修正システムのメカニズムを提案した。本システムは類推機構に基づいた手法を用いている。類推機構を使用する利点は、不完全な領域知識に対し、他の領域知識を利用することで不完全性を補うことができる点にある。すなわち、類似した別の領域知識から情報を得ることができるため、1つの限定された領域内で推論を行うよりも柔軟性があると考えられる。

本研究で提案したシステムの問題点としては、次の2点が挙げられる。

1. 複数の修正案を絞り込む方法が不十分である。
2. 物理現象を複数のプロセスで説明する場合、アクセプタによる矛盾解消順序の決定ができない。

問題 1. の解決方法として、本研究では、複数の修正案を1つに絞り込むための明確な指標が存在しないために、どちらの修正案を採用するかユーザに求めるという手法を採っている。逆に、修正案を絞り込む方法をユーザに提案するという手法も考えられる。この手法を実現するためには、1章で説明した Rajamoney の ADEPT の手法が適用

できると考えられる。すなわち、複数の修正案が提案された場合、実験構築のための補助理論を用意しておき、候補を1つに絞り込むための実験方法を提案するようなシステムの構築が考えられる。

また、問題 2. は、インパスの発生したパラメータが1つのプロセス内では直接影響のパラメータとして記述されているのに対し、別のプロセスでは間接影響のパラメータとして記述されている場合に、アクセプタはインパスの解消順序を決定することができないために生じている。この問題に対しては、インパスの発生がどのプロセスによる記述の誤りによるものであるか発見するような手法の開発が必要であると思われる。

参考文献

- [1] Doyle, J.: A Truth Maintenance System, Artificial Intelligence, Vol.12, pp.231-272 (1979).
- [2] Falkenhainer, B., Forbus, K. D., and Gentner, D.: The Structure-Mapping Engine: Algorithm and Examples, Technical Report UIUCDCS-R-87-1361, University of Illinois at Urbana-Champaign, pp.1-63 (1987).
- [3] Falkenhainer, B.: Learning from Physical Analogies: A Study in the Analogy and the Explanation Process, Doctoral dissertation, Department of Computer Science, University of Illinois at Urbana Champaign (1989).
- [4] Falkenhainer, B.: Towards A General-Purpose Belief Maintenance System, Uncertainty in Artificial Intelligence 2, pp.125-131 (1988).
- [5] Forbus, K. D.: Qualitative Process Theory, Artificial Intelligence, Vol.24, pp.85-168 (1984).
- [6] Gentner, D.: Structure-Mapping: A Theoretical Framework for Analogy, Cognitive Science, Vol.7, No.2, pp.155-170 (1983).
- [7] Kass, A.: Adaptation-Based Explanation: Explanation as Cases, Proc. of the Sixth International Workshop on Machine Learning, pp.49-51 (1989).
- [8] Rajamoney, S.: Experimentation-Based Theory Revision, Proc. of the Seventh AAAI, pp.7-11 (1988).
- [9] Schank, R.: Explanation Patterns: Understanding Mechanically and Creatively, Lawrence Erlbaum Associates (1986).