

知的CADのための設計シミュレーション

○武田英明（東京大学）

薛德意（東京大学）

吉川弘之（東京大学）

林千登（富士ゼロックス）

富山哲男（東京大学）

1. 設計シミュレーションの意味

設計者を設計全般にわたって支援するCADの実現のためには、対象物の表現に関する研究と設計過程の研究、すなわち要求仕様に基づいて対象物を作り出す過程についての研究が必要である。対象物の表現に関しては、形状の表現は幾何モデリング、さらに属性情報等も含めた表現にはプロダクト・モデリング等が研究されてきた。これに比べて、設計過程に関する研究はまだ少ない。しかし単に対象物の表現のみでは、設計を知的に支援しているとはいえない。設計自体の支援のためには、要求から設計解に至る過程のなかで計算機に可能な部分は計算機が担当するといった形で、設計者とCADが協調して設計を行うことが必要である。このためには設計とはどのようなものであるか、設計がどのように行われるか、等の設計過程に関する研究が不可欠である。筆者らはこの設計過程に注目し、設計過程のモデルについて研究してきた[1][2][3][4]。

知的CADのための設計過程のモデルは単に認知科学的に妥当なモデルであるだけでは十分ではなく、計算可能 (computable) なモデルであることが必要である[5]。前者の意味でのモデルも設計過程の研究を進展させるには重要であるが、それ自身の評価は困難である。これに対し計算可能なモデルは、

- (1) 数学的な取り扱いが可能である
- (2) 計算機上に実現することにより、実際の過程との関係を知ることができる。
- (3) 知的CAD構築に直接結び付く。

といった利点を持つ。したがって計算可能なモデルこそ知的CADにとって重要である。本稿ではさらに、この(2)の意味での計算可能なモデルに注目する。すなわち設計過程のモデルを計算機システムとして実現して、実際の設計と比べることによりそのモデルの有効性・妥当性をみるというものである。このように計算機によって設計過程を実現することを設計シミュレーションと呼ぶことにする。

今のところ、設計には検証可能な理論が存在するわけではないので、(1)の意味での検討は困難であるので、設計シミュレーションは設計研究にとって重要な研究

手段である。また、知的CADは設計者と協調し対話的に設計を行なう必要がある。この点からみた場合、設計シミュレーションはその対話実験としての意味を持つ。

本稿では以下設計過程のモデルについて議論した後、そのモデルによる設計シミュレーションの実際をみていく。

2. 設計過程の表現

1節で述べたように、設計過程を計算可能なモデルで考えることが必要である。このためにはまず、設計過程を形式化枠組みを用いて表現する必要がある。本稿ではこの方法として論理的な手段を用いることにする。すなわち、仕様や設計対象等を論理式で表現し、その中で設計過程を議論する。まず、論理式集合としての要求仕様をR、得られた設計解をDS、ものの性質や振舞いに関する知識をKとすると、設計解DSは

$$K \cup DS \vdash R \quad (1)$$

を満たすようなものであり、与えられたRからこのDSを求めることが設計であると定義することができる。しかし、実際には要求仕様や利用する知識は設計解時に全て定まっているものではない。むしろ、要求仕様や利用される知識の変化も設計の重要な特徴である。例えば、要求仕様は設計途上で追加・詳細化されたり、あるいは変更されたりする。知識も設計途上で必要に応じて増えたりする。また、実際の設計においては、要求仕様全体や設計対象全体を対象に設計を進めるのではなく、各時点ではそのある部分が考慮の対象になっている。

すなわち実際の設計では、(1)におけるKやDS、Rそのものを扱っているわけではなく、その一部や近傍を対象にしているに過ぎない。逆にいえば、これらを直接扱えないということが設計過程を成立させているといえる。このように考えるとき、設計過程とは、このKやDS、Rに当たる部分を変化させていき、最終的には(1)を満たすようなK、DS、Rを求めるというものである。その途上での $K \cup DS$ をS、RをGと表すと、

