

知的CAD開発のための設計過程の論理による定式化⁴⁻⁷

Logical Formalization of Design Processes for Intelligent CAD

武田英明 冨山哲男 吉川弘之

Hideaki TAKEDA, Tetsuo TOMIYAMA, Hiroyuki YOSHIKAWA

東京大学工学部精密機械工学科

Dept. of Precision Machinery Eng., Faculty of Eng., Univ. of Tokyo

Formalization of design processes is needed to develop intelligent CAD systems. We use two methods to formalize design processes in this paper.

First, Design processes are analyzed experimentally with the protocol analysis method. We pick up transitions of object descriptions and transitions of designers' viewpoints from the protocol data.

Second, we formalize design processes based on modal logic and non-monotonic logic. Modality is used to represent alternative design solutions and nonmonotony is used to represent "trial and error" in design processes. We translated the protocol data into this logical framework and show that it is appropriate to represent design processes.

1 はじめに

機械設計の分野におけるCADは幾何形状の計算機による処理の実現という点ではかなりの成功を納めた。しかし製図作業は設計の最終段階である詳細設計の一部であり、それより前に概念設計や基本設計が行われている。これらの段階の計算機による支援は現在のところ、あまり行われていない。

この理由は設計には、現実世界（物理世界）でものを作る行為であると同時に、人間の思考行為であるという二つの側面があることによると考えられる。すなわち従来のCADは前者の一部に注目し、計算機内に物理世界を表現する幾何モデルというものを用意することで設計支援を行った。しかし、このアプローチではこれより上流の設計段階への適用は難しく、インテリジェントCAD [1,2]の開発に結び付かない。というのは概念設計といった段階では、人間の思考行為という設計のもう一つの側面が強く現れるからである。したがって、概念設計等のより上流の設計を支援するためには思考行為としての設計にも注目して、これを計算機上に実現する方法を求めていく必要がある。この研究がなされてはじめて設計を記述する言語を開発することができ、設計過

程全般を支援するインテリジェントCADが合理的に構築できるようになる。

本研究では思考としての側面を中心として設計過程に注目し、設計過程を形式的に表現する方法を求めることを目的とする。

この思考としての設計を知るために、まず実験的手法を用いて設計を観察する。そしてその結果を論理によって分析する。具体的には、設計実験を行うと共に論理を用いた設計過程の形式的な表現法を作り、これをプロトコル解析のデータに適用することで実験的に表現の有効性を確かめ、形式化をより発展させる。

2 設計実験

思考行為としての設計過程を知る方法として、実際の設計を観察する設計実験を行い、設計過程の特徴・特質を明らかにする。

設計実験とは模擬的に設計を行わせて、その設計行為をなんらかの方法で観察することである [3]。設計実験の方法としては行為の観察、回想による記録、プロトコル解析等が考えられる。ここではこのうちプロトコル解析を用いる。設計作業のプロトコル解

析は既にいくつか行われており、この結果から分析的考察を導いている [4]。

プロトコル解析とは、被験者に声を出しながら設計を行ってもらい、その発話を記録するというものであり、心理学実験等で用いられる手法である。プロトコル解析から得られる情報は、人間の短期記憶の情報を示しているといわれている [5]。

今回行った実験の課題は自動販売機の搬出部の設計であり、被験者は5人の学生と1人の技術者であった。実験は同一課題で二人一組で3組行った。実験の記録は発話と図の指示等の動作をVTRで記録し、図等は一定間隔でコピーして記録に残した。

図1は得られたプロトコルの一部である。ここでは押しだし機構部の機構について議論している。

- ・問題はどのように取り出すか？
- ・一つ一つ取り出す必要はない。
- ・フォークリフトのようなもので押し出すか？
- ・収納部の課題の図のようであれば、一つづつ押し出せばよい。
- ・落とすだけでよいならば、それでよいか？
- ・落とすか壁で運び出すならば、押し出してからベルトコンベア等で運ばばよい。
- ・課題の図の収納部では図A(a)のように押し出せない。
- ・図A(b)のように下から出せばよい。
- ・図A(b)のdは1個より大きければよい。
- ・今度は押し出す機構を考えよう。
- ・ゴム・ベルトを用いる。
- ・ゴムの摩擦を利用してよいし、図Bのように舌下の突起をつけるのもよい。
- ・回るスピードによっては放がうまく敵らないのではないかな？
- ・横から見れば図Cのようになり、1個出してスタンバイしていればよい。
- ・図Dのような機構も考えられる。
- ・少し無理がある。
- ・リンク機構という手もある。
- ・カム機構でもできる。
- ・全部カムで押し出すならば、かなりの長さを取る。
- ・押し出した後は、どうするか。
- ・図Eのようにガイドラインをつけるとうよい。
- ・とにかく1個でればよい。
- ・重量は問題にならないかな？

