機械系 CAD 支援のための一つの知識ベースシステム*

車 周 憲^{*1}, 横 山 正 明^{*2}

A Knowledge-Based System for Supporting Mechanical CAD

Joo-Heon CHA and Masaaki YOKOYAMA

We present a framework for building a knowledge-based system that can consistently and automatically utilize graphs, tables and formulae as a knowledge source in mechanical CAD. In this paper, we describe both design object and design process using constraint networks which consist of constraint assemblies, constraint objects and constraint modules. The design object is described by a hierarchical structure of the constraint networks, and the design process is represented by constraint propagation between them. We introduce the concept of functional constraint in the constraint network and show that the database search for design variables can be carried out automatically. The usefulness of this system is illustrated by the application to the design of a gear system.

Key Words: Knowledge-Based System, Expert System, Intelligent CAD, Constraint Satisfaction, Constraint Propagation, Constraint Network, Problem Solver, Artificial Intelligence

1. 緒 言

人間による「設計」には大別して「創造的な設計」と「定式化された設計」の二つのカテゴリが考えられる。全く自由な発想で新しい人工物を創作しようとする場合は前者であり、定まった設計公式に従って例えば歯車設計を行おうとする場合は後者である。

本報告は、これら二つのカテゴリのうち後者の定式化された設計を支援するシステムを作るべく従来から新たに開発・報告してきた三つの方式、すなわち

- ① 設計のための知識源としての設計資料グラフおよびテーブルの持つ情報の記憶方式⁽¹⁾
- ② 記憶された情報の自動探索とその活用方式⁽²⁾
- ③ 制約ネットワークによる設計対象の記述と制約充足方式^{(3) (4)}

を統合して、定式化された機械系 CAD を支援する一つのシステムの構築を試みるものである。

近年、種々の分野においてエキスパートシステムと

呼称される知識ベースシステムの開発が盛んに行われている。機械設計の分野においても人工知能関連の技法や知識処理関連の技法を導入した機械設計の知的支援システムの構築が試みられているが^{(5)~(12)}、知識表現のために手続きの表現を加えなければならなかったり、宣言的な記述ができなかったり、大規模な対象の記述が困難であったり、あるいは本報告のような設計資料グラフおよびテーブルの情報が活用できなかったりするなど、まだ十分とは言えない。

定式化された機械設計を支援するシステムに関するこのような現状に鑑み、本報告は先人たちが蓄積してきた設計資料グラフおよびテーブルの持つ情報を有効に活用すべく、前述の三つの方式を組み込み、かつ今回新たに制約ネットワークにオブジェクト指向の概念と関数形の制約を導入して、一つの機械系 CAD 支援システムの構築を試みる。設計資料あるいは設計公式などは重要な設計知識であると考えて、このような設計知識に基づいた (knowledge-based) システムを本報告においては知識ベースシステムと呼称している。

なお、設計のための知識源として現在丸軸状部品・板状部品の応力集中係数ならびに歯車の歯形係数しか用意されていないので、本報告において取扱う問題はそのような機械部品の強度設計問題に限定されている

* 原稿受付 平成5年11月15日。

^{*1} 正員, 東京工業大学大学院 (〒227 横浜市緑区長津田町4259)。

^{*2} 正員, 東京工業大学。

図1 知識ベースシステムの概略枠組

が、ここに報告する方式は他の多くの設計問題にも拡大・適用可能であると考えている。

2. 知識ベースシステムの概要

本報告において構築しようとする機械系CAD支援のための一つの知識ベースシステムの概略枠組を図1に示す。図から明らかなように、このシステムはつぎの六つの機構から構成される。

- ①知識ベース (knowledge Base)
- ②知識獲得機構 (Knowledge Acquisition System)
- ③問題解決機構 (Problem Solver)
- ④ソリッドモデラ (Solid Modeler)
- ⑤形状認識機構 (Shape Recognition System)
- ⑥ユーザインタフェース (User Interface)