S ⊢ G

となる。あるときにはSからGを導くことを行なう（演繹）。あるいはGからSを導くことを行なう（仮説形成、abduction）。あるいはSを変化させる、Gを変化させることが行われる。実際の設計過程はこのようにSとGが順次変化していく過程であるといえる。

さらにこれまで述べてきた中から、以下の部分に注目する。

(1) SとGの相互作用

いかにSからGを導くか、あるいは以下にGからSを導くかについての論理的な方法についての議論。

(2) 変化するSの管理

Sは設計途中で順次変化する。場合によっては分岐する。このようなSの変化を扱う方法に関する議論

(3) 設計知識の適用の過程としての設計

知識（K）がある一定の単位としてあり、その適用がSの変化とみるときの設計過程

なお、各シミュレーションで用いた例は設計実験[1]により収集されたものである。

3. AbCDモデル[3]

ここではSとGの相互作用について考察する。すなわちSからGを導くこととGからSを導くことを考える。SからGを導くことを演繹といい、GからSをつくることを仮説形成（abduction）という。直感的に言えば、{A, A → B} から {B} を導くことが演繹であり、{B, A → B} から {A} を導くことがabductionである。演繹はある一定の制限があれば計算可能であることが判っている。しかし、abductionについてはまだ定まった方法はできていない。ここでは単純に、ある命題が与えられたとき、含意式（以下ルールという）の集合からの後件部とその命題が一致するルールを取り出し、前件部を得ることを仮説形成とする。

この仮説形成と演繹を交互に行ない、Sを変化させる。ルールをひとつつ導入しては演繹を行う。さらに矛盾が生じるときは、サーカムスクリプション[6]によりルールを変形させる。仮説形成、サーカムスクリプション、演繹の循環により、設計過程をモデル化する（AbCDモデル）。

図1にAbCDモデルに基づいて作成したシミュレーション・システムの動作の例を示す。このシステムはあらかじめ用意されたルールを順次取り込んで、Sを順次変化させていくものである。図1のその1サイクルを示す。S₁とG₁であったものが、新しいルールを取

り込んで、S₂とG₂に変化している。

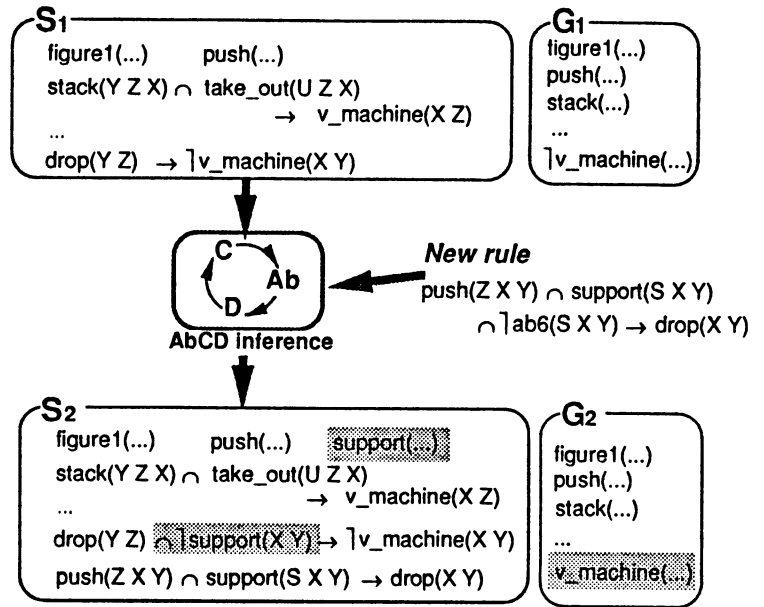


図1 AbCDモデルによる例

4. MTMS[4]

ここではSの変化を管理する方法について考察する。先に述べたように、Sは設計中の設計対象の記述と利用する知識を含んでいる。Sは設計対象が詳細化されるにつれ増加する。しかし、取消ややり直しがあった場合、Sの一部が変更され、新しいSになる。また、設計過程においては複数の解候補が出てくることがある。この場合、Sはいくつかに分岐することになる。

