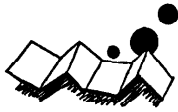


解説

設計エキスパートシステム†



長澤 勲†

1. ま え が き

近年、製品設計の高度化、効率化、省力化、低コスト化に対する要求が多く、製造業において強まってきており、CAD (Computer Aided Design) 技術の役割はますます重要なものとなっている。ところが、CAD のための一般性のある方法論が確立されておらず、CAD システムの開発や保守に困難を生じている。この基本的な理由は設計作業の多様性や複雑性によってその体系的理解がなされていないことや CAD システムを構築するための汎用性のあるプログラミング概念が未成熟であることによる。このような状況において知識工学的方法は設計者の設計過程の詳細な分析や、汎用性のある CAD システム構築のための有力な方法として注目を集めている。本稿では CAD において知識工学が果たすべき役割、設計知識と設計過程の範例、設計エキスパートシステムの例について述べる。

2. CAD の発展と次世代 CAD の課題

2.1 CAD の発展

計算機を設計支援に用いるという着想は MIT の CAD Project¹⁾ に始まるといわれている。この計画において掲げられた目標は、(1)人間と計算機の対話による計算機の利用 (2)図形を媒介にした計算機との対話 (3)計算機支援によるシミュレーションの三つに要約され CAD の概念は明確にされた。なかでも、グラフィック・ディスプレイとライトペンを用いて計算機との対話を試みた SKETCHPAD²⁾ は重要ないくつかの概念を提案しており、後の発展の基礎となった。しかしながら、この時期は計算機のハードウェア、ソフトウェア技術ともに未発達であり具体的な成果を得るには至らなかった。

1960 年代中期以後、計算機技術全体の発達にともない具体的な成果が現れ始めた。最初にトランスやポンプのように設計作業がルーチン化されており、標準化が進んでいる製品を自動設計するシステムがつけられた。これらは従来、図面などを用いて行われていたルーチン化された設計業務を直接プログラム化したもので、問題向設計システム³⁾ と呼ばれている。この種のシステムは実用上大きな成果をあげはしたが

- (1) 方法論の基礎付けがなく、あまりにも個別的であり一般化できない。
- (2) 通常 FORTRAN などの手続き型言語で書かれており、大規模であるので硬直化しやすく保守が困難である。

という問題点を残している。一方、設計評価に用いられる設計案の解析には、より汎用性の高いアプローチがとられた。電子回路解析^{4),5)}、ラーメン・トラスの構造解析システム⁶⁾ などである。これらのシステムの特徴は、対象の構造が比較的簡単であること、また解析に必要な基礎理論が完備していたことである。

計算機による図形処理技術はコンピュータ・グラフィックスとして独自に発展した。なかでも、対話図形

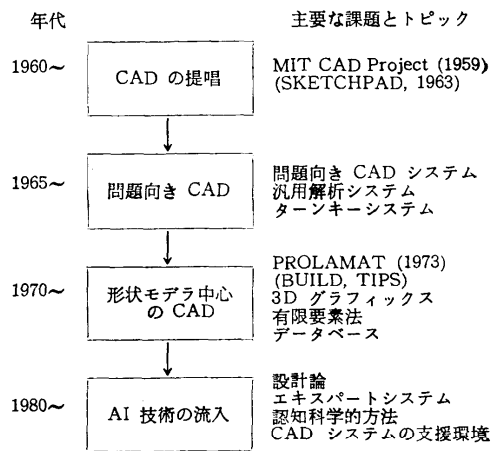


図-1 CAD の発展

† Design Expert System by Isao NAGASAWA (Computation Center, Kyushu University).

†† 九州大学中央計数施設

形処理, 3次元図形処理技術の発展は設計支援に多大の貢献をしている。最初に実用化された図形処理技術はワイヤフレーム(針金細工の表現)処理を基本とした製図支援システムである。これらは, 一般に2次元あるいは3次元空間に与えた図形要素に拡大・縮小・回転などの操作を行い, 組み合わせて複雑な図形を記述するシステムであり, ターンキーシステムと呼ばれている。ターンキーシステムは MIT の CAD Project の2番目の目標をほぼ完成させた技術となったが, まだ形状を取り扱うシステムとしては十分ではなかった。

1970年代にはいと3次元立体形状の計算機モデルとして BUILD⁷⁾, TIPS-1⁸⁾などのソリッドモデルが提案された。これらのシステムのモデル化の原理はプリミティブと呼ばれる立方体や円柱からなる基本要素を集合演算的に組み合わせて一般の立体を表現する方法を用いている。ソリッドモデルの概念により立体形状を計算機中に表現する手法が確立され, 最近では CAD/CAM (Computer Aided Design/Computer Aided Manufacturing) システムの中核として位置付けられ, 次第に実用化されている。

2.2 次世代の CAD の課題

CAD は, この 20 数年間に長足の進歩を遂げてきた。しかしながら, LIS CAD や形状モデラと汎用解析システムによって構成されたシミュレーション型 CAD⁹⁾ など一部における高度な発展にもかかわらず, CAD システムのための一般性のある方法論が確立されておらず CAD システムの開発や保守に困難を生じている。次世代 CAD の主要な目標は

(1) 設計危機への対応…設計支援の質を高度化し, 設計者の負担を軽減すること。CAD の理想は設計者の有能な助手であるといわれている。

(2) ソフトウェア危機への対応…CAD システム開発・保守のための一般性のある方法論を確立すること。

である。このような目標に到達するために解決しなければならない課題を以下に列挙する。

(1) 設計論の展開: 設計者の設計行為は大雑把に言えば次のようなものである。機械設計を例にとれば, まず, 概念設計に始まり, 構造・機構の設計, 形状・諸元の決定と次第に詳細化を進めていく。この設計作業の各段階において設計案を解析・評価することにより, 性能・信頼性・使いやすさ・経済性・加工性などの各項目の調和を図るために何度も設計案の変更

