

設計シミュレーション

武田 英明*・富山 哲男*・吉川 弘之*

ABSTRACT This article discusses building a computable design process model, which is a prerequisite for realizing intelligent computer-aided design systems and design simulation based on the computable model. First, we classify the methodology of modeling design processes and describe the approach of our research. Then we show a cognitive design process model obtained by observing design processes using a protocol analysis method. We then discuss a computable model that can explain most parts of the cognitive model. In the computable model, a design process is interpreted as an iterative logical process realized by abduction, deduction, circumscription, and meta-level reasoning. Based on the computable model, we implemented a design simulation system by which we show the computable model is appropriate for representing design processes.

1. まえがき

近年多くの設計においてCAD (Computer Aided Design) が使われるようになってきた。その多くは製図作業の支援であり、製図作業の効率化、設計図面の計算機化という面では大きな貢献をしている。しかし、製図は設計の一作業であり、製図作業の支援だけでは設計の本質的支援とはいえず、製図に至るまでの過程、すなわちより上流の設計過程から支援していく必要がある。設計の上流過程といった知的な部分を含め設計全体を計算機で支援するシステムがインテリジェントCADであり、近年多くの研究がなされている(例えば^{1),2)}など)。

ここでは、インテリジェントCADシステムの実現にあたって重要な問題である設計過程のモデリングについて論じる。インテリジェントCADによって設計者を統合的かつ知的に支援するためには、まず多様な設計対象が計算機で統合的に表現でき、かつ操作できることが必要である。しかしそれだけではなく、さらにシステムが設計過程のモデルを用意することにより、設計者の行為を理解して、適当な支援を行なうことができる。すなわち、設計者の設計作業の意味を理解して適当な支援や設計者の思考の代替を行なうため

には、設計過程の表現という面も考察する必要がある。

本稿では設計過程のモデリングのいくつかの側面を紹介すると共に、インテリジェントCADの実現に必要な設計過程モデルについて考察する。設計過程モデルのCADへの利用を考えたとき、単に設計作業という現象を説明するというモデリングでは留まらず、計算機において表現可能かつ利用可能なモデルでなければならない。そこで本稿では、論理的枠組みによる過程モデルに基づく設計過程の再現(設計シミュレーション)を行ない、設計過程モデルの有効性を検証することによってCADへの利用可能性を検討する。

2. 設計過程のモデリング

設計過程のモデリングはいくつかの観点から分類することができる³⁾。ここでは、3つの観点、すなわち、対象とする設計過程が現実の設計過程かあるいは「あるべき」設計過程かによる分類、設計のどの点を中心にモデル化するかという面からの分類、形式化という面から分類によって考える(表1参照)。

まず、実際に行なわれている設計を対象とするのか、行なうべき理想的設計を対象とするかで異なってくる。後者は「よい設計」をするにはどうしたら良いかを示すモデルであり、規範的(prescriptive)と呼ばれる(例えばSuh⁴⁾の公理的設計論)。このモデルは実際の設計作業で役に立つことがその目的であるが、その根拠は長年の経験の積み重ねなどであるので、一

Design Simulation. By Hideaki Takeda, Tetsuo Tomiyama and Hiroyuki Yoshikawa (Dept. of Precision Machinery Engineering, Faculty of Engineering, The University of Tokyo).

*東京大学工学部精密機械工学科

表1 設計過程モデルの分類

分類の視点	分類の例
対象	実際の設計 (記述的モデル) 理想的設計 (規範的モデル)
注目する側面	設計者の認知的行為 (認知的モデル) 社会的側面 ...
形式化手段	言語的 図式的 数学的 計算可能、推論可能 (計算可能モデル) ...

一般性や拡張性を議論することが難しい。これに対して前者は記述的 (descriptive) と呼ばれる。

設計には多種多様な要因が関わっており、そのなかでものを作り出すときの情報の流れに着目すると、図1のような要因をあげることができる。ここにある要因は、設計者 (designer)、設計 (designing)、製造 (manufacturing)、製造者 (manufacturer)、生産環境・システム (production environment or production system)、使用者 (user)、欲求 (desire)、要求 (requirement)、人工物 (artifact) である。

設計過程を考察するとき、このなかのどの部分を中心に考えるかによって作られるモデルは異なってくる。例えば、要求仕様から解までの流れに注目するのが多くの設計過程のモデルである。認知的な分析によるモデル (認知的モデル, cognitive model) とは設計者の行為を中心にモデル化したものであり、設計者の行為を認知的なレベルで説明する。あるいは、この図の全体的な流れと生産環境・システムという面に注目すると、社会的モデル化となる。

さらにその記述の方法によっても異なってくる。一般に経験的な設計過程モデルは言語による記述がなされることが多いが、なんらかの図式表現を使って表現

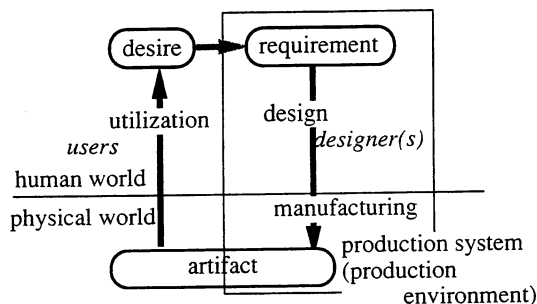


図1 設計に関わる要因

されることもある。しかし、工学的議論となるためには、ある一定以上の形式的手段によって記述されることが必要となる。そのひとつとして数学的手法に基づく方法などがあり、例えば一般設計学³⁾では集合論を用いた公理的方法を用いている。さらに、インテリジェント CAD への利用という点を考えた時は、そのモデルが計算機によって計算できる、あるいは推論できるようなレベルであることが望ましい。これを特に計算可能モデル (computable model) と呼ぶ。