このようなSの変化を様相論理の多重世界の概念を用いて保存・利用することを考える。すなわち、各時点でのSを様相論理の可能世界に対応させ、その世界で成り立っている論理式とする。Sの遷移関係を可能世界間の到達可能性関係で表現する。このとき、最初の仕様のように常に真であるものは必然記号（□）を用いて、設計解になりうるものは可能記号（◇）を用いて表せる。

また、取消ややり直しが起こった場合は、新たな枝別れの世界をつくる。この際のバックトラックにはTMS (Truth Maintenance System) の考え方をを用いて、依存関係をたどってバックトラックを行ない、最適な位置に新しい世界をつくる。

図2は以上のことを実現するために試作したシステム、MTMS (Modal logic based Truth Maintenance System) による例である。各枠が一つの世界を示し、矢印線は世界間の到達可能性関係を示す。ここではある世界で矛盾が生じたとき、バックトラックを行ない、新たな世界を作っている。

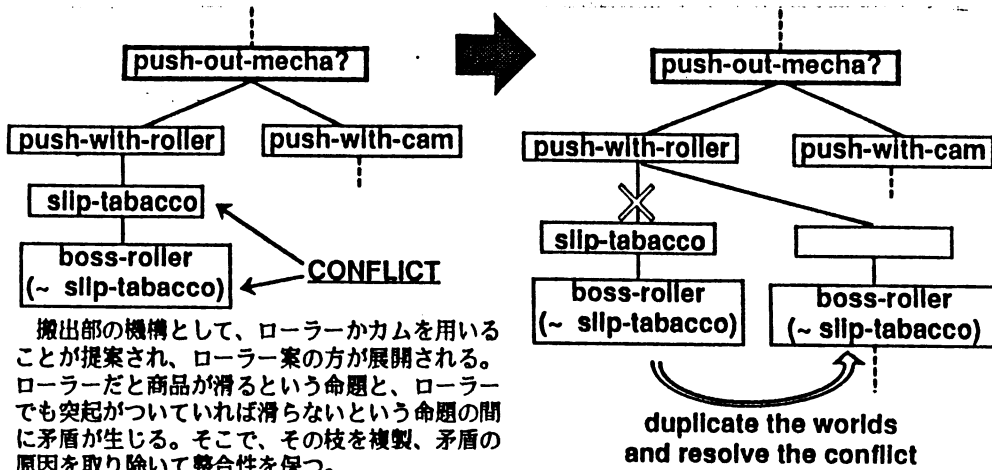


図2 MTMSによる例

5. IDDL [7][8]

手順等の知識が知識ベースとして用意されているとき、知識ベースの適用によるSの変化をIDDLによってシミュレートする。IDDL (Integrated Data Description Language) は設計知識を記述するために開発されたものであり、このなかでは設計対象はオブジェクトとして、設計知識はシナリオとして記述される。オブジェクトは設計途中ではワールドと呼ばれるデータベースの中におかれる。シナリオはルールの集合であり、関連した知識の集合または設計手順を示す。シナリオはワールドに対して適用され、その中のオブジェクトを変化させる。

図3はIDDLの実行例を図示したものである。ここでは3つのシナリオが順次ワールドに適用され、ワールドの内容(object)を変えている。また、途中でワールドを二つに分けて、別々の設計を行っている。

