

## シンセシスの推論フレームワークに関する研究(第2報)

23-05

- モデルベース・アブダクション -

Research on Reasoning Framework for Synthesis (The Second Report)

- Model-based Abduction -

吉岡 真治

Masaharu Yoshioka  
yoshioka@nii.ac.jp

武田 英明

Hideaki Takeda  
takeda@nii.ac.jp

富山 哲男

Tetsuo Tomiyama  
tomiyama@race.u-tokyo.ac.jp

国立情報学研究所

National Institute of Informatics

東京大学人工物工学研究センター

RACE, The University of Tokyo

Abstract: Synthesis plays a major role in design and abduction is regarded a critical element within a creative synthesis process. While there are many research efforts for formalizing abduction within design, most of them can only deal with limited problems and highly depend on designers' creative thinking. In this paper, based on the results of previous reports, we propose model-based abduction that is a new reasoning method within a proposed framework for reasoning systems for synthesis. We also propose an algorithm of this model-based abduction and evaluate it by comparing design protocol data.

## 1. はじめに

本研究では、シンセシスを解明し理解し、それを説明するモデルの構築を構築し、それを推論フレームワークとして実現するための研究を行っている。そのために、前報[1]では、この推論フレームワークに必要なとされる基本的な要件について考察し、その要件をみたくような推論フレームワークの提案を行った。

本稿では、シンセシス過程において非常に重要な役割を果たすと考えられる設計解を想起する過程であるアブダクションについての議論を行い、提案した推論フレームワークに基づくモデル化を試みる。

## 2. 設計におけるアブダクション

## 2. 1. アブダクション

前報で述べたように、設計のようなシンセシスが重要な役割を果たす思考は、論理的に捉えるとアブダクションと考えられる。設計におけるアブダクションは、次式において、定理 (Th) の一部と知識 (K) を与えて、定義事象 (Fd) を求める行為として定式化される。

$$K \cup Fd \vdash Th \quad (1)$$

このアブダクションは、知識が多ければ多いほど定義事象をより求めやすくなるという特徴を持つ。

ところで、アブダクションは Peirce によって提唱された [2] わけであるが、人工知能分野では精力的にその推論方法が研究されてきた。アブダクションを計算機アルゴリズムとして実現するための方針としては、次のようなものが考えられる。

- 純粋に論理的な演算として実現

式(1)の枠組みにおいて、

$$K = \{p \rightarrow q, p \rightarrow r\}, Th = \{q, r\}$$

が与えられたときに、後ろ向き推論を利用して

$$Fd = \{p\}$$

を導出する方策である。

例えば武田らの設計シミュレータにおいてもアブダクションは、このようにして実現されていた [3]。

- アナリシスの利用

解空間がある程度、与えられていて解候補の生成法がわかっている場合には、生成検査法が利用できる。解候補を生成した後に、アナリシスによって解候補を検査し、合格したものだけを解として採用する。したがって、アブダクションは解のランダムな生成に対応し、アルゴリズムそのものは簡単であるが、効率的でないことは当然である。類似の方法として遺伝子アルゴリズム、遺伝子プログラミングなどがある。これらは、生成検査法における最適手法ということができる。

- 手続き的知識の利用

解を生成するための手段として、手続き的知識などが利用可能な場合には、アブダクションはその知識を利用することで実現できる。例えば、解空間が極めて限定的で、考慮しなければならない問題点がリストアップできるような場合には、カタログ検索 (データベース検索) 問題に帰着する。また、もっと単純に、解の性質と要求との関係が計算式のような形で明に与えられる場合には、逆関数を計算することに帰着する。また、やや異なるアプローチとしてはケースベース推論 (CBR) がある。これは典型解を用意しておき、問題ごとに軽微な修正を加えて適用していくというやり方である。