この中で本稿では、記述的、認知的、計算可能の各点に注目する。まず、設計を観察することからはじめ、設計過程のモデルを認知的観点から考察する。このモデルにおいては図式的表現によって行なう。次にこの認知的モデルを踏まえ、そのモデルを説明できる形式的モデルを考察する。ここでは単に記述的であるだけでなく、計算可能であるようなモデルを考察する。

3. 設計実験と認知的モデル

本節では設計の観察とその結果を利用した認知的モデルについて述べる。

3.1 設計実験

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⁶⁾。観察方法とは主に心理学での実験で用いられているプロトコル解析 (protocol analysis) の手法を利用している。プロトコル解析とは、被験者に考えたことを声を出しながら (think aloud)、課題を解いてもらい、その発話を記録して解析するというものであり、心理学で思考過程を調べるときに用いられている。一般にプロトコル解析では被験者は1人で課題を解いていくが、ここでは被験者を2名とし、対話をしながら1つの課題を解くという対話型の実験とした⁷⁾。

このようにして収集したプロトコル・データをもとにいくつかの視点から分析を行なった。

まず、設計対象の変化と設計者の思考の推移に注目して分析を行なった。設計者は設計対象を徐々に詳細化していき、最終的な設計解にしていく。そこでプロトコル・データの中の設計対象の記述に関する部分に注目して、設計対象の変化を抽出した。図2にその一部を示す。連結した1つの図形が、ある段階の設計対象の記述を示している。既に設計した対象を元に新たに情報を加えて新たな対象の記述を作った場合、それを対象の進化と呼ぶことにする。この設計対象の進化の方向を黒色の矢印で示す。また、設計者がどの順で

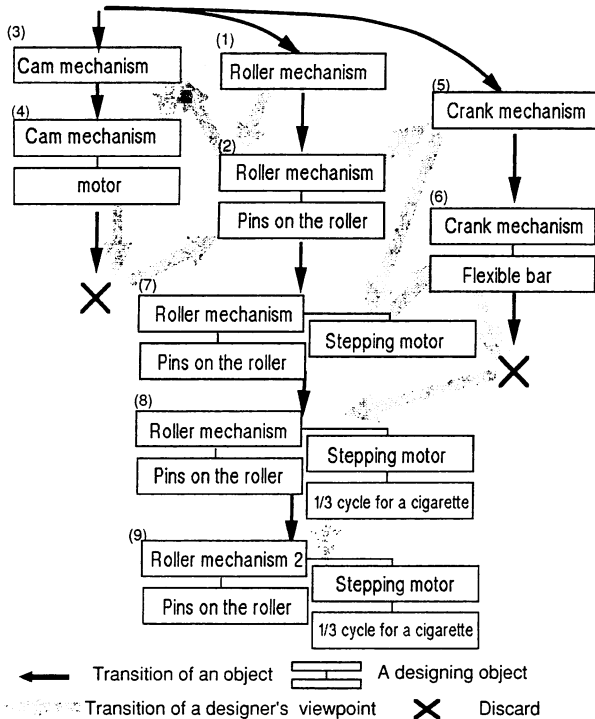


図2 設計対象の変化と設計者の視点の変化

設計対象を考えていったかも同時に調べた。図2の灰色矢印は設計者の視点の変化の方向を示す。

2番目の分析としては、1つの決定がなされるまでの過程を分析した。プロトコル・データを観察して、この中から「問題提起」、「提案」、「展開」、「評価」、「決定」という5つの過程を抽出した。

問題提起 解決すべき問題の発見・指摘。

提案 問題提起に対し、解決候補を出す段階。

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。この途上で問題点が生ずると、これを新たな問題提起とする。

評価 展開した各候補をある評価基準で評価。この段階でも途上で問題点が生ずると、これを新たな問題提起とする。

決定 評価を基にどれを採用するか決定。

プロトコル・データの発話の多くはこれらの段階のいずれかに分類することができる。設計過程をこの5段階の繰り返しで設計対象の各部が決定されていくとみることができる。すなわちこの5つの段階の繰り返しによって図2での1つの黒色矢印が作られるので、これを設計サイクルと呼ぶことにする。図3にこの5段階による設計の進行の例を示す。

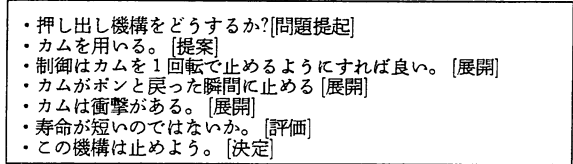


図3 設計サイクルの例

4. 論理的枠組みによる設計過程モデル

4.1 基本的枠組み

前節では認知的な視点からの設計過程のモデル化の試みについて述べたが、本節ではこの認知的なモデルを説明するような形式的かつ計算可能なモデルについて述べる。そこでまず設計過程を論理を用いてモデル化することを考える。すなわち、設計対象の記述やその対象のもつ性質、要求仕様、設計に用いられる知識などを論理式の集合と考え、論理的な推論で設計をモデル化する。