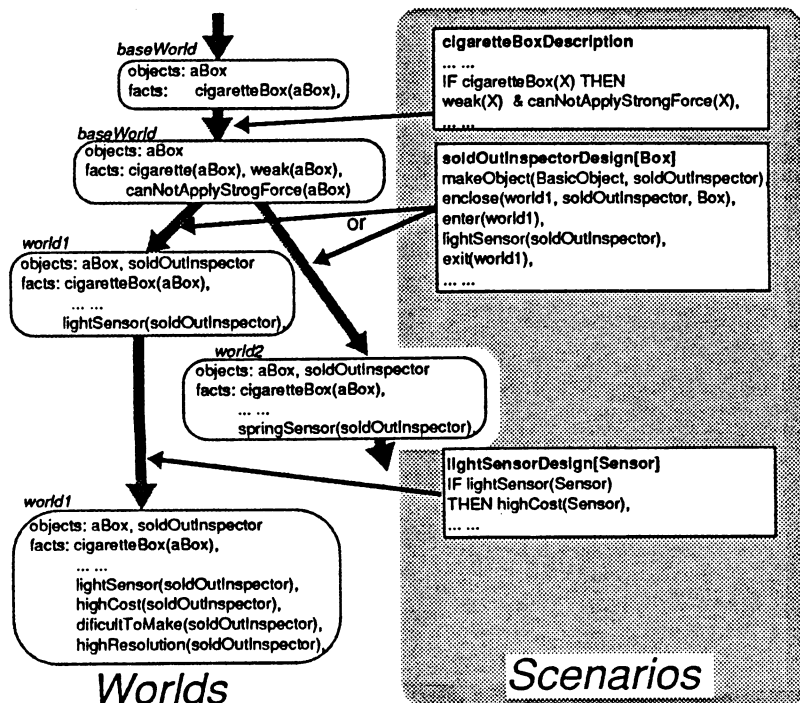


図3 IDDLによる例

6. まとめ

前節までで設計シミュレーションの3つの例をみてきた。各々のモデルはそれぞれ設計過程のある特徴に注目し、その特徴を自然に表現することが目標であった。これを確認するために、設計実験のデータをもとにシミュレーションを行ない、

設計者の設計過程との対応をみた。例えば、AbCDモデルでは設計者の思い付きとその確認という点の表現がひとつの目標であった。このモデルによるシミュレーションは、設計者のプロトコルでの思い付きとその確認の部分でよく似た設計過程を実現し、AbCDモデルがこの点をよく説明しているといえる。このようにして、各モデルは設計過程のシミュレーションを行うことで、その有効性を確認した。

今後はこれらのモデルを統合してより他面的な設計過程のモデルを考察し、さらに設計過程のシミュレーションを行なう必要がある。

参考文献

- [1] 武田、勢川、富山、吉川：設計過程の分析と論理による形式化(第1報) - 設計実験の分析 - 昭和63年度精密工学会春期大会講演論文集(1988)131-132
- [2] 武田、石原、林、富山、吉川：設計過程の分析と論理による形式化(第2報) - 設計知識の分析 - 1989年度精密工学会春期大会講演論文集(1989)5-6
- [3] 林、武田、富山、吉川：設計過程の分析と論理による形式化(第3報) - サーカムスクリプションとアブダクションによるモデル化 - 1989年度精密工学会春期大会講演論文集(1989)7-8
- [4] 河合、武田、林、富山、吉川：設計過程の分析と論理を利用したTMS - 1989年度精密工学会春期大会講演論文集(1989)9-10
- [5] J.R.Dixon: "On the research methodology towards a scientific theory of engineering design", AIEDAM, 1(3), 1987, pp.145-157
- [6] J. McCarthy: Circumscription, - A form of nonmonotonic reasoning, Artificial Intelligence Vol 13, pp. 27-39 (1980)
- [7] 富山、薛、吉川：設計知識表現言語IDDLの開発(第1報) - 言語仕様 - 1989年度精密工学会春期大会講演論文集(1989)37-38
- [8] 薛、富山、吉川：設計知識表現言語IDDLの開発(第2報) - プロトタイプの開発 - 1989年度精密工学会春期大会講演論文集(1989)39-40