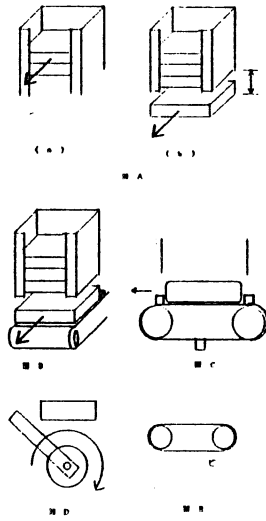


図1 プロトコル・データの例

このプロトコルを分析した例を図2に示す。この図は設計対象の変化と設計者の思考の推移を示している。設計対象は徐々に詳細化され細部が決定されていき、最終的な設計解になる。そこで設計途上での設計対象の変化と視点の変化は設計過程において重要な意味を持つ。図2から分かることは、設計者は複数の解を同時並行的に考えており、適宜一つの解の詳細化のみを進めたり、注目する解を切り替えたりする。また解決不可能な（あるいは困難な）問題に至った場合、自らの設計過程を振り返り一部をやり直すことで、問題解決を図っているということ

である。

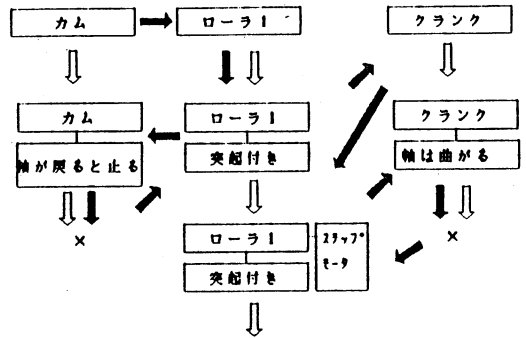


図2 設計対象の変化と設計者の視点の変化の例

さらにここで一つの決定がなされるまでのプロトコルを観察して、この中から以下の5つの過程を抽出した。設計過程をこれらの段階の繰り返しとみる。これは問題解決の過程として一般的にいわれるものと整合する（例えば [6]）。

- 1) 問題提起：解決すべき問題の発見・指摘
- 2) 提案：問題提起に対し、解決候補を出す段階
- 3) 展開：提案した候補を設計者が有している知識をもとに具体化
- 4) 評価：展開した各候補をある評価基準で評価
- 5) 決定：評価を基にどれを採用するか決定

このサイクルが繰り返され、設計対象の各部分が決定されていく（図3）。

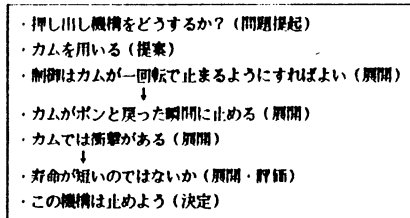


図3 設計過程の5段階の例

3 設計過程の論理による定式化

3-1 設計過程と論理

本研究では設計過程を定式化する方法として「(述語)論理」を用いる。論理は人間の思考を定式化

して表現する一つの有力な方法である。この定式化を設計に適用することで、設計過程を明示的に捉えることが可能であると考えたからである。ここでは設計過程を論理的推論(演繹推論)の過程とみることにする。

設計実験のプロトコルを観察すると、ある言明は前に行われた言明に明らかに依存している場合がある。この依存関係を論理的帰結とみることができる。

しかしまったく他の言明との依存関係がない言明も多く観察される。この中には経験や常識といった知識、すなわち長期記憶の情報やカタログといった外部の情報が含まれる。これらの知識を元にして演繹推論が行われる。

さらにこの中にはプロトコルでは観察不可能な短期記憶の情報から導かれた言明も含まれる。このような発話の解析で唐突にみえる言明は一般に発想、思いつき、直感と呼ばれる。発想、直感といったものは現時点ではそのメカニズムは明らかではなく、少なくとも論理的枠組みではとらえることはできない。そこでここでは論理的枠組みにとって長期記憶にある情報や外部の情報と同様な外から与えられる言明とする。すなわち設計過程はこれらの外から与えられる言明と前に行った言明をもとに行う演繹推論とすることができる。

