

Title	ネットワークモデルによる設計過程の支援 : エキスパートCADシステムにおける設計処理の機能
Author(s)	赤木, 新介; 藤田, 喜久雄
Citation	日本機械学会論文集 C編. 54(505) P.2300-P.2306
Issue Date	1988-09
Text Version	publisher
URL	<a href="http://hdl.handle.net/11094/3221">http://hdl.handle.net/11094/3221</a>
DOI	
rights	
Note	

*Osaka University Knowledge Archive : OUKA*

<https://ir.library.osaka-u.ac.jp/>

Osaka University

ネットワークモデルによる設計過程の支援\*  
— エキスパートCADシステムにおける設計処理の機能 —

赤木新介\*1,

藤田喜久雄\*2

A Network Model Supporting the Process of Basic Design  
— New Approach to design modelling for Expert CAD System —

Shinsuke AKAGI and Kikuo FUJITA

A network model is developed to support the process of the basic design of engineering systems. The model is built according to the expert CAD system which has been developed and introduced in the previous report by the authors. In the model, design parameters and their relations are represented as the nodes and arrows in a network. The model satisfies handle design parameters flexibly in a computer system using object-oriented knowledge representation : 1) interactive control of design process by the designer, 2) flexible model building and easy modification, 3) assistance in determination of design parameters by the quantitative information on relationships among them.

The validity and effectiveness of the model is ascertained by applying it to the expert CAD system for the basic design of a ship. It is also applicable to expert CAD systems for various mechanical systems.

Key Words : Design Engineering, Expert System, Network Model, CAD

## 1. 緒言

コンピュータによる設計支援は、数値計算や図形処理などの数値処理を中心として行われてきたが、数式処理やエキスパートシステムに代表されるような記号処理に基づく設計支援が試みられつつある<sup>(1)</sup>。著者らは、エキスパートシステムによる設計支援の試みとして、機能設計を支援する目的から、汎用設計知識処理システム<sup>(2)</sup>を開発した。設計は、設計条件を満足し十分な機能を有する対象物の像を作り出す過程であり、なんらかの満足化や最適化を含む<sup>(3)</sup>。単純な問題であれば設計解は条件をもとに直ちに得られるが、多くの問題(特に評価項目が多く構造が良くない問題の場合)は、設計者の柔軟な判断による試行錯誤により設計解を求める必要がある。上記のシステムは、設計モデルの表現にオブジェクト指向の概念<sup>(4)</sup>を導入して、知識表現のモジュール性を向上させるとともに、

設計パラメータ間の依存関係を自動的に処理し、さらに、設計解の評価診断をルール型の知識表現により行うことによって、設計における試行錯誤過程を柔軟に支援することをねらいとしたものである。このシステムでは、設計モデルの修正は、評価診断結果を参考にしたユーザの判断に任されており、設計者の経験に依存していた。

本研究では、設計パラメータ間の依存関係に対応して、設計モデルをネットワークとして扱う。これに基づいて、モデルの生成過程に対して、設計パラメータ間のネットワークを表示し、また、数式微分を用いた感度解析により適切な設計変数の値を提示し、それによる設計モデルの変化を予測することにより、より有効に設計過程を支援することを試みる。このようなネットワークモデルの特徴は、設計変数間のつながりが陽に把握でき、ビジュアルに表示できる点にあり、設計者はこれによって、効率よく設計モデルを生成することが可能となる。事例として、既報<sup>(5)</sup>でも取り上げた船舶の基本設計におけるモデルの生成過程に適用し、本手法の有効性を検証する。

\* 昭和63年3月19日 関西支部第63期定時総会講演会において講演, 原稿受付 昭和62年6月1日。

\*1 正員, 大阪大学工学部 (〒565 吹田市山田丘2-1)。

\*2 学生員, 大阪大学大学院。

## 2. 設計プロセスの理解と設計モデルの表現

本章では、準備として、設計プロセスを理解するとともに、設計モデルの表現方法<sup>(2)</sup>について示す。

2.1 設計プロセス 設計のプロセスは、「a) 扱うべき設計モデルの作成」・「b) モデルによるシミュレーション」・「c) シミュレーション結果に対する評価」・「d) 評価結果に基づくモデルの修正」に大別できる<sup>(6)</sup>。d)は、さらに「e) 修正案の生成」と「f) 修正の実行」に分れる。設計エキスパートシステムでは、これらの過程をいかに有効に支援するかが問題となる。しかし、設計過程では、ある設計項目の値を決定するために付随する他の項目の値を用いるという関係が再帰的に連鎖しているために、設計モデルを適切に操作することは経験者にとっても困難な問題である。