知識ベース①には機械工学便覧などに掲載されている設計資料グラフおよびテーブルなどの公的知識が格納されており、設計対象に関する物理量、荷重条件、境界条件、形状情報などと共に階層構造によって格納されている⁽²⁾。知識獲得機構②は設計資料グラフあるいはテーブルデータを有理Bezier曲線で統一的に記述し、データを圧縮して記憶し⁽¹⁾、あるいは設計知識を増やすなど知識ベースの増・改築や保守を行う。問題解決機構③は設計対象を設計変数(属性)および設計公式(制約)から構成される制約ネットワークで記述し、知識ベースに格納されている設計知識を活用してすべての制約を満足するように制約伝播により全属性値を決定する⁽³⁾。ソリッドモデラ④は設計対象の図面データを読み取り、B-rep表現のソリッドモデルを生成する。形状認識機構⑤は④のソリッドモデルから生成された幾何学的情報を面関係グラフに変換し、特徴パターンとこの面関係グラフのパターンマッチン

図2 設計対象記述の階層構造

グによって設計対象の形状認識を行う⁽²⁾。ユーザはユーザインタフェース⑥を通してシステムと対話し、あるいはシステムを制御することができる。例えば、設計仕様の入力、制約ネットワークにおける制約充足結果の表示あるいは矛盾の表示などのようなウィンドウ操作を実行できる。

本報告においては以上説明したような六つの機構の統合化により、定式化された機械系CAD支援のための一つの知識ベースシステムの構築を試みる。

3. 制約ネットワークによる設計記述

3.1 制約ネットワークによる設計対象の記述 本報告では設計対象(機械および機械部品)を制約ネットワークによって記述する。図2は本システムにおける設計対象記述の階層構造を示したものである。この図において、一番低い層に制約(Constraint)と属性(Attribute)がある。制約は設計公式および関数形の制約から構成され、属性は入力仕様定数および設計変数から構成される。この二つの要素が制約ネットワークにおける基本要素であり、これらの基本要素の相互結合関係によって部分過程の制約ネットワークである制約モジュール(Constraint Module)が構成される。

この制約モジュールの上層に位置する制約オブジェクト(Constraint Object)は制約モジュールの有機的集合によって構成される。この制約オブジェクトは機械部品に対応する。現実の機械部品の設計における制約式の数は多く、一つの制約ネットワークで記述するとあまりにも複雑になる。このような複雑な制約ネットワークの場合属性値の決定に際して見通しは悪く、すべての制約を満足するように全属性値を決定するのは容易ではない。そこで、本報告では設計の全過程を

図3 制約ネットワークによる一段変速歯車系の設計概念図

いくつかの小さな部分過程に分割し、設計対象を記述する制約ネットワークをいくつかの制約モジュールの集合として記述して取扱う。例えば、変速歯車の設計の場合は幾何形状モジュール、曲げ強さモジュール、歯面強さモジュールの3つの制約モジュールに分割して考える。この制約モジュール間における制約充足方式については別報⁽³⁾を参照されたい。

さらに上層に位置するものとして、制約オブジェクトの有機的な集合によって構成される制約アセンブリ (Constraint Assembly) を考える。ここで、制約アセンブリは組立てられた機械構造に対応する。例えば、一段変速歯車系は三つの機械部品、すなわち一つの歯車対と2本の軸によって表現する。一段変速歯車系を一般的なオブジェクト指向によって表現すると、2個の歯車と2本の軸のように四つの要素により表現される。しかし、本報告においては設計対象をオブジェクト指向によって表現するのではなく、制約ネットワークによって記述するので、2個の歯車を一つのオブジェクトとしての歯車対として取扱う。

3.2 制約ネットワークによる設計過程の記述 前述のように、本報告では設計対象を制約ネットワークの有機的な集合によって記述する。しかし、設計では設計対象とその設計過程を分離して考えることはできないので、ここでは設計過程をその制約ネットワーク間の制約伝播、すなわち制約モジュール間の制約伝播および制約オブジェクト間の制約伝播の方向付けによって記述し、一貫性のある設計過程の記述を図る。

例えば、機械系の定式化された設計過程においては、

ある部分を設計した後、つぎの部分の設計に進み、前の設計との整合性をチェックし、満足すればさらにつぎの部分の設計に進むというように、その設計過程を複数の小さな部分過程に分割して考えた方が理解しやすい。また、このような分割は制約ネットワークによって表現された設計対象に対してその制約充足方式を考えやすいので、別報⁽³⁾においてもこの方式を採用している。