を繰り返している。設計論^{10)~17)}の主要な目標はこの設計者の行為を統一的に理解し, CAD システム開発の基礎付けをすることである。

(2) 設計過程のモデル化: CAD システムの目的は設計者の設計行為の部分的自動化であるという立場がある。しかし, 知的な程度の高い CAD システムを構築するには, 設計者の設計行為そのものを計算機向きに適切にモデル化する必要がある。

(3) 設計対象のモデル化: 従来工学諸分野では, 設計対象の動作や性質を解析するために解析モデルを使用してきた。解析モデルの特徴は設計対象のある特定の注目している性質についてのみ簡潔に正確に表現することである。これに対して設計者が設計過程で使用するモデルは一般に多面的である。工業製品を例にとれば, 機能, 強度, 形状, 組み立て構造, 加工条件などを統一的に表現できる必要がある。

(4) 設計システムのソフトウェア支援環境: CAD システムは上の(2), (3)のように多くの側面を持つ複雑な知識を取り扱う必要に迫られてくる。このように十分構造化されていない知識を試行錯誤しながら整理し, 体系化していくのに適したプログラミング概念の開発およびそのソフトウェア支援環境としての実現が必要である。

3. CAD の方法論

2章では, CAD の発展を概観し, CAD の課題について述べたが, 次にそのための方法論として現在提案されているいくつかのアプローチについてふれる。ただし, このような試みは始まったばかりであり, ここで与える分類は便宜的なものである。なお, 設計対象のモデル化という立場からは伊藤の解説¹⁸⁾がある。

3.1 公理論的方法

述語論理や集合論など数学的基礎の明確な体系をモデル化の枠組としようとするものである。このアプローチは設計過程一般の理解と CAD の基礎付けを主なねらいとしており, 個々の設計問題には立ち入らず一般的議論を展開している。吉川は公理的集合論を基礎として設計過程一般の記述を試み, 一般設計学^{12), 13)}を展開した。柳生は, 一階の述語論理と集合論によって設計対象のモデル化を行うことを提唱し^{16), 17)}, データベースの形式的操作言語 LMS (Logical Structure Manipulation) を開発した。大須賀は, 一階述語論理を対象物の構造記述向きに拡張した多層論理 MLL (Multi Layer Logic)^{19), 20)}を提案した。

3.2 設計対象モデリング

解析や設計結果の表現に用いられてきた解析・表示モデルを設計対象のモデルとして発展させようとするものである。機械設計の例では、木村らのプロダクトモデル^{21),22)}、沖野の O/O (オブジェクト・オペレーション) 二元論²³⁾、中島らの FD (Feature Description)²⁴⁾ がある。Geometric Reasoning²⁵⁾ や論理装置設計における記号シミュレーション²⁶⁾ もこのような研究の流れの一部と考えることもできよう。筆者らは設計過程では、寸法や材質などの属性や構造の詳細が未定であるモデルを取り扱う必要があることを指摘し、このようなモデルを記述するための汎用言語 ADL^{27),28)} を開発した。伊藤は認知科学的な立場から研究を行い、設計者が持っていると考えられる機械部品のメンタルモデルの表現をねらいとした形状モデラ CONMOTO²⁹⁾ を提案した。

3.3 アルゴリズム論的方法

現在現場で実用に供されている多くの設計システムは対象領域固有の理論を基礎に FORTRAN などの手続き型言語で開発されている。たとえば LSI 設計におけるフロアプラン、プリント基板上の部品の配置・配線システム、板金部品の板取システムなどである。この種の設計システムの特徴は考慮すべき設計要件が比較的少なく、技術の発展にともなう設計知識の変化が少ないこと、解くべき問題のサイズ (計算量) が比較的大きいことである。このため効率のよいアルゴリズムの開発が主要な研究課題となっている。ところが、近年 VLSI 設計などでは設計問題が著しく大規模化しつつあり、従来のアルゴリズム的方法だけで CAD システムを構築することは困難になりつつある³⁰⁾。

3.4 知識工学的的方法

知識工学の分野で発展した知識ベースシステムあるいはエキスパートシステムの方法を用いて設計者の設計行為自体を計算機上にモデル化しようとするものである。知識工学的的方法を用いる研究の動機はさまざまである。論理装置設計では、大規模な設計問題の解法を熟練設計者が経験的に獲得している設計知識に求める研究³¹⁾⁻³⁷⁾ がある。機械設計では多様な設計問題を汎用的な枠組で記述する研究³⁸⁾⁻⁴⁰⁾ や概念設計へ応用する研究^{41),42)} がある。

4. 設計過程のモデル

知識工学的的方法を用いて CAD システムを構築する

には、設計過程のモデル化が最も基本的な課題となる。しかし、設計過程は対象分野固有の方法論^{30),43)-46)} との関係もあり、一般的な議論は容易でない。ここでは比較的よく研究されている論理装置設計と機械設計の分野の話題を概観し、設計論と知識工学的的方法との関係について述べる。

4.1 設計論

設計論の研究はすでに 20 数年の歴史があり、建築や機械の分野で個別的に発展してきた¹⁰⁾。この中で一般性があると考えられる最初の理論は Mitchel による次のような提案者・検証者 (Generator-Filter) モデル¹¹⁾ である。

『議論の対象領域は構成可能な人工物すべてからなる集合である。与えられた設計要求を満足する人工物を次のような手順で求める。まず、提案者は実際に作成できる程度に詳細な一つの設計案を提案する。検証者は提案された設計案を解析し、設計要求を満足しているかどうかを検証する。もし満足していればそれは一つの解である。もし満足していなければ提案者に次の設計案を求める。』