2.2 設計モデルと設計処理 このような設計プロセスに対して、オブジェクト指向の概念<sup>(4)</sup>を導入し、上記の依存関係を自動的に処理することにより支援を行う<sup>(2)</sup>。つまり、設計知識を個々のパラメータにまで細分化し、それぞれに関する知識をモジュール化して、オブジェクト(インスタンス)として表現し、設計モデルをその集合として取り扱う(図1)。設計処理は、処理を起動するメッセージを、このオブジェクトに送信することにより行い、メソッドの処理機能により、そのオブジェクトから付随する他のパラメータのオブジェクトに対して、同様のメッセージが再帰的に送信されることにより、設計項目間の関係が処理される。図2はこの処理の具体例であり、ある設計パラメータ(インスタンスA)の決定に関するものである。この処理は、「インスタンスA」に対して、決定処理を起動するメッセージ①を送信することにより行われる。メソッドの内容は、各インスタンスに共通する概

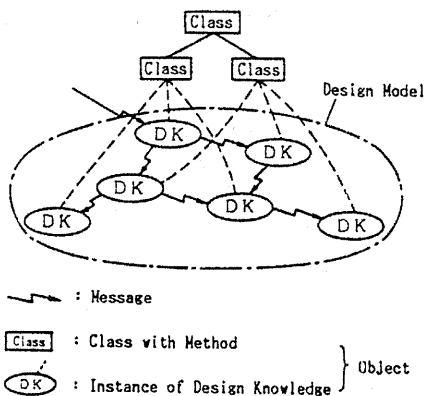


図1 設計知識のオブジェクトモデル

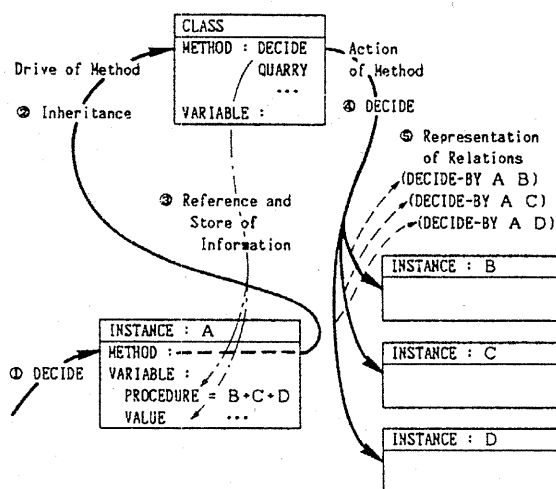


図2 オブジェクト間の継承と設計処理の連鎖

念を定めた「クラス」のなかに記述されており、これは「インスタンスA」に継承②されている。メソッドが実行されると、必要な情報③が「インスタンスA」の内容から参照され、さらに、Aを決定するためにその値が必要となる項目(「インスタンスB」、「C」、「D」)に対して、同様のメソッドを起動するメッセージ④が送信される。このとき、自動的にパラメータ間の依存関係AとB、AとC、AとDの関係が述語として記録⑤される。この述語は、非構造的な知識を表現するための知識表現の方法であり(いわゆる述語論理の意味ではない)、二つのインスタンスの間の依存関係を述語として表現することにより、双方向からこの記述に対してアクセスすることができるようになる。これに対して、パラメータ間の依存関係により、あるパラメータを修正(一次的修正)した時に、その影響が及ぶ他のパラメータも修正(二次的修正)する必要がある場合の操作は、⑤の記録をたどりながら、旧データを消去しつつ行われる。なお、具体的なインスタンスには、各パラメータの設計手続きが、インスタンス変数であるPROCEDUREに記述されており、設計の過程で定められるパラメータの値なども保持される。