このような側面から先の認知的モデルを見た場合、設計過程、対象の記述、知識の記述などには以下に示す特徴がある。

1. 設計では必ずしも仕様と知識が事前に完全に与えられているわけではない。仕様は設計が進行するうちに詳細化されたり、修正されたりする。また、知識も予め用意されているのではなく、設計過程において順次利用する知識が明確になってくる。
2. 設計は詳細化の繰り返しであり、段階的に進行するが、単に仕様と知識を入力すると結果として解がでるのではなく、試行錯誤的に進行する。
3. 設計における知識には、対象の性質や挙動に関する知識と「いかに設計をするか」という設計方法知識があり、この両方を扱う必要がある。
4. 知識は矛盾することがある。矛盾が生じた場合、その矛盾を解決することが設計が進行する。

そこで、次のような基本的な枠組みで考察する。

まず、対象に関する知識 K_o とは、設計対象の記述 D_s からその対象のもつ性質 P を導出するものであるとする。すなわち、

$$D_s \cup K_o \vdash P \quad (1)$$

である。このとき、設計とは対象に関する知識 K_o のもとで、要求仕様 R が示されたとき、

$$D_s \cup K_o \vdash R \quad (2)$$

を満たす設計対象記述 Ds を求めることになる。当然、この設計対象の性質の記述 P は R を含むはずである。

$$P \supseteq R \quad (3)$$

すなわち、この枠組みにおいて、設計とは要求されている性質 (R) を満たすような設計対象 (Ds) を abduction により見出す行為である。Abduction とは Pierce により提唱されたものであり、ある定理が与えられた時、それを導くことができる公理を仮説を求める推論である⁹⁾。

このとき、設計対象を段階的に詳細化していく設計過程は abduction と演繹 (deduction) を交互に繰り返しながら設計対象を段階的に詳細化していく過程と定義される。Abduction によって出された仮説 (設計対象) から対象のもつ性質を演繹によって導出することにより、その仮説の働きを検査することができる (式 (3) の確認)。結果によっては仮説を修正する、あるいはその仮説を放棄し別の仮説を探したり、その上でさらに abduction を行ない、設計対象を詳細化する必要がある。

さらに矛盾の取り扱いと知識の変更に対しては次のように考える。

設計における知識の矛盾の多くの場合、知識自身の不完全性から生じている。すなわち本来用いられるべき状況以外の場面で知識が用いられた、あるいは本来あるべき条件が書かれていなかったことから、見かけ上矛盾になると考える。したがって、表面には記述されていなかった条件を明示化すれば見かけ上の矛盾は解消する。しかし、そういった条件を全て求めることは不可能であるので、ここでは矛盾が生じた時はじめて、その条件を求めることにより、不完全な知識での推論を行なう。この条件を求めるには、circumscription⁹⁾ で行なうことができる (次項参照)。このようにして知識の条件を付加することにより、設計知識自身も詳細化される。

また、設計に必要な知識ははじめから用意されているわけではなく、その設計を進めるのに必要な知識が設計過程に判明して、それらの知識が順次用いられていくと考えられる。このような利用する知識の変化を扱うためにメタレベル推論 (meta-level reasoning) という方法を利用する。この推論では対象言語とメタ言語の関係は、対象言語での公理と論理的帰結の関係がメタ言語での素論理式になる。設計対象や対象の性質を表現する言語系を対象言語とすると、そのメタ言語において知識の利用方法や設計の進め方に関する知

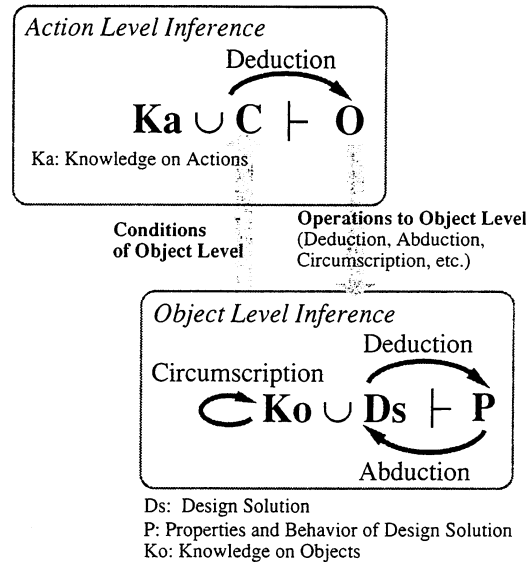


図4 設計過程の論理的モデル

識をもとにした利用する対象知識の変化に関する推論を行なうことができる。

これを模式的に示すと図4になる。すなわち、対象レベルにおいて演繹と abduction の繰り返しにより設計対象が詳細化され、矛盾が起こると circumscription により知識が変更される。また、対象レベルにおいて利用される知識や仕様や対象は、行為に関する知識 (メタ言語における知識) による推論によって決定される。

4.2 設計過程の推論モデル

ここでは3節で挙げた認知的設計過程モデルを論理的枠組みで解釈する推論モデルについて議論する。

「提案」段階では設計者は求められている性質・挙動からもっともらしい解の候補を導きだす。しかし、それが本当に求めている解であるかどうかまでは決定していない。これは一種の仮説形成の推論であり、abduction として解釈することができる。