このモデルは分野によらない一般性の高いものであったが、設計要求や設計案をどのように表現すべきかという表現問題には立ち入っていなかった。この点について吉川は人工物の集合はその属性によって表現できることを指摘し、一般設計学¹²⁾を展開した。概略は次のようなものである。

『属性についての命題 (たとえば、色が白である。重さが 10 グラム以下である。) は人工物のある部分集合を指定する。これらの論理結合もまた人工物のある部分集合を指定する。すなわち、属性は人工物の集合における位相である。設計要求は通常、人工物の機能を表す属性の集合 S によって、設計案は人工物の構造を表す属性の集合 P によって表される。したがって、設計とは人工物のすべてからなる集合上の異なる位相間の写像: $S \rightarrow P$ である。』

この吉川の設計論は設計問題とは何であるのかを理解しやすくするのに効果的であった。しかし、実際にはこの機能から構造への写像が容易に行えないというところに問題がある。吉川は設計過程の範例として次の 4 つをあげている。

(1) 全数対応モデル: 有限個の異なる設計要求と設計解との間に 1 対 1 の対応関係が存在する場合。この場合、設計問題は設計資料の検索問題に還元される。

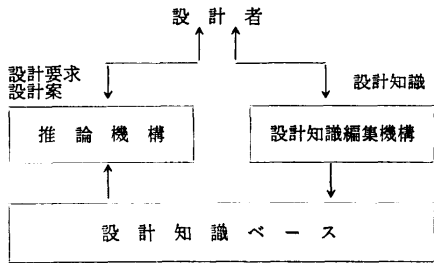


図-2 知識ベース型設計システム概念図

(2) 計算モデル：設計要求と設計解を指定する属性がそれぞれ数値で表現されており、前者の属性と後者の属性との間に数学的依存関係が存在する場合。この場合後者の属性は、前者の属性から直接計算されるか、あるいは最適化手法によって求められる。

(3) 生成モデル：(2)における属性計算を数値属性に限定せず生成規則を用いて一般化したもの。この場合、設計要求から設計解への変換は段階的に行われ、変換の中間段階をメタモデルと呼ぶ。

(4) 範例モデル：範例モデルは設計案をかなり自由に修正できる提案者（通常、人間）を仮定している。設計要求と設計案の間に矛盾が発見されれば提案者は設計案を設計要求に近付くように修正する。

設計の一般論に対しておおむねこの分類は妥当であろう。しかし、CADシステムを構築する立場ではこれらの設計過程をさらに詳細にモデル化する必要がある。

4.2 知識工学的的方法

知識ベース型設計システムの概念図を図-2に示す。設計知識は設計者に理解しやすい形で設計知識ベース

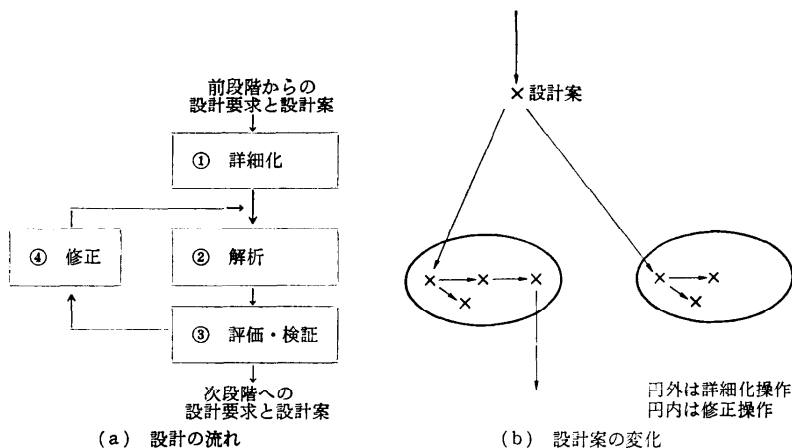
知識の分類	内容 (括弧内は一例)
設計対象知識 (設計対象に関する知識)	① 設計対象に関する拘束条件知識 (設計公式)
	② 設計対象の機能と構造に関する知識 (機構学)
	③ 設計対象の性能やコスト評価に関する知識 (評価関数)
設計操作知識 (設計案の操作)	④ 設計要求から設計案への詳細化を行う操作 (設計カタログ)
	⑤ 設計案を修正する操作
設計制御知識 (設計解の探索を行う知識)	⑥ 設計者が経験的に獲得した探索戦略
	⑦ 設計計算の順序を制御する探索戦略
	⑧ 設計評価尺度を利用した探索戦略

図-3 設計知識の分類

に貯えられる。この設計知識ベースは技術の発展にともない常に開発や保守を行う必要があるため、設計者は設計システムの開発や保守も分担する。設計者は設計システムの開発や保守に際して、この設計知識ベースを設計知識編集機構を用いて編集する。設計問題を解く機能は設計知識ベースと推論機構の組み合わせによって実現される。

4.2.1 設計知識の分類

機械やLSIのように複雑な構造物の設計では、複数の設計段階を設定し、抽象的な設計要求から出発し、次第に構造や構造諸元の詳細化を行うのが普通である。この各段階において、設計案を解析、評価することにより何度も設計案の修正を繰り返している。この設計過程において使用される知識は図-3のように分類することができる。設計対象知識は設計対象に関する知識であり、設計案の解析や評価に用いられる。設計操作知識は設計案の詳細化や修正を実行する知識で