以上のオブジェクト処理機能に加えて、ルール型知識表現により、シミュレーション結果の評価診断を支援する<sup>(2)</sup>。

これらの二つの機能により、前述の設計プロセスのかなりの部分が支援できるが、「e) 修正案の生成」は、評価結果を参考にしたユーザの判断に任される。しかし、効率的な設計を行うためには、有効な修正案を生成することが重要である。本研究では、以下のように、このようにオブジェクト指向に基づいて処理さ

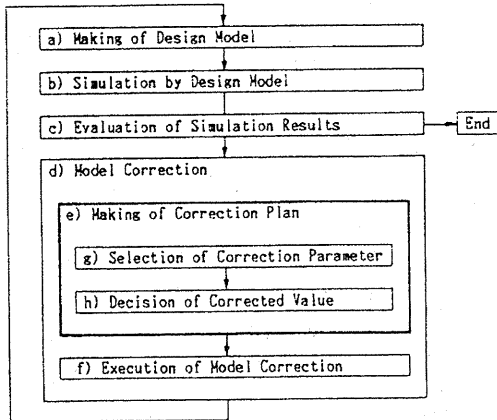


図3 設計プロセス

れる依存関係を、ネットワークモデルにより表現することにより、設計モデルの生成に対して指針を与えることを試みる。

### 3. ネットワークモデルによる機能

#### 3.1 オブジェクト指向とネットワークモデル

設計プロセスにおける「e) モデルの修正案の生成」の過程は、「g) 修正すべきパラメータの選定」と「h) そのパラメータの新しい値の設定」の二つに大別できる(図3)。この二つの過程に対して、図2のようにして、オブジェクト指向に基づいた処理により、自動的に記録される設計パラメータ間の依存関係を、改めてネットワークモデルとしてとらえることにより支援を行う。すなわち、前者を、様々な設計パラメータ間の依存関係を明示的に表現したネットワークを表示することにより、後者を、ネットワークの節点に対応する設計パラメータのインスタンスに記述された設計手続きに対して数式微分を行い、望ましい設計モデルを感度解析的に表示して、指定したパラメータをどのように設定すべきかを予測することにより支援する。

このようなネットワークモデルによる支援の利点は、緒言でもふれたように、設計パラメータ間のつながりや量的な依存関係を明示的に表現できる点にある。一般に設計者は、設計プロセスの中で、ある設計パラメータの値を変化させたとき、それが評価項目や中間的な設計項目の値にどのように影響するかを知ることが求められるものである。たとえば、それが最適設計などのシミュレーション過程であっても、その過程がブラックボックス的に処理され、結果だけが表示されるのでは、設計者は満足しない。本ネットワークモデルは、このような点の改善もねらいとしている。

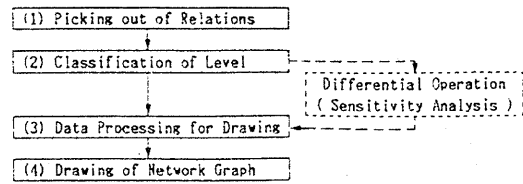


図4 設計パラメータのネットワーク表示のアルゴリズム

3.2 設計パラメータ間の依存関係：ネットワークの表示 設計パラメータ間の依存関係は、設計の過程で図2に示したようにして記録されるが、この関係は、各設計パラメータに対応する節点と、依存関係に対応して依存される方から依存する方への向きを持つアローとから構成される有向グラフとして理解できる。そこで、設計パラメータ間の依存関係に対応する有向グラフを、関係を表現した述語に対する記号処理により、要素(設計パラメータ)を階層化して、自動的にネットワークとして表示することにする。これにより、設計パラメータの選定に関して、見通しを与えることができる。

3.2.1 ネットワーク表示のアルゴリズム ネットワーク表示のおおまかなアルゴリズムを図4に示す。図中の各モジュールでは、それぞれ、以下の処理を行う。

(1) 依存関係の抽出：現在の設計モデルにおける依存関係の中から、注目したいあるパラメータに関連するパラメータ間の関係を抽出する。

(2) 依存関係のレベル分け：抽出された依存関係に含まれるパラメータを、その関係に従って階層に分ける。

(3) グラフ表示のための前処理：パラメータ間の依存関係やレベル分けのみでは、グラフ表示を行うことは困難であり、これらのデータを表示のために加工する。

(4) グラフ表示：加工されたデータをもとにグラフの表示を行う。

次項以降に、これらの処理の詳細を示す。

3.2.2 依存関係の抽出とレベル分け 前述のように、設計パラメータ間の依存関係は、

( DECIDE-BY パラメータA パラメータB )