これらの手法はいずれも、問題が予めうまく整理できる場合には有効かつ効率の良い方法である。しかし、そうでない場合には全く無力であるという欠点が存在する。

- 2. 2. 設計の文脈の利用

前節で述べた、従来から行われているアブダクションの定式化では、クローズドワールドアサンプション (真偽の判断のつかない事実是否定として扱う) が多く用いられて

おり、その結果として、解の数が制限されるという事が多く見られた。この仮説はルーチン設計のような、解のバリエーションが限定されているような設計には良い方法であるが、創造的な設計には適していない。

人間が行う真に創造的なアブダクションを限られた知識とアルゴリズムで実現するのは容易な問題ではなく、近似としてのみ実現可能な問題である。しかし、近時ではあるにせよ、人間が行っているアブダクションに、より近い近似としての推論方法を考案することが必要である。

実際に設計者が行っている設計活動を考えると、設計を進めて、解のイメージが固まってくると同時に必要な知識が明らかになっていくといった過程が観察される。

よって、創造的な設計を支援するためには、問題領域を限定した単一の知識ベースを利用するのではなく、様々な知識ベースを柔軟に組み合わせて利用する必要がある。しかし、この様なアブダクションを単純に計算しようとした場合にはその計算量が NP 困難な問題であり[4]、これを回避するためには、設計問題という文脈を利用する必要がある。

そこで、設計におけるアブダクションの役割について考察すると、その多くは機能や属性によって与えられる仕様から、それを実現する実体やより具体的な仕様を導く操作である。よって、設計におけるアブダクションを考えるにあたって、機能、属性、実体と言った設計に特有の事実の分類をセマンティクスとして用いることにより、その推論の方向性を制御する、semantics constrained abduction が考えられる。この推論の方向性の制御により、単純なアブダクションの計算量の問題を避けることが出来る。

これに対し、実際の設計を分析するために、模擬的な設計を行い、その過程を分析する設計実験が行われている。この結果から、設計者が新しい設計解を考える場合に、目の前にある実体や、今までに使ったことのある実体を参照している事が多い事が分っている[5]。

また、設計実験中で用いられた知識の分析からは、機能と属性を直接結びつけるような知識は利用されず、実体を媒介として機能と属性の関係性を述べる知識が多く利用されている事が示されている（表 1）[6,7]。

表 1 設計実験データにおける設計知識の分類と利用回数

| 設計知識の分類      | 利用回数 |
|--------------|------|
| 機能→実体        | 30   |
| 実体→機能        | 8    |
| 属性→実体        | 1    |
| 実体→属性        | 38   |
| 属性→属性        | 4    |
| 位相的接続関係→接続方法 | 24   |
| 実体→製造方法      | 9    |
| 製造方法→属性      | 17   |

そこで、この設計における実体概念の役割に注目して、アブダクションの定式化を行う。まず、最初に設計知識として、次の 2 種類の知識を想定する(e:実体、f:機能、p:属性)。

- 機能と実体の関係を表す知識  $e \rightarrow f$
- 属性と実体の関係を表す知識  $e \rightarrow p$

これらの知識は、機能として与えられる設計仕様から実体を導くアブダクションの知識として用いられるだけでなく、設計解である実体からその実体をもつ機能や属性を導く演繹のための知識としても用いられる。しかし、これら

の知識はこれらの知識の中から適当に選んで使えば良いのではなく、与えられた問題に取ってふさわしい知識を選択して使う必要がある。

この様な設計における実体概念の役割に注目すると、設計過程では、既存の実体とこれから作ろうとしている設計解を表現した仮想的な実体を利用していることが指摘されている[5]。我々は、この様な仮想的な実体を扱う枠組として「抽象的な実体概念」を提案している。この抽象的な実体概念は、オブジェクト指向言語のクラス定義や Minsky のフレームに似た概念で、その実体概念を特徴付ける有限個の属性により定義されるものである[8]。