例として、図3に制約ネットワーク記述による一段変速歯車系の設計概念図を示す。図中の2種類の矢印は共通属性の伝播の方向を示す。それらは同時に、設計過程の流れの方向も示す。

最初は、歯車対の設計過程である。歯車設計の場合、まず幾何学的形状を決定してから、大・小歯車において歯元曲げ応力の計算を行う。この歯元曲げ応力が許容曲げ応力より小さくなるように幾何学的形状を決定する。つぎに歯面接触応力の計算を行い、最大ヘルツ接触応力が許容ヘルツ応力より小さいか否かをチェックし、満足できれば歯車設計は一応終了する。満足できなければ幾何学的形状や曲げ強さにおける属性値を修正し、歯面強さを満足するように設計変更を行う。

この設計過程を制約ネットワークで表すと図3の左側にあるような歯車対の制約オブジェクトになる。図に示されるように、歯車対の制約オブジェクトは幾何学的形状、曲げ強さ、歯面強さの三つの制約モジュールによって構成される。

つぎは、動力伝達軸の設計過程である。軸設計の場合は図3の右側のようにそれぞれの軸に対して二つの

制約モジュールを考える。すなわち、1本の軸の強度設計において、ねじりトルクと曲げモーメントを同時に受ける部分と曲げモーメントのみを受ける部分に分けて設計を行い、軸の無駄な肉付きをさける。この軸の設計過程も歯車対の場合と同様に制約モジュール間のネットワークで表現する。

図3における2種類の矢印は共通属性の伝播方向を示し、それらは同時に設計過程の流れの方向を示すものであると前述した。この図3において、矢印「 $\Rightarrow$ 」は歯車対から2本の軸の方向にのみ向いている。これは歯車対の設計を先ず行ってから、つぎに2本の軸の設計に移行することを意味する。このように、共通属性の伝播方向を示す矢印により設計過程の流れの方向を示すことができる。本報告の場合、先に設計したい機械部品の制約オブジェクトを先に入力する方式を採用している。

以上のように、本報告では設計の全過程を複数の小さな部分過程に分割し、その部分過程を制約ネットワークで記述し、その制約ネットワークの有機的な集合によって設計対象および設計過程を統一的に記述することができる。

3.3 関数形の制約記述 前述のように、制約は設計公式と関数形の制約によって構成される。関数形の制約はグラフあるいはテーブルから作成されたデータベースから制約充足に必要なデータを自動的に探索・獲得するために導入したものである。

また、設計公式には等式や不等式があるが、本報告では制約ネットワークにおける制約ノードとしては二つの制約、すなわち等式および関数形の制約のみを取扱うことにする。

通常の機械設計において、不等式の制約は制約ネッ

図4 関数形の制約を含んだ制約ネットワークの例

トワークにおけるすべての制約充足を行った後、求められた属性値がその不等式を満足するか否かの評価のみに使われるので、本報告では不等式の制約は使わず、その代わりにユーザがシステムと対話しながらその評価を行う方式を採用する。

関数形の制約は基本的に数学における関数と同様であり、引数の値を与えればその値を求めることができる。図4に関数形の制約 $\alpha_s = F(d_s, D_s, r_s)$ の例を示す。この図はねじりトルクと曲げモーメントを同時に受ける動力伝達軸の制約モジュールを示す。属性ノードの軸径 d_s を求めるためには、応力集中係数に関するデータベースから応力集中係数 α_s の値を獲得しなければならないが、本システムの場合には形状認識機構を用い、軸の形状を自動認識することにより、形状属性である引数 d_s, D_s, r_s が自動的に求められる。

以上のように、本報告のシステムにおいては、制約ネットワークに関数形の制約を導入することにより、データベースから必要な設計資料データを自動的に探索・獲得することができる。

4. 知識ベースシステムの実行例

4.1 システムとその実行法 本システムは、HP Apollo 9000/425t 上でC++言語およびX Windowシステムによって作成された。

図5に本システムにおいて重要な問題解決機構におけるクラスの階層構造を示す。クラス「制約アセンブリ」から生成されるインスタンスは組立てられた機械構造を表し、それを構成する機械部品のインスタンスはクラス「制約オブジェクト」によって生成される。また、制約オブジェクトはクラス「制約モジュール」から生成されるインスタンスによって構成され、それらの有機的な関係によって設計対象における設計過程が表現される。