(a) 設計の流れ

(b) 設計案の変化

図-4 設計過程

ある。設計制御知識は設計過程の全体の制御に用いられる知識である。この知識は設計の効率を決定する。

4.2.2 設計過程

設計過程は、設計問題によっても異なるが、およそ図-4のように表現できる。設計過程の入力は前段階からの設計要求と設計案である。ここで設計要求とは設計解が持たなければならない機能仕様と満足しなければならない拘束条件の集合、設計案とは設計の部分的な解と考えてよい。この設計過程のモデルは、使用する設計知識や設計問題の性質によって、拘束条件解法理論 (Constraint Solving Theory) と問題解決理論 (Problem Solving Theory) に分類できる。これらはそれぞれ、吉川の生成モデルと範例モデルに対応している。

拘束条件解法理論^{38)~40), 47), 48)}

拘束条件解法理論は、“設計問題は近似的に分解可能な問題として取り扱える”。というやや楽観的な立場をとっている。すなわち、設計問題は設計解の構造を仮定することによって、あるいはいくつかの部分問題に

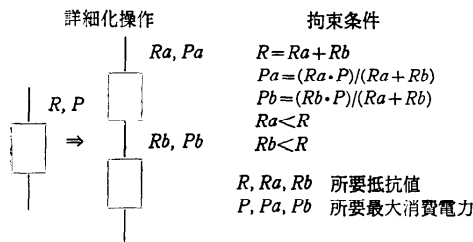
分解され、部分問題間の関係はパラメータおよびパラメータ間の拘束関係によって陽に記述可能であると考えられる。設計問題を解くとは、設計解の構造と構造の持つパラメータを決定することである。

図-4において設計問題は次のように解かれる。まず、①抽象度の高い設計要求からはじめて、設計案の詳細化操作を行い、いくつかの部分問題に分割する。この分割において部分問題間の関係はあらかじめ定められており、パラメータと拘束条件によって記述されている。また、この詳細化操作は設計解が得られるとは限らないため一般には仮定的に行われる。②パラメータ間の拘束関係を解くことにより詳細化の時点では不明であったパラメータ値の決定を行う。③拘束条件を評価することにより、設計案が妥当であるかどうかの検証を行う。(i)設計案の検証が成功あるいは検証を保留する場合には次の段階に進む。(ii)検証の結果が失敗である場合には設計案の代案を求めて①へ後戻りする。④の設計案の修正は行わない。

拘束条件解法理論の概念は最初、遺伝子組み替え実

「カタログに与えられた抵抗器を用いて、合成抵抗値 10Ω 最大消費電力 40 W の直並列回路を設計せよ。ただし、直並列の入れ子の深さを最大2とする。」

(a) 設計問題

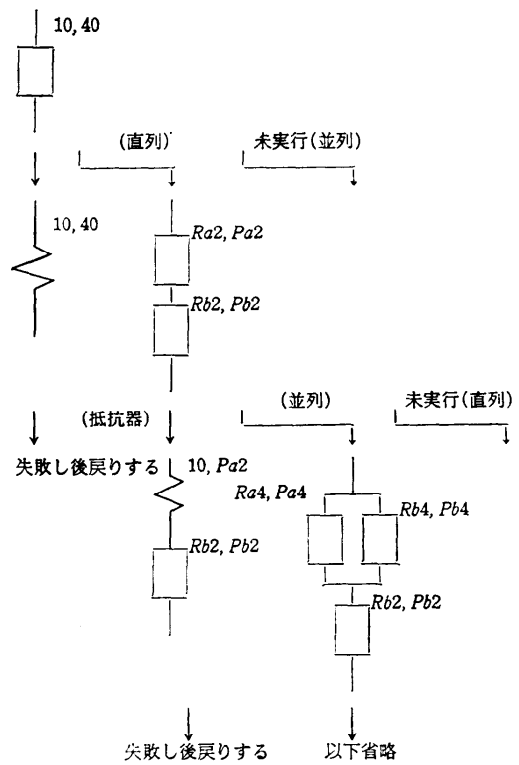


(b) 回路構造の詳細化操作と拘束条件の例

```

refine (?D, R, P, Cir):-
    !! member (T, [r, s, p])
    // box (D, T, R, P, Cir).
box (?D, ?T, R, P, s (Ea, Eb)):-
    T=s
    !! D ≥ 1, Di is D-1
    // refine (Di, Ra, Pa, Ea),
    refine (Di, Rb, Pb, Eb, ),
    Ra + Rb == R,
    Pa is (Ra * P) / (Ra + Rb),
    Pb is (Rb * P) / (Ra + Rb),
    Ra < R,
    Rb < R,
    
```

(c) 設計プログラムの一部



(d) 探索の概要

図-5 簡単な電気回路の設計

設計システム MOLGEN⁴⁷⁾ によって提案された。このシステムは拘束条件の伝播、仮定生成と検証、最小決定の原則など基本的な概念を提案した。しかし、システムとして成功しているとはいえない。拘束条件解法理論は設計対象の構成要素間の拘束が強く構造の可変性が少ない設計問題に適している。このような設計問題には

(1) 設計対象は、それがあらかじめ使用される環境が指定されていなければ設計に使用するモデルが特定できない。(設計対象モデルの環境への従属性)

(2) 構成要素間の強い拘束のため設計案を部分的に修正する修正操作が困難である。

という特徴がある。筆者らは論理プログラミングを基礎に、この理論を拘束条件リダクション法 (The Method of Constraint Reduction) と呼ぶプログラミング概念に発展させ^{27), 28)}、機械設計のうち基本設計に応用した^{39), 40)}。図-5 に簡単な電気回路を設計する例を示す。