他方、「展開」と「評価」の段階は演繹として解釈する。これらの段階では設計者は自分の持つ知識を適用して、わかりうることを全て導くことにより、解候補が求めている性質を持っているかを調べている。つまり先の「提案」段階とは逆の過程であり、演繹で解釈する。「展開」と「評価」の段階の違いは、「展開」段階では主に解の候補が持つ性質等を導く知識が使われるのに対して、「評価」段階では性質を比較する知識が使われる。

「展開」段階と「評価」段階において、設計者は解に問題点・困難な点を見つけると、新たな問題提起を行なう。この過程は矛盾を起こした知識を対象にした

circumscription で解釈する。Circumscription とは「与えられたこと以外は考慮しない」ときの人間の思考の形式化として提案されたものであり⁹⁾、指定された述語の外延を最小にすることにより実現される。

この推論は矛盾する知識での例外事項の計算に用いられる。すなわち例外を示す述語を付加してその外延を最小にすることにより、必要な例外を求めることができる。例えば、

$$x \text{ はバネである} \rightarrow x \text{ で } w \text{ と } d \text{ は比例する} \quad (3)$$

$$x \text{ はバネである} \wedge x \text{ は過負荷である}$$

$$\rightarrow \neg (x \text{ で } w \text{ と } d \text{ は比例する}) \quad (4)$$

という知識 (x は設計対象を、 w は重量を、 d は変位を示す変項) があるとき、

$$\{x \text{ はバネである}, x \text{ は過負荷である}\} \quad (5)$$

を仮定すると矛盾が生じる(「 x で w と d は比例する」の肯定と否定が両方導かれる)。そこで、

$$x \text{ はバネである} \wedge \neg ab1$$

$$\rightarrow x \text{ で } w \text{ と } d \text{ は比例する} \quad (6)$$

$$x \text{ はバネである} \wedge x \text{ は過負荷である} \wedge \neg ab2$$

$$\rightarrow \neg (x \text{ で } w \text{ と } d \text{ は比例する}) \quad (7)$$

というように例外を示す項 (abnormal 述語 ab_i) を各式に付加して、circumscription を行なう。1つの解は、

$$ab1 = x \text{ はバネである} \wedge x \text{ は過負荷である} \quad (8)$$

$$ab2 = \text{false} \quad (9)$$

である。これらを式 (6), (7) に代入を行うと、

$$x \text{ はバネである} \wedge \neg (x \text{ は過負荷である})$$

$$\rightarrow x \text{ で } w \text{ と } d \text{ は比例する} \quad (9)$$

$$x \text{ はバネである} \wedge x \text{ は過負荷である}$$

$$\rightarrow \neg (x \text{ で } w \text{ と } d \text{ は比例する}) \quad (10)$$

となる。このとき (5) の仮定において矛盾はなくなる(「 x で w と d は比例する」の否定しか導かれない)。しかし、もしここで要求仕様が「 x で w と d は比例する」であった場合、この仮定は要求仕様を満たさないことになる。そこでこの要求仕様を満たすような仮定を式 (9), (10) を利用して求めると、今度は

$$\{x \text{ はバネである}, \neg (x \text{ は過負荷である})\}$$

$$(11)$$

となり、「 x は過負荷である」を否定する必要が出てくる。これは解決すべき新しい問題になる。

「展開」や「評価」段階で生じる問題提起はこのように対象レベルの circumscription として解釈できるが、それ以外の場合のうちある場合は行為に関するメタレベル推論として解釈することができる。「問題提

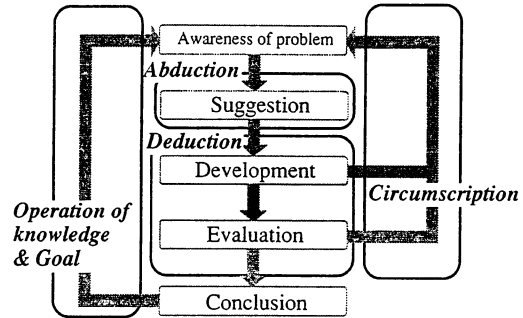


図5 設計サイクルの推論モデル

起」段階は現在の対象のなかで問題とする部分の決定とその問題を解決するのに必要な知識を用意することである。これは、行為レベルの推論による対象レベルの状態 (D_s や P , K_o) の操作を行なう段階と解釈することができる。

例えば以前にも設計したことがある場合では、設計者は設計対象のどの部分を問題とすればよいかを知っている。このような経験的設計知識は対象に直接関連する知識ではなく、設計者の行為に関する知識(ここではメタレベルの知識)である。また、問題が決められたときに、その問題を解決するために必要な対象知識を指定する知識も、設計者の行為に関する知識である。

設計サイクルは以上の対象レベルの3つの推論 (abduction と演繹と circumscription) と行為レベルの推論によって解釈される (図5参照)。

4.3 設計対象の表現モデル

次に設計過程の中で詳細化されていく設計対象の表現について考察する。ここでは設計の進行に従い、段階的に変化する対象の記述やその性質の記述を様相論理での可能世界で表現する。ただし、ここではデータ論理 (data logic) という様相論理を利用する。