図5 問題解決機構におけるクラスの階層構造

さらに、前述のようにクラス「制約」は二つのクラス、すなわち「等式」と「関数形」により構成される。クラス「等式」では、設計式を二分木で表現し、変数消去などの数式処理を行う⁽³⁾。また、クラス「関数形」は設計に必要な知識ベースへのポインタを持っており、設計資料データの自動探索を行う。

図示したように、クラス「**ウィンドウ」はインスタンス(制約アセンブリ、制約オブジェクト、制約モジュール、属性など)が制約ネットワークに生成されるたびに、そのインスタンスに対応したウィンドウを階層構造に生成し、画面に出力させる。

本システムの実行法であるが、ユーザは各制約モジュールごとに制約式、すなわち設計式を入力するだけでよく、制約ネットワークは自動的に構成される。この際、先に設計したい機械部品の制約オブジェクトをその名前と共に先に入力し、各制約オブジェクトにおいては先に制約充足したい制約モジュールをその名前と共に先に入力する。

図6に歯車対について制約式の入力法の例を示す。図において「OBJECT」、「MODULE」、「CONSTRAINT」はそれぞれ制約オブジェクト、制約モジュール、制約式を表すキーワードである。図6から制約オブジェクトである歯車対は三つの制約モジュール、すなわち幾何形状モジュール、曲げ強さモジュール、歯面強さモジュールから構成されていることが分かる。また、各制約モジュールは数個の制約式によって構成されている。制約式は等式制約と関数形の制約によって別々に記述される。例えば、式 $d2=z2[i]*m[d1.dat]$ のように記述される制約は等式制約を表し、式 $?Y1\{d3.dat\}=z1[i]$ のように記述される制約は関数形の制約を表す。ここで、 $d2$ のように $[\]$ が付いてない設計変数は連続値属性を表す。また、 $z2[i]$ のように $[\]$ の中に i が入っている設計変数は整数値属性を表し、歯車モジュールの $m[d1.dat]$ のようにデータファイル名が入っている設計変数は離散値属性を表す。

このような制約式が入力されると、本システムは直ちにその制約式から制約インスタンスおよび属性インスタンスを生成させ、制約モジュールごとに制約ネットワークを自動的に構成する。

4.2 一段変速歯車設計における実行例 一段変速歯車系の設計を例に取って、本システムの実行例を説明する。

図3に示したように、一段変速歯車系は三つの機械部品(制約オブジェクト)、すなわち1個の歯車対と2本の軸によって構成される。さらに、歯車対は三つの制約モジュールで、各軸はそれぞれ二つの制約モジ

```

OBJECT
GearPair
MODULE
Geometric
CONSTRAINT
a=(d1+d2)/2
d2=z2[i]*m[d1.dat]
u=z2[i]/z1[i]
d1=z1[i]*m[d1.dat]
MODULE
Strength
CONSTRAINT
Sig1=F1*Y1/b/m[d1.dat]
Sig2=F2*Y2/b/m[d1.dat]
F1=19100000*P/d1/n1
F2=19100000*P/d2/n2
u=n1/n2
?Y1{d3.dat}=z1[i]
?Y2{d3.dat}=z2[i]
MODULE
Surface
CONSTRAINT
K=F1/b/d1*(1+1/u)
Sgh=SQRT(K)*Zh*Zm

```

図6 歯車対における制約式の記述例

ュールで構成される。

歯車対に対する設計仕様として、伝達動力 $P=21\text{ kW}$ 、駆動歯車の回転数 $n1=300\text{ rpm}$ 、中心距離 $a=150\text{ mm}$ 、変速比 $u=3$ 、圧力角 $\alpha=20^\circ$ 、モジュール $m=5\text{ mm}$ 、大小歯車材の許容曲げ応力 $\sigma_s=200\text{ MPa}$ 、許容ヘルツ応力 $\sigma_h=800\text{ MPa}$ 、領域係数 $Z_h=2.5$ 、材料定数係数 $Z_m=200$ を与える。転位のない歯車対とする。また、軸の設計仕様として長さ $L=300\text{ mm}$ 、軸材料の許容ねじり応力 $\tau_s=100\text{ MPa}$ を与える。