問題解決理論^{31)~36), 49)~51)}

問題解決理論は人工知能における問題解決法を用いて設計者が設計案を動的に修正する過程をシミュレートしようとするものである。この方法は設計要求に対して必要な構造を試行錯誤しながら構成する構造の設計に適しており、複雑な機能を持つ対象の設計には不可欠な手法である。設計問題の複雑さの程度によってさまざまな範例がある。

〔詳細化の制御〕

計算機機器構成システム R1⁵¹⁾ は Match 法と呼ぶ手法を提案している。Match 法とは前段階からの設計案と未解決の設計要求との関係を分析し、次に実行すべき詳細化操作を決定する方法である。この方法は設計解の探索の必要性を大幅に減少できる特徴がある。

〔局所的修正による設計案の最適化〕

論理回路の合成を行う SOCRATES³¹⁾ や Darringer³²⁾ らのシステムは局所的な設計案の修正操作を繰り返し、大規模な論理回路を最適化している。この種のシステムは設計案の修正後の状態を先読み (look-ahead) 評価することにより修正操作を制御している。

〔大局的設計知識の利用〕

論理合成システム DE^{33)~36)} は詳細化操作と修正操作を組み合わせるにより、設計者が長年の経験から獲得している大局的な設計知識の定式化を行っている。DE の方法は、まず詳細化操作によって設計解の粗な近似を与え、次にこれを修正操作によって高品質

な設計解に近付ける方法である。

5. 設計エキスパートシステムの例

設計エキスパートシステムの例として特色のある、R1、機械設計システム、REDESIGN、SOCRATES、DE について説明する。

R1⁵¹⁾

Carnegie-Mellon 大学で開発され、DEC 社で使用されている計算機機器構成システムである。R1 は計算機のメモリ、CPU、ディスクなどの主要構成機器が与えられたとき、電源、コネクタなど付属機器を選定し、キャビネット内に自動配置する。設計作業の内容は高度ではないが実用化された数少ないシステムの一つである。R1 は汎用プロダクションシステム OPS4 によって実現されている。システムはデータベース、ルールベース、作業記憶からなる。データベースには計算機の構成機器についての知識、ルールベースには設計を実行するルールを記述する。作業記憶には、設計要求である構成機器のリスト、設計途中の部分的に構成された機器構成、設計計算に使用する補助情報、設計のステージを表す制御情報が格納される。作業記憶の初期状態は空である。作業記憶に主要機器構成リストを記入すると、注文の誤り検査、cpu と cpu 拡張キャビネット内の機器配置、unibus 拡張キャビネットとボックス内の機器配置、unibus 拡張キャビネットのパネル取り付け、床配置、配線の順に設計が行われる。

R1 は Match 法と呼ばれる方法を用いて設計の途中で後戻りすることなく効率よく設計を行っている。実用的な問題を数分で解いたと報告している。

機械設計支援システム^{39)~40)}

筆者らのグループが開発した、機械設計のうち基本設計の支援を行うシステムである。基本設計の特徴は、(1)設計対象の機械の構造は設計カタログなどに与えられていることが多く、比較的少数の候補の中から選定すればよい。(2)性能、信頼性、寿命、加工性などの各項目の調和を図るため、機械の構成要素のパラメータ (材質、加工法、寸法、精度など) を決定する設計計算を試行錯誤的に繰り返す。(3)(2)の設計計算に必要な資料は設計公式や設計資料として整備されていることである。筆者らはこのような設計問題の記述には拘束条件解法理論が最適であると考え、設計システム記述言語 ADL^{27), 28)} を開発した。このように次の3種類の設計知識を使用して、歯車減速機や

Vベルトの設計システムを構築した。

- (1) 機械や構成要素が満足すべき拘束条件 (設計公式, 設計資料, 動作特性式)
- (2) 機能から構造への詳細化を行う詳細化操作 (設計カタログ)
- (3) 設計計算の大略の流れを制御する知識 (優先順位制御)

実験の結果, 数百の拘束条件を持つ設計問題を数分で解いている。このシステムの特徴は, (1)設計対象のモデル表現に適した拘束条件集合を基本概念として採用している。(2)設計問題を近似的に分解可能な問題として定式化しているため, システムのモジュール性がよい。(3)設計知識の大部分はカタログや規格などの公的知識に求めることができる。(4)単純で見通しのよい推論機構でさまざまな設計計算法を表現できる。(5)設計知識をその役割ごとに明確に区別して表現できる。(6)設計制御知識の表現力が十分でなく, 大きな設計問題の自動化に適していない。

REDESIGN⁵²⁾

Rutgers 大学で開発された, 論理回路の機能変更の際して再設計を会話的に支援するシステムである。本システムでは回路の構造を回路モジュールとデータパスからなるネットワークとして表現している。回路モジュールは階層構造をなし, これらには機能仕様

(specification), 動作 (behavior), および使用目的 (purpose) という概念が定義されている。回路モジュールの動作は因果推論によって, 使用目的の決定は目的推論によって行われる。このシステムで重要なのは Design Plan という概念である。これはある回路モジュールの仕様がどのように分割され実現されているかを示すデータ構造である。図-6 に CGM (Character Generator Module) の Design Plan を示す。図中→は設計におけるモジュール実現法を示す。たとえば USE-ROM は “入出力間に, ある有限な対応付けを実現するには適当な ROM を用いよ。” という詳細化ルールを適用したことを示す。また→→はある詳細化の選択から発生する矛盾を示す。システムは既設計の回路図, 機能仕様, この二つを関係付ける Design Plan が入力されたとき次の順に再設計をガイドする。
 (1)機能変更に関係する回路モジュールを指定する。
 (2)回路モジュールの再設計案を提案する。(3)再設計にともなう副作用を指定し改修する。設計意図を陽に表現し, 再設計に使用するシステムである。基本的なアイデアは拘束条件解法理論と似ている。