3-2 非単調論理と様相論理

設計過程では試行錯誤が数多く行われている。新規設計などの前例のない設計であれば、当然であり設計の大部分が試行錯誤で行われている。ルーチン設計においても、試行錯誤的に値等を仮定することにより設計を進める場合が多い。

また、設計において設計対象に関する情報は重要である。2節でみたように、設計過程の途上において設計解にはいくつかの候補があり、それらを同時に扱っている。複数の解をうまく表現できる並列的な取り扱いが必要である。

設計の持つこれらの特徴を自然に表現する方法として、いわゆる古典的な論理に限定せず非単調論理(例えば[7],[8])と様相論理(例えば[9])を用いることにする。すなわち非単調論理で試行錯誤という特徴を表現し、様相論理で設計解の並列性の特徴を表現するが、今回はこれらの論理を用いるにあ

って、利用法を限定する。

まず根拠が明らかでない言明(発想からくる言明等)は「仮定」として扱う。これらの言明はまだ事実として決定したのではなく、現在の時点では事実であるという弱い言明であると考えられる。すなわち仮定は非単調性を含んだ言明であり、後の言明によって取り消されることが許される。例えば単純な提案は仮定となる。ここでは非単調性を実現する方法としてReiter[6]の体系を用いることにする。例えば仮定を $A\beta$ として表す。これはReiterのデフォルト・ルール「 $M\beta/\beta$ 」にあたる。

様相論理における「可能」は設計解として可能であるということにする。例えばある機構部の実現方法に複数の候補が存在するときとは、注目している世界ではその機構部に関する言明は可能様相 \diamond を用いて複数存在するが、その世界から見える世界の中に可能な言明のそれぞれが真である世界が存在する。すなわち一つの可能な解は一つの可能世界に対応する。必然様相 \square は可能な解全てにおいて真である言明に用いる。ここでは可能世界間の関係は対称的である必要はないが、推移的である必要がある(候補の候補といった場合が起こるため)。また対象領域は厳密に一致する必要はない。そこで様相体系としてはS4体系程度が利用可能であると考えられる。

4 プロトコル・データの定式化表現

前節で述べた方法にしたがって実験のプロトコルを定式化したものの例が図4である。

```

問題提起(売り切れ検知機構(本体、?))
(w1)  $\diamond$  A 売り切れ検知機構(本体、光センサ)
(w1) D 大(コスト(光センサ))
(w2)  $\diamond$  A 売り切れ検知機構(本体、ばね)
(w2) 売り切れ検知機構(本体、ばね)  $\rightarrow$  力がかかる(タバコの箱)
(w2)  $\diamond$  力がかかる(タバコの箱)
(w2) 弱い(x)  $\rightarrow$   $\sim$ 力がかかる(x)
(w2) 弱い(タバコの箱)
(w2)  $\sim$ 力がかかる(タバコの箱)
(w2)  $\diamond$  力がかかる(タバコの箱) and  $\sim$ 力がかかる(タバコの箱) = 矛盾
(w2)  $\diamond$   $\sim$ 売り切れ検知機構(本体、ばね)
(w1)  $\sim$ 大(コスト(光センサ))
評価()
決定((売り切れ検知機構(本体、?)、? = 光センサ)
売り切れ検知機構(本体、光センサ))

問題提起(押し出し機構(本体、?))
(w3)  $\diamond$  A 押し出し機構(本体、カム機構)
(w4)  $\diamond$  A 押し出し機構(本体、ローラ機構)
(w4) 押し出し機構(本体、ローラ機構)  $\rightarrow$  D 滑る(タバコの箱)
(w4)  $\diamond$  D 滑る(タバコの箱)
機構簡單(x, y)  $\rightarrow$  大(信頼性(x), 信頼性(y))
機構簡單(ローラ機構、カム機構)
    
```

図4 プロトコル・データの定式的表現の例

この結果から設計過程と論理の関係について以下
のことがいえる。

- ① 演繹は設計過程を部分的ではあるが説明することができる。全体としては演繹とそれ以外が混ざり合いながら設計は進んでいると見ることができる。また単純な演繹だけではない。
- ② 様相記号の導入によって設計の進行と設計者の視点の変化を自然に表現できる。また知識の"有効範囲(scope)"の違いを表現できる。
- ③ 設計においては非単調推論と似た形の矛盾の回避や後戻りが行われていることが分かる。