一般設計学[9]で用いられる位相空間で考えると、抽象的な実体概念  $e$  は、属性空間における属性概念の積集合  $T$  に対する代表元

$$e \in \cap T_i \quad (e \in T_0 : \text{属性概念位相})$$

として捉えることが出来る。これに対して通常の実体概念は実体概念集合の元であることに注意する必要がある。

例えば、「机」という抽象的な実体概念は天板と足という属性から定義される物であり、実際に存在する多くの机の代表である。この抽象的な実体概念は、属性に関する情報を増やすことにより、より具体的な実体概念に対応付けることが出来る。例えば、天板の形を円形にすると「丸机」というより具体化された抽象的な実体概念になる(図 1)。

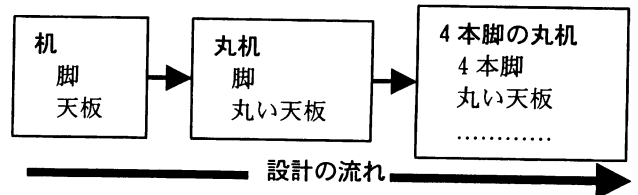


図 1 抽象的な実体概念の進化

この抽象的な実体概念を用いて、考慮する属性概念を有限個に制限することにより、設計者は考えを発散させずに設計を進めることが出来る。また、抽象的な実体概念の詳細化（すなわち、抽象的な実体概念を記述する属性概念の積集合の濃度を増加すること）や、実際の実体概念を参照（これは ZF 集合系においては選択公理によって保証されている）することにより、設計者は設計解をより具体化することが出来る。

### 3. モデルベース・アブダクション

抽象的な実体概念は有限個の属性により表現できるので、これらの実体概念はある特定の視点に基づくモデルベース推論システム上に作成されたモデルとして表現することが可能である。例えば、幾何モデルは形状などの幾何的な属性情報を扱うことが出来る。そこで、この様なモデルベース推論システムにおけるモデルへの操作としてアブダクションを定式化する。また、アブダクションのようなシンセシス過程においては、設計知識の数が多ければ多いほうが良いので、この定式化では、複数のモデルベース推論システムに対する操作とする。すなわち、このモデルベース・アブダクションは前報[1]で提案したシンセシスの推論フレームワークのモデルベース推論システムを制御するための思考過程モデルの一部に対応する。

#### 3. 1. 基本的なアルゴリズム

まず、最初にモデルベース・アブダクションのアルゴリズムの概要を述べる。このアルゴリズムでは、複数の知識ベースの利用方法と実体、機能、属性といった概念分類の

利用法に注目する。

1. ある知識  $K_1$  を用いて設計解の近傍系を導出する
  - a) 設計仕様として与えられた要求  $Th$  のうち、ある知識  $K_1$  で扱う事が出来る要求( $Th_1$ )を設定する。
  - b) 解候補集合  $Fd_1 = \{e_1, e_2, \dots\}$  をクローズドワールドアサンプションを用いたアブダクションにより導出する。  
 $K_1 \cup Fd_1 \vdash Th_1$   
 $\{e_1 \rightarrow p_1, e_1 \rightarrow p_2, \dots, e_2 \rightarrow f_1, \dots\} \cup Fd_1 \vdash \{p_1, \dots, f_1, \dots\}$   
 ここで  $e_i$  は実体概念で、 $p_i$  は属性概念、 $f_i$  は機能概念である。

2. 各解候補について抽象的な実体概念の考え方にに基づき近傍系を調べ、解候補の存在可能領域  $Fd_1'$  を導出する。また、この近傍系に対して公理系を用いて演繹を行い、属性や機能の情報を導出し、設計解が持つ属性や機能に関する情報  $Th_1$  を増やす。
3. 1.の操作を異なる知識  $K_i$  に対して行い、同じく解候補の存在可能領域  $Fd_i'$  を導出する。

$\bigcup_{\lambda \in \Lambda} Th_\lambda \supset Th$  を満たす添数集合  $\Lambda$  に対し、 $\bigcap_{\lambda \in \Lambda} Fd_\lambda'$  を計算することにより解候補を絞り込む。