SOCRATES⁵¹⁾

General Electric で開発された組み合わせ論理の最適化を行うシステムである。SOCRATES はゲートアレイやセルライブラリなどで与えられた特定の実現技術に対して最適化を行うことができる。図-7 に示すように SOCRATES は組み合わせ論理をブール (Boolean) 式と回路ネットワーク (Netlist) の二つの方法で表現している。最適化の手順は, (1)数学的手法を用いたブール式の最適化, (2)ブール式の回路ネットワークへの変換, (3)局所変換の反復による回路ネットワークの最適化である。(3)の部分は任意の実現技術に対応するため, データ (最適化の対象となる回路ネットワーク), ルール (回路の変換操作), メタルール (回路の変換操作の適用を制御するルール) からなるルールベースシステムとして実現されている。回路の変換は一意的でないため変換の良さを評価する必要があり, この評価尺度には回路を LSI として実現したときの面積と速度を使用している。最適な変換を発見する戦略

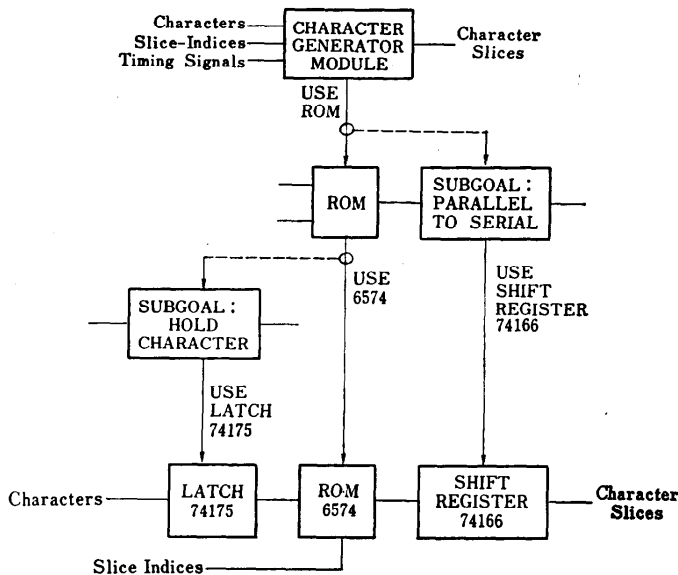


図-6 Character Generator Module の Design Plan.

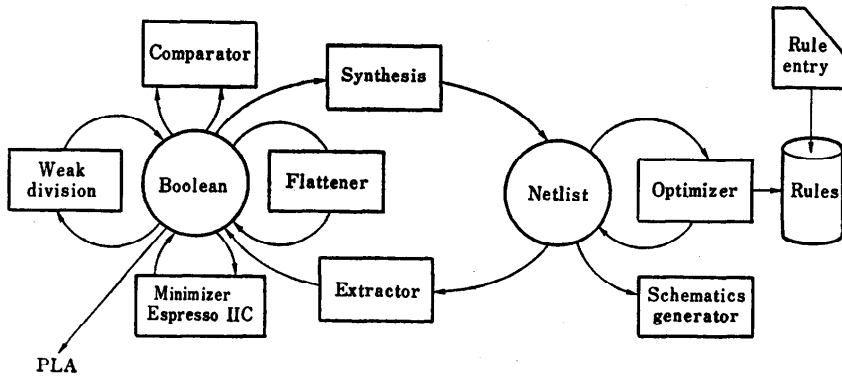


図-7 SOCRATES システムの構成

として、変換後の回路を先読み (look-ahead) 評価する。この先読みの範囲はパラメタ (先読みの深さ、広さ、隣合うゲートの範囲) によって制御でき、メタルルールは最適化のフェーズに対応してこのパラメタを制御する。実験の結果、熟練設計者と同程度の最適化能力があったと報告している。このシステムは設計制御知識の取り扱いに特色がある。

DE³³⁾~36)

武蔵野通研が開発中のレジスタ転送レベルの動作仕様から論理合成を行うシステムである。このシステムのねらいは、設計者が長年の経験から獲得している大局的な設計知識を活用することによって、従来のアルゴリズム的方法では困難であった高品質な設計を効率的に実行することである。DE ではデータベース設計、制御回路設計、ALU 設計、データベースの検証・補正などが試みられている。実験の結果、熟練設計者と同程度の高品質の設計結果であったと報告している。熟練設計者の設計手順の定式化を行った例である。

6. むすび

本稿では汎用的な CAD システムを構築するという立場から CAD に対するさまざまなアプローチと知識工学との関係について述べた。しかし、CAD は現実的な設計計算や設計結果の表現問題から人工知能研究にとっても本質的な課題である設計者の創造性に至るまで幅広いテーマと関連しているため、一面的な見解にすぎないことをおことわりしておく。

本稿ではふれることができなかったが、プロダクションシステム⁵³⁾、拘束条件言語^{27), 28), 54), 55)}、知識表現言語⁵⁶⁾⁻⁵⁸⁾などの設計システム記述用言語、設計と検証・診断の関係⁵⁹⁾⁻⁶¹⁾、設計用データベース⁶²⁾なども CAD にとって重要な課題である。また、CAD に対す

る設計現場のニーズについては調査資料^{63), 64)}がある。

設計者の知的助手 (Clever Assistant Designer)* を目指した研究は始まったばかりといえる。今後の研究が期待される。末筆ではあるが資料を提供して頂いた方々に感謝する。