設計対象の記述は前項で述べたように設計の進行に伴い変化していくが、abduction により変化した場合、そこから導かれる対象の性質は保存される。すなわち、ある時点での対象の記述 D_{s_n} 、対象の性質の記述 P_n 、対象知識を K_{o_n} 、次の時点での対象の記述 $D_{s_{n+1}}$ 、性質の記述 P_{n+1} 、対象知識 $K_{o_{n+1}}$ とするとき、 $D_{s_{n+1}}$ は

$$D_{s_{n+1}} \cup K_{o_{n+1}} \vdash D_n \quad (12)$$

なので、 D_{s_n} と $D_{s_{n+1}}$ の論理的帰結の集合である P_n と P_{n+1} は、

$$P_{n+1} \supseteq P_n \quad (13)$$

である (ただし、 $D_{n+1} \neq D_n$ である)。すなわち、性質

の記述は単調増加する（これまで導かれていたことが覆ることはない）。

このように変化する対象に関する記述を管理する問題を考える時、推論で導かれたものとそうでないものを区別して扱う必要がある。そこで、ここではデータ論理 (data logic)^{10),11)}の枠組みを利用する。データ論理は直感的には部分論理と様相論理を合わせたものであり、真理値としては真 (t)、偽 (f) の他に未知 (u) がある。この3番目の真理値は直感的には「未定」という意味を持ち、いずれは真か偽になることが期待される真理値である。この真理値において、 $t \sqsubseteq u, f \sqsubseteq u$ という半順序をつけ、可能世界間の到達可能性関係はそこでのすべての命題において \sqsubseteq が満たされる時のみつけられる。すなわち、ある可能世界から到達可能な世界では、もとの世界で真や偽である命題の真理値は変わらず、さらにもとの世界では未知であったものが真か偽になっていることがある。すなわち、ある可能世界から到達可能な可能世界は「より真偽が決定した」状態をもつ可能世界である。また、この論理における必然様相とはその世界から到達可能な世界で決して偽にならないことを示し、可能様相はひとつでも真になる世界があることを示す。

この各可能世界を設計の各段階に対応付ける。

取り消しや修正がない設計の進行は可能世界を順次つくって枝を伸ばしていくことに相当する。これに対して、修正や取り消しを行なった過程は適当な世界まで可能世界の枝を遡り、新しい枝として可能世界をつくることに相当する。すなわち、可能世界のつくる構造は、要求仕様が与えられた段階から設計解にいたる木構造になる。例えば図2に対応する可能世界は図6になる。ここで、対象 (8) から対象 (9) への変化は修正をしているので、世界8へつながるのでなく、技

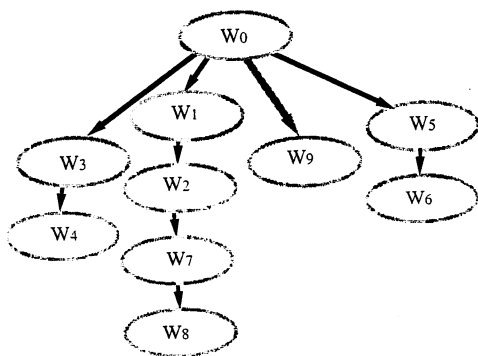


図6 可能世界による設計過程の表現例

を遡って適当な世界（ここでは世界0）からの枝として新しい世界9が作られている。

また、このとき、様相を利用した記述をすることが可能であり、例えば要求仕様は必然様相を用いて表現される（どこでも偽にならないことが許されない）。ひとつの世界で真になった命題は、その世界をみることのできる世界（同じ枝のより根に近い世界）ではその命題の可能様相は真となるので、可能な解を可能様相により知ることができる。

5. 設計シミュレーション

前節では認知的モデルを論理を用いて形式的に表現可能であることを示した。この論理モデルは推論可能な要素により構成されており、計算可能であることが期待される。本節ではこのモデルに基づくシステムを構成し、実際に推論が可能であることを示すと共に、実際の設計プロトコルを利用して設計シミュレーションを行なって、このモデルが設計過程を表現するのに妥当であることを示す。

5.1 システムの構成

図7に実際に作成した設計シミュレータの構成を示す。このシステムは主に行為レベル推論部、対象レベル推論部、知識ベース、多重世界管理部、ATMS (Assumption-Based Truth Maintenance System) 部からなり、さらに対象レベル推論部は対象記述 (D_s)、対象性質記述 (P)、対象知識 (K_o) のワークスペースと abduction, deduction, circumscription の3つの推論部からなる。

行為レベル推論部は一種のルールベースシステムであり、そこでの知識は対象レベルの知識ベースの選択知識、あるいはスケジューリング知識などを含む設計方法に関する知識である。行為レベルの推論は対象レ