という形式の述語として表現される<sup>(2)</sup>。なお、この例は「パラメータAの値がパラメータBの値に依存して決定された」ことを意味している。このような依存関係は設計の状況に応じて数多くの関係が存在するため、すべての依存関係に関するネットワークを表示す

ることは適切ではなく、注目したいパラメータを指定して、その起点となるパラメータが依存する(あるいは、パラメータに依存する)各パラメータ間のネットワークのみを抽出して、表示を行うことにする。この依存関係の抽出は、関係を記述した述語に対するパターンマッチングを行うことにより処理することができる。続いて行うネットワークのレベル分けも、抽出した関係に対して、「そのパラメータに依存して決定されたパラメータが存在するかどうか」をパターンマッチングにより判断する処理を繰り返すことにより、実行できる。

以上の処理の具体例を図5に示す。いま、仮に(a)のような依存関係があったとする。これには、(b)のネットワークが対応する。パラメータAが依存する関係に注目した場合に抽出される述語が(c)であり、(a)の中から、(DECIDE-BY B D)が除かれる。つまり、(b)の破線で囲まれた部分のネットワークに焦点が当てられる。続いて、レベル分けが行われ、(d)の述語が生成される。

3.2.3 グラフ表示 前項で示した処理に続いて、グラフ表示のためのデータの加工が行われる。グラフを表示するためには、各パラメータに対応する節点を画面上でどのように配置し、依存関係に対応するアローをどのように引くかを決定する必要がある。ここでは、あるレベルの節点は画面上で横一列に配置するものとし、また、離れたレベル間の関係は両節点の間の各レベルに中継点を設けて、それらの上を通過してアローを引くことにする。例えば、図5の(b)におけるAとDのような関係を図6のように置き換えることにより対処する。同図の \*P1 はアローを引くための中継点を表し、図5(b)中の黒丸は、このような点に対応している。その他、各レベルの節点を横一列でどのような順番で並べるかが決定される。実際の表示例は、4章で示す。

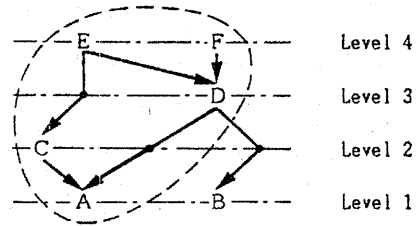
3.3 数式微分を用いた設計モデルの生成支援

設計過程において、設計結果の評価診断結果が表示され、さらに、そのような結果に関連している設計パラメータが、前節のネットワークで表示されると、修正すべきパラメータが明らかになる。次に、その新しい値をどのように設定するかを決定しなければならない。このとき、「そのようなパラメータの設定が他の項目にどのように影響するか」あるいは、「ある評価項目をある値にするためには、そのパラメータをどのような値に設定すれば良いか」がわかれば、設計モデルの生成を支援することができる。

あるパラメータの値の変更による他のパラメータの

- (DECIDE-BY A C)
- (DECIDE-BY A D)
- (DECIDE-BY B D)
- (DECIDE-BY C D)
- (DECIDE-BY C E)
- (DECIDE-BY D E)
- (DECIDE-BY D F)

(a) 依存関係を表現した述語



(b) (a)に対するネットワーク

- (DECIDE-BY A C)
- (DECIDE-BY A D)
- (DECIDE-BY C D)
- (DECIDE-BY C E)
- (DECIDE-BY D E)
- (DECIDE-BY D F)

(c) 抽出された述語

- (LEVEL A 1)
- (LEVEL C 2)
- (LEVEL D 3)
- (LEVEL E 4)
- (LEVEL F 4)

(d) レベル分けを表現した述語

図5 ネットワーク表示処理

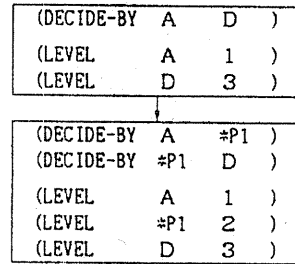


図6 ネットワーク表示のためのデータ加工

値の変化は、それぞれの設計手続きや現在の値に依存する。つまり、値を変更しようとするパラメータをx、ある評価項目をz、zの設計手続き中に含まれる他のパラメータをy<sub>1</sub>, y<sub>2</sub>, ..., y<sub>m</sub>としたとき、

$$\frac{dz}{dx} = \sum_{i=1}^n \frac{\partial z}{\partial y_i} \frac{dy_i}{dx} \dots\dots\dots(1)$$