以上のような設計仕様をシステムに入力し、一段変速歯車系の設計を行う。図3に示したように、歯車対において先に幾何制約モジュールの制約充足を行い、未知属性のピッチ円直径 $d1$ 、 $d2$ および歯数 $z1$ 、 $z2$ の値を求め、その結果を他の二つ制約モジュールに伝播させる。このとき、曲げ強さ制約モジュールには $d1$ 、 $d2$ 、 $z1$ 、 $z2$ 、 m 、 u の六つの属性値が、歯面強さモジュールには $d1$ 、 u の二つの属性値がそれぞれ伝播される。

つぎに、曲げ強さ制約モジュールにおける制約充足を行い、すべての未知属性の値を求める。ここで、大・小歯車の歯形係数 $Y1$ と $Y2$ は歯数 ($z1$ と $z2$) と転位係数による関数形の制約として取扱われるので、その属性値はデータベースから自動的に探索・獲得される。

さらに、歯面強さ制約モジュールへの制約伝播を行う。このとき、幾何制約モジュールには伝播されず、歯面強さ制約モジュールにのみ共通属性である $d1$ 、 u 、歯幅 b の値が伝播される。この歯面強さ制約モジュールにおいて制約充足を行い、未知属性の値を求める。図7にその結果を示す。この図において最大ヘルツ接

書かれている属性は関数形の制約の引数を表す。

つぎに、この結果を曲げモーメントのみを受ける部分の制約モジュールに伝播させ、そこで制約充足を行う。図10にその結果を示す。ここにおいても関数形の制約によるデータベースの自動探索が行われている。

図9と図10から、ねじりトルクと曲げモーメントを同時に受ける部分の軸径 d_{s1} は29.04mm であり、曲げモーメントのみを受ける部分の軸径 d_{s2} の28.09mm に比べて大きいことが分かる。最後に歯車の穴径 D_{s1} と D_{s2} の値を安全側に一致させ、軸の設計を終了する。

以上の過程から分かるように、ユーザは本知識ベースシステムと対話しながら、設計過程に従って制約モジュールごとに制約充足させ、制約ネットワークの中の全ての制約を満足するように設計対象の全属性値を決定する。

5. 結 言

設計資料グラフおよびテーブルなどの設計知識源が従来の機械系CADシステムにおいて十分に活用されていないことに鑑み、本報告においてはこれらの設計知識を有効かつ自動的に活用すべくこれまで開発してきた諸技法を統合して、定式化された機械系CAD支援のための一つの知識ベースシステムを構築した。

この構築に際して、制約ネットワークにオブジェクト指向の概念を導入し、それらの制約ネットワークの有機的な集合によって設計対象および設計過程を統一的に記述する方式を提案した。また、制約ネットワークにおける制約充足および制約ネットワーク間の制約

伝播によって定式化された設計が行えることを示した。

さらに、制約ネットワークに関数形の制約を導入することにより、設計知識のデータベースから必要な設計データを自動的に探索・獲得できることを示した。

最後に、構築した知識ベースシステムを一段変速歯車系の設計に適用し、本システムの有用性を確認した。

文 献

- (1)横山・車, 日立地方講演会論文集, (1990), 88.
- (2)車・横山, 機論, 58-556, C(1992), 3767.
- (3)車・ほか2名, 機論, 59-568, C(1993), 3889.
- (4)Cha, J-H., Yokoyama, M. and Okabe, I., Proceedings of ICED 93, (1993), 1335.
- (5)Dym, C.L., Application of Knowledge-Based Systems to Eng. Analysis and Design, (1985), ASME.
- (6)Gero, J., Knowledge Engineering in Computer-Aided Design, (1985), North Holland.
- (7)長澤・古川, 情報処理学会論文誌, 27-1(1986), 112.
- (8)赤木・藤田, 機論, 54-500, C(1988), 1017.
- (9)今井, 設計シンポジウム講演論文集, (1992), 1.
- (10)岸, 精密工学会誌, 58-6, (1992), 59.
- (11)Serrano, D. and Gossard, D.C., in Tong, C., and Sriram, D. (ed.), Artificial Intelligence in Eng. Design, Vol.1, Academic Press, (1992), 71.
- (12)Young, R.E., Greef, A. and O'Grady, P., in Gero, J. S. (ed.), Artificial Intelligence in Design '92, (1992), 79.