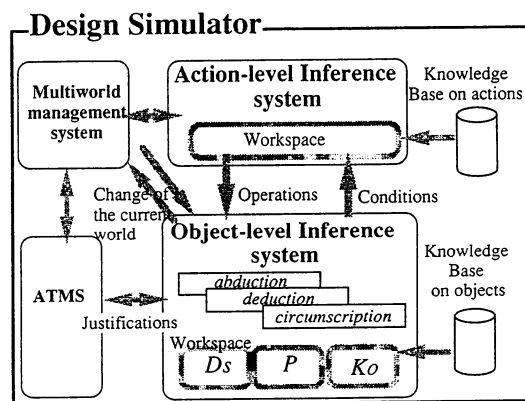


図7 設計シミュレータの構成

ベルからその状態の報告を受け、対象レベルの状態の操作あるいは対象レベルで行なうべき推論を決定する。

対象レベル推論は D_s , P , K_o の各ワークスペースに対する、abduction, deduction, circumscription のいずれかである。Abduction は D_s の内容を、deduction は P の内容を、circumscription は K_o の内容を変化させる。 K_o の内容は Horn 節を基本とする論理式（ルール）であり、 D_s , P はそれぞれアトム式（事実）の集合である。知識は知識ベースの単位で管理・操作される。

abduction を実現する推論はいくつかの方法が提案されているが、ここでは Poole の default 推論¹²⁾に基づいて実現している。circumscription は中川¹³⁾によって示された方法に従って行なった。また、Abduction による対象記述の変化と世界間の関係は Morris¹⁴⁾の方法を利用した ATMS¹⁵⁾を作成して管理している。

最初に要求仕様が入力すると、システムは行為レベルの推論と対象レベルの推論を交互に実行することを

繰り返す。そして、もはや何も新しいことができなくなるとシステムは終了する。対象レベルの推論は D_s や P や K_o の内容を abduction, deduction, circumscription のいずれかの推論によって更新し、場合によっては矛盾を起こす。このような対象レベルの状態は行為レベルに報告され、行為レベルの推論はこの状態を基に推論を行ない、対象レベル推論部への操作（ワークスペースへの操作または推論の実行）を指示する。以上の繰り返しにより、 D_s および P は段階的に詳細化が行われていく。

ユーザは行為レベルから対象レベルへ変わるときに、次の操作の選択というかたちで関与する。またユーザは適宜、作ってきた可能世界の構造を移動して、複数の候補を比較したり、指定した世界からの対象レベルでの推論を継続することができる。

システムは Sun-4 上で Allegro Common Lisp, X11, CLX (Common Lisp X Interface) を用いてインプリメントされた。図 8 にその表示の例を示す。