の関係を用いると、 $x$ を $x_0$ から $x_n$ に変更したときの $z$ の変化量は、 $dz/dx(x_n-x_0)$ として、感度解析的に得られる。一方、 $z$ を $z_0$ からの $z_n$ とするような $x_n$ も、これにより、 $x_0+(z_n-z_0)/(dz/dx)$ により求めることができる。以上の処理の中で必要となる各パラメータの値や設計手続きは、2章で示したように、ネットワーク中の対応するオブジェクトのインスタンス変数として保持されており、自由に参照できる。また、式(1)中で必要となる各パラメータの偏微分値も、各パラメータの設計手続きを参照し、それに含まれる他のパラメータを抽出して、それぞれによる偏微分式を数式微分により求め、それを評価することにより得られる。なお、数式微分はパターンマッチングを用いたLISPプログラムにより行うことができる<sup>(7)</sup>。一方、このような処理の手順は、前節で示したレベル分けに従って定まる。以上のようにして得られる新しい設計モデルの予測値も、前節で示したネットワーク上で表示され、設計モデルの生成を支援することができる。

なお、以上の機能は、大型コンピュータACOS1000上のUTILISP言語により記述されており、ネットワークの表示はFORTRANプログラムを呼び出して行う。

#### 4. 船舶の基本設計過程における設計モデルの生成支援

本手法を、既報<sup>(5)</sup>でも取り上げた船舶の基本設計過程における設計モデルの生成過程に適用した事例を示す。

船舶の基本設計過程は、要求される船のサイズや速力を満たし、かつ安全性や経済性などの評価項目を満足化するように、船の要目(長さ、幅、深さなど)を定めるプロセスである<sup>(8)</sup>。この過程では、ある程度設計者の自由裁量を許しながら最適化を行うところに特徴があり、2・1節で示したような設計プロセスを繰り返す。前述のように既報<sup>(5)</sup>では、このような基本設計過程に対して、汎用システム<sup>(2)</sup>のオブジェクト処理機能により支援を行ったが、修正案の生成に関しては、設計結果の問題点を指摘するのみで、ユーザの判断に任されていた。ここでは、前章で示した手法により、この部分を支援する。

図7は、あるばら積み貨物船の基本設計において、基本性能の推定値をもとに主要寸法を決定している部分の実行例で、ある設計結果を評価診断し、それをもとに新しい設計モデルを生成して、再設計を行ったも

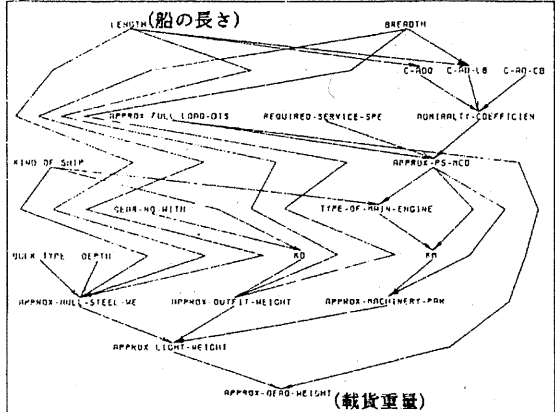
#### ①: 船の要目(初期値)

DESIGN RESULTS		STEP = 1	TIME = 1
KIND OF SHIP	.....	87LE-CA	1
REQUIRED-DEAD-WEIGHT	.....	34000.0	1
REQUIRED-SERVICE-SPEED	.....	14.8999	1
APPROX-FULL-LOAD-DISPLAC	.....	49720.4	1
LENGTH	.....	187.370	1
BREADTH	.....	27.1517	1
DEPTH	.....	15.2181	1
FULL-LOAD-DRAFT	.....	11.1092	1
APPROX-BLOCK-COEFFICIENT	.....	0.50765	1
APPROX-GM-ON-FULL-LOAD	.....	2.63130	1
APPROX-GM-ON-HULL-AST	.....	3.49710	1
TYPE-OF-MAIN-ENGINE	.....	1.01-SPE	1
APPROX-PS-MCO	.....	6550.50	1
APPROX-LIGHT-WEIGHT	.....	2257.73	1
APPROX-HULL-STEEL-WEIGHT	.....	7442.95	1
APPROX-OUTFIT-WEIGHT	.....	854.900	1
APPROX-MACHINERY-PART-WE	.....	600.284	1
APPROX-DEAD-WEIGHT	.....	37432.6	1