以上の操作では、semantics constrained abduction を行う 1-b) の操作が核心である。すなわち、ある知識  $K_i$  を選択するところで、特定のルール群にしか着目すると言う点と、機能または属性と実体を関連付ける知識のみを用いると言う点で semantics にガイドされた推論を行う。

### 3. 2. シンセシスの推論フレームワークによる実現

また、前報で述べたシンセシスの推論フレームワークに対する操作として上記の一連の操作を考えると、各推論システムで表現されている抽象実体概念に相当するモデルを作成し、各推論システムが持っている知識に応じて行われたアブダクションの結果として属性などの情報を増やすという操作に相当する。

そのためには、各々の推論システムが持つ知識  $K_i$  を

$$K_i = \{e_1 \rightarrow p_1, e_1 \rightarrow p_2, \dots, e_2 \rightarrow f_1, e_2 \rightarrow f_2, \dots\}$$

の形式に予めアレンジしておく必要がある。これは、例えばある実体概念の持つ属性や機能を整理してルール化しておくのであれば容易に作成できる。このルール化の操作は先に述べた抽象的な実体概念を知識ベースとして記述していることに相当する。つまり、このアルゴリズムは、アブダクションが可能な知識ベースの整理法をも規定していることになる。

しかし、ソリッドモデラのようにモデルの入力が完全に設計者任せになっている場合や、数値計算による最適化が出来るようなシステムでは、特定の操作の結果、どのような入力に対してどのような出力が得られるかを予め知識  $K_i$  として論理世界側に用意しておく必要がある。つまり、ある操作によって、抽象実体概念がどのような変化を受けるかのかを予め調べ、抽象実体概念間の関係として対応表にしておく必要がある。例えば、幾何モデルでは、この操作によって、この実体にこのような属性を付加できるというような形状フィーチャレベルでの知識を整理して、 $K_i$  として蓄積していく必要がある。

すなわち、この様な知識を持った各々のモデルベース推論システムは、その目的に応じた設計知識ベースと考える

ことが出来る。ただし、これらの知識ベースは、その目的に応じて機能概念や属性概念の一部のみが扱えるものであり、これらのシステムを組み合わせる必要がある。その統合的な視点にたった知識は前報[1]の論理推論機構によって取り扱われる。

前報[1]で述べたように、本研究では、シンセシスの推論フレームワークをこれまでに作成している知識集約型工学環境 KIEF (Knowledge Intensive Engineering Framework) [10] を用いて実現することを検討している。そこで、本報では、前節で述べたアルゴリズムを KIEF に対する操作として実現し、その妥当性の検証を試みる。

この KIEF では、プラグブル・メタモデル機構と呼ばれる推論機構により、複数のモデルベース推論システムの統合を実現される。このプラグブル・メタモデル機構では、各々のモデルベース推論システムに関する知識として、どのような情報を入力として扱い、結果としてどのような情報を出力として返すことが出来るかが記述される。また、各々のモデルベース推論システムにおけるモデルや推論結果は、その関係をネットワークモデルとして表現したメタモデルと呼ばれるモデルによって関係付けられる。このメタモデルは等価な論理式に変換することが可能であり、論理推論機構のための論理モデルとして機能することが出来る。

この KIEF を用いて、モデルベース・アブダクションを実現するために、前節で述べたアルゴリズムをこの KIEF に対する操作としてのアルゴリズムへと詳細化する。

前節のアルゴリズムとの対応関係を考えると、1 番目の手順である複数の知識の利用による設計解の導出や 2 番目の手順である設計解が持つ属性や機能の導出が各々のモデルベース推論システムにおける操作に対応し、1 番目の手順や 2 番目の手順で行われるモデルベース推論システムの選択や、3 番目の手順である結果の取りまとめと解の絞込みが、プラグブル・メタモデル機構によるモデルの統合やメタモデルを用いた論理推論機構に対する操作に相当する。これらの操作を具体化すると次のようになる。