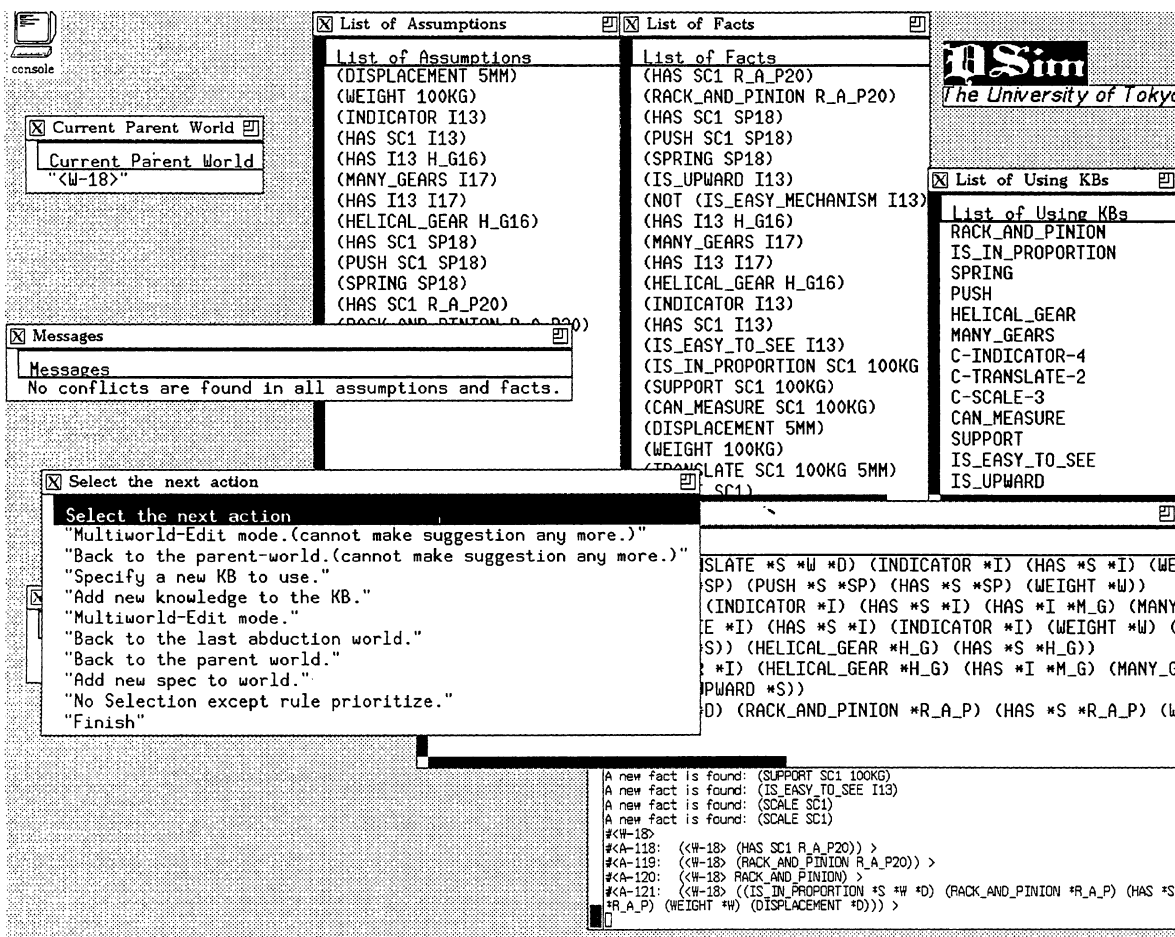


図 8 設計シミュレータの表示の例

5.2 設計シミュレーションの例

この設計シミュレータを用いて、設計実験から得られたプロトコルデータを利用して知識ベースをつくり、簡単な設計過程のシミュレーションを行った。

もともとなったプロトコルは図9に示すもので、体重計の設計を課題としたプロトコルの冒頭部である。この部分では、設計者は体重計の基本構造を決めているが、単に構造が順次決定していくのではなく、何度か自己の案に修正を加えながら進めている。この部分に対応する知識を論理式の形式で用意し、設計シミュレーションを行った。

図10はDs(対象の記述)とP(性質の記述)の変化を示す。Dsは対象の記述をPは性質の記述(Dsと重複する内容はこの図では省略している)を示す。世界1は仕様が入力された状態である(図10(a))。

ここから行為レベルの推論によって用意された関連する知識を用いて、abductionとdeductionを3回づつ行なうと、世界5になる(図10(b))。ここでは表示部(indicator)を用いるという点まで進んでいる(発話(8)までに対応する)。

その後、発話(11)(12)に対応する知識がKoに入ると、translateという述語において矛盾を生じる(図10(c))。これは発話(11)(12)の部分と発話(5)の

- 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1)。
- 3 ばねは下から押すのではなくて逆でもいい。
- 4 押すとすればこうなる(図2)。
- 5 ラックとピニオンでこうすれば(図3)、重さと変位が比例するのを利用して測れる。
- 6 他に重さをささえられるものはあるか。
- 7 やはりばねしかない。
- 8 とりあえずその前に表示を考えよう。
- 9 要するに変位(図1のx)が測ればなんでも良い。
- 10 見易い方がいいだろう。
- 11 5mmの変位を100kgにするのだから1kg当りの変位は0.05mmだ。
- 12 こんなのはこれ(図3)では無理だ。
- 13 精度はともかくとしてギアを何段もかませればできる。
- 14 しかし売っているものは表示が上から見えるようになっている。
- 15 するとこれでははすば歯車を使ってやるしかないが、売っているものはもっと単純な機構に違いない。

fig. 1 x fig. 2 fig. 3

図9 例題のプロトコル

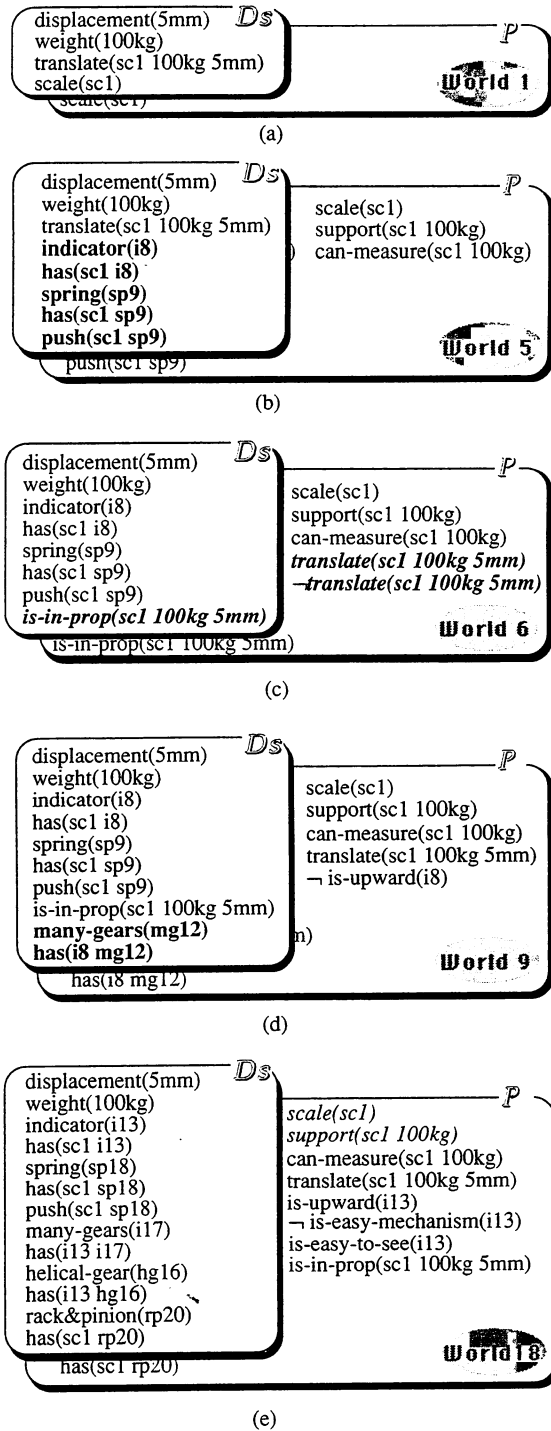


図10 対象とその性質の変化

部分が矛盾するからである。そこで circumscriptionを行ない、発話(5)に対応する論理式を変更する。すると矛盾は回避され、新しい案(多段歯車を使う案)が出てくる(世界6, 図10(d))。