#### ②: ①に対する評価診断

APPLY DIAGNOSIS RULE  
DEAD-WEIGHT WILL BE TOO SMALL AS COMPARED WITH REQUIRED-DEAD-WEIGHT

#### ③: 載貨重量に関係する項目



#### ④: 修正すべき設計項目の指定

SELECT DESIGN VARIABLES TO BE CORRECTED  
>>> PLEASE INPUT VARIABLE NAME WANTED TO CORRECT ! ( E - RE-DESIGN )  
? LENGTH  
>>> PLEASE INPUT VARIABLE NAME WANTED TO CORRECT ! ( E - RE-DESIGN )  
? E  
>>> SELECT VARIABLE NAME DECIDE BY OTHER MAIN DIMENSION VARIABLE .  
>>> ( 1 - APPROX-FULL-LOAD-DISPLACEMENT , 2 - FULL-LOAD-DRAFT ,  
3 - APPROX-BLOCK-COEFFICIENT , 4 - BREADTH , )  
... ? 1

#### ⑤: 設計項目(船の長さ)の感度値

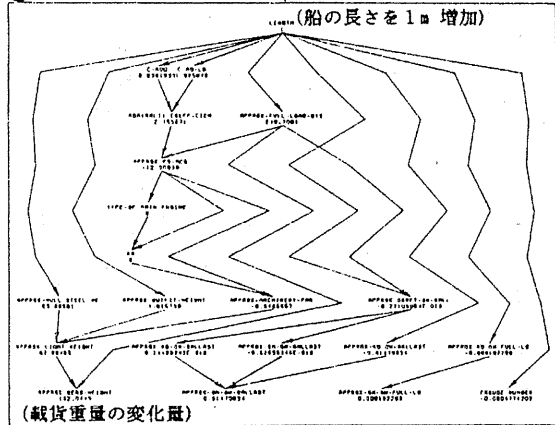


図7 実行例

のである。この基本設計の初期段階は、設計手続きが比較的簡単な算術式で記述されており、3・2節の手

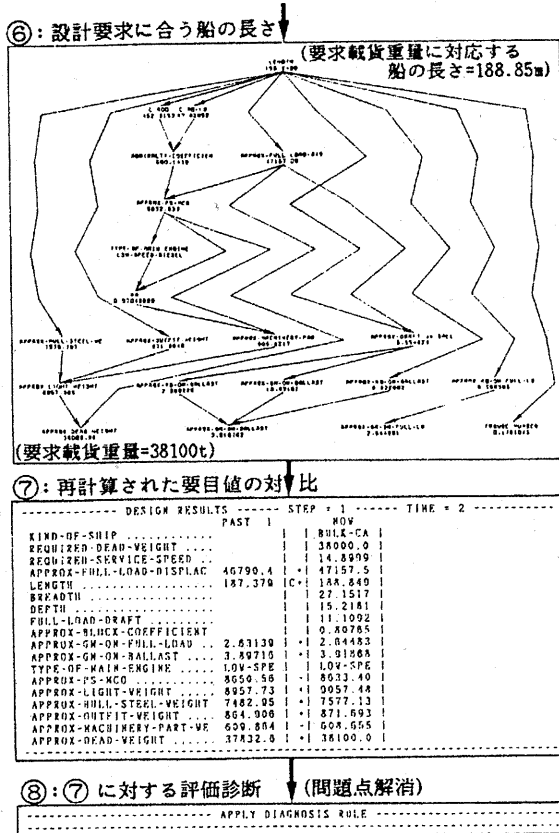


図7 実行例 (続き)