1. 要求である  $Th$  をメタモデルに設定する。
2. モデルベース推論システムに関する知識を用いて、要求  $Th$  と関係のあるモデルベース推論システムを導出し、その中から 1 つのシステムを知識ベースとして選択する。利用可能な新しい推論システムが存在しない場合は終了する。
3. 選択したシステム上でモデルを作成する。
  - a) 設計者は選択したシステムを用いて要求をモデル化する。システムが与えられた仕様からそれを実現する実体を導出することが可能な場合には、仕様の入力に相当する。
  - b) 入力された仕様から、それを満たす設計解をモデルとして作成する。このモデルは、抽象的な実体概念に相当するものである。選択したシステムが持つ知識ベースにより、カタログ検索や数値計算による支援などが可能な場合は、それを利用する。
4. 作成したモデルを用いて、メタモデルを詳細化する。この新しい設計解に対して、関連する他の推論システムを論理推論機構により導出し、これらのシステムを用いることにより設計解を分析する。この操作は演繹的な操作であり、その結果として、設計解の属性や機能に関する情報が詳細化される。

5. 導かれた属性や機能に関する情報を要求と比較し確認する。

- a) 問題がなければ、更なる検討を進めるために、手順 2 に戻る。
- b) 問題がある場合は次のどちらかを行う。
  - i) 手順 3-b に戻り、新たな設計解を導出する。
  - ii) 知識が不完全なために問題が生じる場合があるので、問題の原因となっている知識を修正する。

### 3. 3. 例題による検証

この提案したアルゴリズムの妥当性の検証を試みるために、実際の設計実験のプロトコルデータ (図 2) とそれに対応する KIEF に対する操作を比較する。

1. 普通の体重計はどうなっているのか?
2. こうやって変位を測るのだろう(図 A)。
3. ラックとピニオンでこうすれば(Figure B), 重さと変位が比例するのを利用して測れる
4. とりあえずその前に表示を考えよう。
5. 5mm の変位を 100kg にするのだから 1kg 当たりの変位は 0.05mm だ。
6. こんなのはこれ(図 B)では無理だ。
7. 精度はともかくとして、ギアを何段もかませれば出来る。

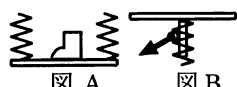


図 2 体重計設計のプロトコル

現在、KIEF システムには、機能設計支援ツール FBS モデラ [11]、因果関係の知識に基づく物理現象推論システム、定性物理に基づく挙動シミュレーションシステム、ギア機構の解析システム、ギア機構の標準部品ライブラリが接続されている。

また、従来の FBS モデラでは、機能をサブ機能に分解し、挙動と対応付けるといった知識に基づく設計支援が行われていたが、ここでは、抽象的な実体概念に対応する知識として機能と実体の関係を表現した FBS モデルのライブラリを知識として利用するように修正を行った。ユーザは従来の FBS モデラの知識を直接利用するのではなく、このライブラリに基づいて、FBS モデルの検索、修正を行う。また、このライブラリは実体概念の抽象具体関係に応じて、対応する FBS モデルの検索も可能とした。

ここでは、各々の発話について、システムによる実行結果と前述のアルゴリズムの手順について対応関係を示す。

初期設定: 初期仕様として機能概念「重さを量る」と「100kg を 5mm の変位に変換」をメタモデルに設定する (手順 1)。

発話 1 と 2: 機能概念を扱うシステムとして FBS モデラが選択され、「重さを量る」という機能概念をキーとしてライブラリの検索を行う (手順 2)。結果として、「普通の体重計」が検索された (手順 3)。この段階のモデルは、抽象度が高いため、他の知識ベースを用いて新たな知見が得られなかった (手順 4 と 5)。