参考文献

- 1) Mann, R. W. : The CAD Project, Mechanical Engineering (ASME), p. 41 (1965).
- 2) Sutherland, I. : Sketchpad : A Man-Machine Graphical Communication System, Proc. SJCC, p. 329 (1963).
- 3) 沖野教郎 : 自動設計の方法論, 養賢堂, p. 192 (1982).
- 4) Malmberg, A. F., Cornwell, F. L. and Hofer, F. N. : NET-1 : Network Analysis Program, Report LA-31119, Los Alamos Scientific Laboratory, Los Alamos, N. M. (1964).
- 5) Jensen, R. W. and Lieberman, M. D. : IBM Electronic Circuit Analysis Program, Prentice Hall (1968).
- 6) STRESS : Structured Engineering System Solver, A User's Manual, MIT Press (1965).
- 7) Braid, I. C. and Lang, C. A. : Computer-Aided Design of Mechanical Components with Volume Building Bricks, proc. PROLAMAT '73 (1973).
- 8) Okino, N., Kakazu, Y. and Kubo, H. : TIPS-1 : Technical Information Processing System for Computer Aided Design, Drawing and Manufacturing, Proc. PROLAMAT '73 (1973).
- 9) 沖野教郎 : CAD の研究, 情報処理, Vol. 24, No. 1, pp. 11-16 (Jan. 1983).
- 10) Spillers, W. (ed.) : Basic Questions of Design Theory, North-Holland (1974).

* 東京大学精密工学教室 木村文彦助教授による。

- 11) Mitchel, W.: Computer Aided Architectural Design, Van Nostrand Reinhold (1977).
- 12) Yoshikawa, H.: General Design Theory and A CAD System; in Sata, T. and Warman, E. (eds). Man Machine Communication in CAD/CAM, proc. IFIP WG 5.2/5.3 Working Conference (Tokyo), North Holland, Amsterdam, p.35 (1981).
- 13) Tomiyama, T. and Yoshikawa, H.: Extended General Design Theory., IFIP WG5.2 Working Conference on Design Theory for CAD (Tokyo), Preprints, pp.75-104 (1985).
- 14) Bijl, A.: An Approach to Design Theory., IFIP WG5.2 Working Conference on Design Theory for CAD (Tokyo), Preprints, pp.1-23 (1985).
- 15) Takala, T.: Theoretical Framework for Computer Aided Innovative Design. IFIP WG5.2 Working Conference on Design Theory for CAD (Tokyo). Preprints, pp. 251-260 (1985).
- 16) 柳生孝昭: CAD のための data metamodel, 情報処理学会グラフィックスと CAD シンポジウム論文集, pp.205-213 (1983).
- 17) 柳生孝昭: 形式的・数学的体系としての CAD, 第4回設計自動化学講演会講演論文集, pp. 1-11 (1986).
- 18) 伊藤公俊: 対象の計算機モデリングと設計, 日本ロボット学会誌, 第4巻, 4号, pp. 101-108 (1986).
- 19) Ohsuga, S.: A New Method of Model Description-Use of Knowledge Base and Inference, CAD System Frame Work, K.Bo and Lillehagen, F.M. (eds). North-Holland Pub, Co., IFIP, pp. 285-312 (1983).
- 20) 山内平行, 大須賀節雄: 知識ベースシステム KAUS による構造体表現と推論方式, 電子通信学会技術報告 AL82-69 (1982).
- 21) Kimura, F., Sat, T. and Hosaka, M.: Integration of Design and Manufacturing Activities Based on Object Modelling, Advances in CAD/CAM, North Holland, pp.375-385 (1982).
- 22) Kimura, F., Kawabe, S., Sata, T. and Hosaka, M.: A Study on Product Modelling for Integration of CAD/CAM, Int. J. of Computers in Industry, North Holland, Vol. 5, No.3. pp. 239-252, (1984).
- 23) 沖野教郎: オブジェクト・オペレーション二元論に基づく設計論, 精密工学会, 春季大会シンポジウム資料, pp.18-23 (1986).
- 24) 石田智利, 箕輪元, 中島尚正: 設計対象物の非要求機能検出に関する研究, 精密工学会誌, 第52巻, 2号, pp.106-112 (1986).
- 25) Arbab, F. and Wing, J.M.: Geometric Reasoning: A New Paradigm for Processing Geometric Information, IFIP WG, 5.2/5.3 Working Conference (Tokyo) Preprints, pp.107-121. (1985).
- 26) 斎藤隆夫, 上原貴夫: 論理回路の記号シミュレーション, 情報処理, Vol.26. No.1, pp. 7-16 (Jan.1985).
- 27) 長澤 勲, 古川由美子, 荒牧重登: 論理プログラミングを基礎とした設計システム記述言語 ADL, 情報処理学会論文誌, Vol. 25, No. 4, pp. 606-613 (1984).
- 28) 長澤 勲: ADL II: Prolog を核とした知識表現言語, 九州大学大型計算機センタマニュアル (1985).
- 29) Ito, M., Kono, M. and Hayashi, K.: CONMOTO: A Machine Part Design System based on Designer's Mental Processes. IFIP WG5.2 Working Conference on Design Theory for CAD (Tokyo). Preprints, pp.123-152 (1985).
- 30) 村井真一, 南谷 崇 (編): 大特集: 論理装置 CAD の最近の動向, 情報処理, Vol. 25, No.10, pp.1025-1168 (1984).
- 31) Geus, A. J. and Cohen, W.: A Rule-Based System for Optimizing Combinational Logic, IEEE DESIGN & TEST, pp.22-32 (1985).
- 32) Darringer, J. A., Joyner, Jr. W. H., Berman, C.L. and Trevillyan, L.: Logic Synthesis Through Local Transformation, IBM J. Res. Develop., Vol. 25, No. 4, pp. 272-280 (1981).
- 33) 高木 茂: 論理装置のデータパス自動合成の手法, 情報処理学会論文誌, Vol. 26, No. 4, pp.715-725 (1985).
- 34) 高木 茂: データパスの検証・補正への一アプローチ, 情報処理学会論文誌, Vol. 26, No. 1, pp.9-18 (1984).
- 35) 高木 茂: テンプレートをを用いた ALU 回路の自動合成, 電子通信学会論文誌, Vol. J68-D No. 7, pp.1369-1375 (1983).
- 36) Takagi, S.: Rule Based Synthesis, Verification and Compensation of Data Paths, Proc. ICCD '84, pp.133-138 (1984).
- 37) Kowalski, T. J. and Thomas, D. E.: The VLSI Design Automation Assistant: Prototype System, Proc., 20th DA Conf., pp. 479-483 (1983).
- 38) Okino, N., Kakazu, Y. and Kubo, H.: TIPS-Technical Information Processing System for CAD/CAM in Kitakawa, T. (ed.), JARECT Vol. 7, Computer Science & Technologies, pp. 204-224, OHMSHA, LTD. and North-Holland Pub. Co., (1983).
- 39) 長澤 勲, 古川由美子: 拘束条件リダクション法を用いた機械設計計算支援システム, 情報処理学会論文誌, Vol. 27, No. 1, pp.112-120 (1986).
- 40) 長澤 勲: 設計支援システムのための知識表現