5 解決すべき問題

ここでは前節で挙げた点のうち、特に①の問題に注目する。①の問題はさらに「現実的」演繹という問題と演繹の枠外の過程の問題に分けられる。

5-1 演繹における問題

論理式集合Sに対し、演繹可能な論理式の全体を

$$Th(S) = \{p : S \vdash p, p \text{ は論理式}\}$$

とすると、 $Th(S)$ はSに比べて極めて大きくなる。このため $Th(S)$ を求めることは実際的ではない(多くの場合不可能)。実際問題としては、

$$Th'(S) \subset Th(S)$$

なる $Th'(S)$ を求めることになる。これは当然論理における完全性は満たしていない。しかし制限の方法($Th'(S)$ の作り方)が明示的であれば実際の問題として有用である。

設計中では明示的あるいは暗黙にこのような制限を使っている。例えば注目する問題についての議論のみを進めることは、注目する述語や項に関する論理式のみを導出を行っていると見ることができる。欲しい事実の正当化は後向き推論、見方の転換は用いている論理式集合の切り替え、解くべき手順の指定は論理式集合の順次指定であると見ることができる。

このようなヒューリスティックな制限方法を利用すると、演繹を基本として設計手順や経験的設計法を記述することが可能である。

5-2 演繹自体の問題

演繹自体は公理系を変更しない。公理系を与えられた知識の世界だとすると、与えられた世界を変更しないことになる。このように考えるとき、設計に

含まれる、世界になかったものを作り出す行為(知識のそのものの変更行為)は演繹では捉えることはできない。

例えば、演繹において、

$$(1) p(a), \forall (p(x) \supset q(x)) \vdash q(a)$$

は導ける。しかし、

$$(2) q(a), \forall (p(x) \supset q(x)) \vdash p(a)$$

はできない。これは $p(a)$ 、 $q(a)$ がある新しい公理系を作る行為である。(2)のような行為を可能にする方法としてabduction(仮説形成)やモデルの極小拡大[10]などが考えられる。

実際の設計は(1)と(2)が交互に行われている。例えば展開・評価は(1)、提案は(2)というように交互に現れている。(2)を可能にする方法と(1)との組合せを実現することが必要である。

6 まとめ

本研究では様相論理と非単調論理の枠組みを中心とした設計過程の定式化の方法を提案し、その定式化を設計実験のデータに適用することでこの定式化の有効性を確認した。

今後は、①設計実験の方法論の研究と実験の継続、②設計過程を記述するのに必要十分な論理についてさらに考察を加えること、③本稿で用いた論理を基に推論可能な方式の確立とシステムの実現、という3点を中心に研究を進めていく予定である。

参考文献

- [1] R. ten Hagen and T. Tomiyama (ed.): "Intelligent CAD Systems I: Theoretical and Methodological Aspects", Springer-Verlag, Heidelberg (1987)
- [2] H. Yoshikawa and D. C. Gossard: "Intelligent CAD", North-Holland, Amsterdam (to be published in 1988)
- [3] 吉川弘之・荒井栄司・後藤敏彦: 実験設計学、精密機械47, 7(1981)830-835
- [4] D. G. Ullman et al: Preliminary Results of an Experimental Study of the Mechanical Design Process, Results from the NSF Workshop on the Design Process, (1987)145-186.
- [5] K. A. Ericsson & H. A. Simon: "Verbal reports as data", Psychological Review Vol. 87 (3) 215-251 (1980)
- [6] 三宅なほみ: 理解におけるインターアクションとは何か、佐伯編: 理解とは何か、東大出版会(1985)
- [7] R. Reiter: "A Logic for Default Reasoning", Artificial Intelligence 13(1980) 81-132
- [8] D. McDermott and J. Doyle: "Non-Monotonic Logic I", Artificial Intelligence 13(1980) 41-72
- [9] G. E. Hughes & M. J. Cresswell: "An Introduction to Modal Logic", Methuen and Co., Ltd. London (1968)
- [10] T. Yagih: "A Third Meaning of Logical Formula", IFIP WG.5.2 Workshop on Intelligent CAD(1987)