発話 3 と 4: 「普通の体重計」の一部分である「表示部」について FBS モデラを利用し、概念間の階層関係に基づき、「表示部」を具体化させた「ラック」と「ピニオン」と「表示板」から構成される「回転表示板」を選択する (手順 2)。

発話 5 と 6: 「ラック」と「ピニオン」の「ギア伝達」につ

いては、ギア機構解析システムが解析可能であるので、それを用いて解析を行う (手順 4)。解析のために、「ラック」と「ピニオン」のモジュールなどの定量情報が必要となりギア機構の標準部品ライブラリを検索し (手順 2)、設定する (手順 3)。この解析結果から 5mm の変位ではほとんどピニオンが回転しないことが分り、「100kg を 5mm の変位に変換」と矛盾するとユーザが判断した (手順 5)。

発話 7: まず、「回転表示板」による知識を「減速機構」と「回転表示板」の組合せという知識に修正し (手順 5-b-ii)、原則機構について同様に FBS モデラのライブラリを検索し、ギア減速機構を導出する (手順 3-b)。このギア機構についてギア機構解析システムで解析した結果制度に関する情報を得た (手順 4)。

この結果より、本アルゴリズムを用いる事により複数の知識ベースの使い分けなどが支援できることが確認された。問題点としては、結果の解釈や知識の修正などに人間による判断結果が含まれる形になっており、この様な点についてさらに検討を進めていく必要がある。

### 4. まとめ

本稿では、設計解を想起する過程であるアブダクションについて、設計のセマンティックスを用いる事により、モデルベース・アブダクションとして定式化を行った。また、我々がこれまでに作成している KIEF システムを用いることによりこのアルゴリズムを実現し、例題により、その有効性を検討した。

なお、本研究の一部は、日本学術振興会未来開拓学術研究推進事業「シンセシスのモデル論」プロジェクト (JSPS-RFTF 96P00701) の研究費によって実施された。

### 参考文献

- [1] 武田英明, 吉岡真治, 富山哲男. 「シンセシスの推論フレームワークに関する研究 (第 1 報) - シンセシスのモデル化 -」, 人工知能学会全国大会 (第 14 回) 論文集.
- [2] 井上 克巳. 「アブダクションの原理」, 人工知能学会誌, Vol. 7, No. 1, pp. 48-59, 1992
- [3] 武田英明, 富山 哲男, 吉川 弘之. 「設計過程の計算可能モデルと設計シミュレーション」, 人工知能学会誌, Vol. 7, No. 5, pp. 877-887, 1992.
- [4] T. Bylander, D. Allemang, M. C. Tanner, and J. R. Josephson. "The computational complexity of abduction," *Artificial Intelligence*, Vol. 49, pp. 25-60, 1991.
- [5] 吉川弘之, 荒井栄司, 後藤敏彦. 「実験設計学」, 精密機械, Vol.47, No.7, pp. 830-835, 1981.
- [6] 武田英明, 富山哲男, 吉川弘之. 「実験的手法による設計知識とその利用に関する分析」, 精密工学会, Vol. 60 No. 3, pp.442-426, 1994
- [7] T. Tomiyama. "From General Design Theory to Knowledge-Intensive Engineering," *Artificial Intelligence for Engineering Design, Analysis and Manufacturing*, Vol. 8, No. 4, pp. 319-333, 1994.
- [8] 吉岡真治. 「設計知識操作論」, 第 14 回設計シンポジウム公演論文集, pp.40-47, 1996.
- [9] 吉川弘之. 「一般設計学序説 一般設計学のための公理的方法」, 精密機械, Vol.45, No.8, pp. 906-912, 1979.
- [10] 関谷貴之, 吉岡真治, 富山哲男. 「オントロジーを用いた統合的設計支援環境の実現」, 人工知能学会誌 Vol. 14, No. 6, 1999
- [11] 梅田靖, 富山哲男, 吉川弘之. 「機能設計支援のための FBS モデリングの研究」, 精密工学会誌, Vol.63, No.6, pp. 795-800, 1997.