- と推論機構に関する研究, 九州大学工学研究科学学位論文 (1986).
- 41) 岸 義樹: 知識依存型概念設計システムの研究—機械構造の知識表現—, 精密工学会誌, 第51巻, 11号, pp. 83-89 (1985).
 - 42) 岸 義樹: 知識依存型概念設計システムの研究—数理構造の知識表現と操作—, 精密工学会誌, 第52巻, 8号, pp. 130-137 (1986).
 - 43) 加藤正男 (編): 小特集: VLSI の CAD, 情報処理, Vol. 22, No. 8, pp. 731-796 (Aug. 1981).
 - 44) 松下武史 (編): 特集: 設計・生産の自動化, 情報処理, Vol. 24, No. 1, pp. 2-87 (1983).
 - 45) 山崎利治 (編): 共通問題によるプログラム設計技法, 情報処理, Vol. 25, No. 9, pp. 934-962 (1984).
 - 46) 山崎利治 (編): 共通問題によるプログラム設計技法, 情報処理, Vol. 25, No. 11, pp. 1219-1268 (1984).
 - 47) Stefik, M.: Planning with Constraint (MOL-GEN: Part1) Artif. Intell., Vol. 16, pp. 11-139 (1981).
 - 48) Markus, A., Markus, Z., Farkas, J. and Filemon, J.: Fixture Design Using Prolog: An Expert System, Robotics & Computer-Integrated Manufacturing, Vol. 1, No. 2, pp. 167-172 (1984).
 - 49) McDermott, D.: Circuit Design as Problem Solving, in Latombe, I. C. (ed.), Artificial Intelligence and Pattern Recognition, in Computer Aided Design, pp. 227-245, North-Holland Amsterdam (1978).
 - 50) Gero, J. S. and Coyne, R. D.: Knowledge-Based Planning as A Design Paradigm, IFIP, WG 5.2, Working Conf., Preprints, pp. 261-295 (1985).
 - 51) McDermott, J.: A Rule-Based Configurer of Computer Systems, Artif. Intell., Vol. 19, No. 1, pp. 39-88 (1982).
 - 52) Steinberg, L. I. and Mitchell, T. M.: A Knowledge Based Approach to VLSI CAD: The Redesign System, 21th Design Automation Conf., IEEE, pp. 412-418 (1984).
 - 53) 小林重信: プロダクションシステム, 情報処理, Vol. 26, No. 12, pp. 1487-1496 (1985).
 - 54) Borning, A.: THINGLAB-An Object-Oriented System for Building Simulation Using Constraints, Proc. 5th IJCAI, pp. 497-498 (1977).
 - 55) Sussman, G. J. and Steele, Jr. G. L.: CONSTRAINTS-A Language for Expressing Almost-Hierarchical Descriptions, Artif. Intell., Vol. 14, pp. 1-38 (1980).
 - 56) Bobrow, D. G. and Stefik, M.: The LOOPS Manual, Preliminary Version, Xerox Corp. (1983).
 - 57) Ogawa, Y., Shima, K., Sugawara, T. and Takagi, S.: KRINE, An Approach to Integration of Frame, Prolog and Graphics, Proc. FGCS 84, pp. 643-661 (1984).
 - 58) 渡辺正信, 岩本雅彦, 山之内徹, 出口幸子, 松田裕幸: CLにおけるルール指向プログラミング, 情報処理学会知識工学と人工知能研究会報告 46-3, pp. 17-24 (1986).
 - 59) Barrou, H. G.: Verify: A Program for Proving Correctness of Digital Hardware Designs, Artif. Intell., Vol. 24, pp. 437-491 (1984).
 - 60) Genesereth, M. R.: The Use of Descriptions in Automated Diagnosis, Artif. Intell., Vol. 24, pp. 411-436 (1984).
 - 61) Davis, R.: Diagnostic Reasoning Based on Structure and Behavior, Artif. Intell., Vol. 24, pp. 347-410 (1984).
 - 62) 松家英雄: CAD/CAMにおけるデータベースについて, 情報処理, Vol. 23, No. 10, pp. 1000-1007 (1982).
 - 63) (社)日本電子工業振興協会: 論理設計 CADに関する調査, 61-C-528 (1986).
 - 64) (財)機械システム振興協会: 計算機援助による総合的製品開発システムに関する調査研究報告書, システム技術開発調査研究, 60-R-9 (1986). (昭和61年10月1日受付)