このようにして対象は詳細化され、最終的に図10(e)(世界18)のような結果となる。この間に作られ

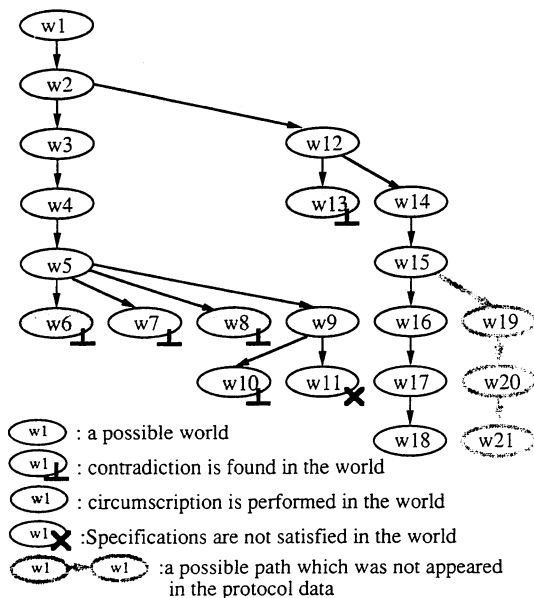


図11 生成された可能世界

た世界を図11に示す。ただし、世界19から世界21までは実際の設計過程では行なわれなかった過程で、世界21は世界18とは異なる結果になっている（バネを圧縮方向に使うが、引っ張り方向に使うかの違い）。

この論理による設計過程モデルに基づく設計シミュレータから以下のことがいえる。

- ・実際に行なわれた過程にそった設計過程を生成することができる。
- ・実際には行なわれなかった分岐も生成することができる。
- ・設計の結果、設計者の知識が変化することを、対象知識の収集と circumscription による知識の変更で実現している。

このように論理的設計過程モデルに基づく設計シミュレータは設計過程の再現を可能にしている。したがって、この論理による設計過程モデルは、計算可能であるだけでなく、設計過程のモデル化の方法として有効であるといえる。なお、現在この設計シミュレータを利用してインテリジェント CAD の開発を行なっている¹⁶⁾。

6. おわりに

本稿ではインテリジェント CAD 構築に向けた設計過程のモデル化について述べた。インテリジェント CAD 構築には、計算可能な設計過程モデルが必要で

あり、その研究のために設計シミュレーションという手法を提案した。ここでは論理的手法による設計過程モデルと設計シミュレーションを取り上げて議論を行なったが、今後はより多様なモデルの提案や設計過程のシミュレーションが行なわれ、インテリジェント CAD 構築への議論が深まることが望まれる。

参考文献

- 1) 吉川弘之, 富山哲男(編): インテリジェント CAD(上) —理念とパラダイム—. 朝倉書店 (1989)
- 2) 吉川弘之, 富山哲男(編): インテリジェント CAD(下) —テクノロジーと展望—. 朝倉書店 (1991)
- 3) J. R. Dixon: On research methodology towards a scientific theory of engineering design. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing (AIEDAM)*, 1-3, 145/157 (1987)
- 4) N. P. Suh, A. C. Bell, and D. C. Gossard: On an axiomatic approach to manufacturing and manufacturing system. *Journal of Engineering for Industry*, 100, 127/130 (1978)
- 5) 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, 45-8, 906/912 (1979)
- 6) 吉川弘之, 荒井栄司, 後藤敏彦: 実験設計学—一般設計学のための実験的方法—. 精密機械, 47-7, 46/51 (1981)
- 7) 武田英明, 富山哲男, 吉川弘之: インテリジェント CAD のための設計過程の分析と論理による形式化. 精密工学会誌, 57-6, (1991). (印刷中)
- 8) 内田種臣(編). パース著作集 2: 記号学. 勁草書房 (1986)
- 9) J. McCarthy: Circumscription—a form of non-monotonic reasoning. *Artificial Intelligence*, 13, 27/39 (1980)
- 10) F. Veltman: Data semantics. In J. A. G. Groenendijk, T. M. V. Janssen, and M. B. J. Stokhof, editors, *Formal methods in the study of language*, 541/565. Mathematisch Centrum, Amsterdam (1981)
- 11) F. Landman: *Towards a Theory of Information: the Status of Partial Objects in Semantics*. Foris, Dordrecht (1986)
- 12) D. Poole: A logical framework for default reasoning. *Artificial Intelligence*, 36, 27/47 (1988)
- 13) 中川裕志, 森辰則: 論理型言語における circumscription. 情報処理論文誌, 28-4, 330/338 (1987)
- 14) P. Morris and R. Nado: Representing actions with an assumption-based truth maintenance system. In *Proceedings of Fifth National Conference on Artificial Intelligence (AAAI-86)*, 13/17. The American Association for Artificial Intelligence, Philadelphia (1986)
- 15) J. de Kleer: An assumption-based TMS. *Artificial Intelligence*, 28, 127/162 (1986)
- 16) 薛徳意, 富山哲男, 吉川弘之: 設計知識表現言語 IDDL の開発 (第三報)—拡張仕様—. 1991年度精密工学会春季大会講演論文集, 23/24 (1991)