法が適用できる。以下、図中の番号に従って説明する。

- ①: 設計条件に対する船の要目(初期値)が表示される。設計条件は、載貨重量が38000t、航海速力が14.9knotであり、それに対して決定された主要寸法は、長さが187.38m、幅が27.15m、...であり、続いて各評価項目が表示されている。
- ②: ①の設計結果に対して評価診断を行うと、「載貨重量が不足する」ことが表示される。
- ③: ②に対して、ユーザが、載貨重量を修正するために、これに関係するパラメータ間のネットワークの表示を求め、そのネットワークが表示される。
- ④: ②の診断結果や③のネットワークの表示をもとに、修正すべきパラメータとして船の長さを指定する。また、それに付随する項目の中で満載排水量を修正することを指定する。
- ⑤: 長さとして依存して決まるパラメータの関係がネットワークとして表示され、同時に、長さを1m増加させた場合に対する他のパラメータ間の増減量が表示される。
- ⑥: 続いて、載貨重量を設計条件を満足する38100t

にするために必要な長さの設定値が計算され、それが、長さをそのように設定したときの他の項目の値とともに、ネットワーク上に表示される。

- ⑦: 長さを⑥に従って、188.85mとすると、再設計が行われ、新しい載貨重量値が決定され、その結果が前回の結果と対比して表示される。表中には、長さを増加させたこと(C+)、載貨重量が増加した(+)ことなどが示されている。
- ⑧: ②と同様に評価診断を行うと、上記の問題点が解消されたことが確認できる。

以上のように、ネットワークの表示や数式微分を利用した適切な設計モデルの予測提案機能により、設計におけるモデルの生成過程を有効に支援することができる。

## 5. 結 言

本研究では、汎用設計知識処理システム<sup>(2)</sup>に対して設計プロセスにおけるモデルの生成過程をより有効に支援するために、設計パラメータ間の依存関係に対応したネットワークを表示する機能と、数式微分を用いて適切な設計パラメータの値を提案する機能を追加した。さらに、これを船舶の基本設計過程の適用して、その有効性を検証した。

設計は、要求される機能に対してモデルを仮定し、それに試行錯誤的に修正を加えながら、設計解を求める逆問題であり、いかに効率よく修正を加えるかが重要となる。本手法によるネットワークの表示機能は、修正すべきパラメータの選定に対して有効な指針を与えることができ、また、設定値の提案機能は、ほぼ所望の機能を有する設計モデルを導くことができる。さらに、設計における各項目間の関係をビジュアルに表示することにより、ユーザは、生成されたモデルの背後にある設計問題の構造を把握することができ、対話的に有効な設計支援を行うことができる。今回導入した諸機能は、船舶に限らず、種々の設計対象に対して有効であると考えられる。

## 文 献

- (1) Sriram, D., and Adeg, R., ed., *Applications of Artificial Intelligence in Engineering Problems*, I, II, 1st Int. Conf., (1986) Spring-Verlag.
- (2) 赤木, 藤田, 機論C編掲載予定.
- (3) 赤木, エンジニアリングシステム設計工学, (昭57), 141, 共立出版.
- (4) 例., 鈴木編, オブジェクト指向, (昭60), 共立出版.
- (5) 赤木, 藤田, 関西造船協会誌, 206, (昭62.9), 掲載予定.
- (6) 溝口, エキスパート・システム—理論と応用, (昭60), 42, 日経マグローウヒル.
- (7) 池田, 山本, マイコンピュータ, 15, (昭59), 52.
- (8) 例., 富田, 船舶基本設計論, (昭57), 丸善出版サービスセンター.

## 討 論

〔質問〕 多田幸生〔神戸大学工学部〕

設計過程の支援のために有用な考え方とシステムを示されたことに敬意を表する。

ある評価項目がいくつかの設計項目に依存していることがわかったとき、どの項目を修正すれば最も効果的であるか、あるいは、一つの項目を修正するだけではだめなときに、どの二つを修正すればよいのか、ということについての支援もエキスパートシステムとしては必要と考えられるが、このシステムを使用するときは、どのような処置が考えられるか。

〔回答〕 ご指摘のように、一般にある評価項目に対して複数の設計項目が関連している場合は多く、そのような場合に、そのうちのどの設計項目に修正を加えるか、さらに、複数の設計項目に対して同時に修正

を加えるような場合にどのように対処するかは、重要な問題である。本手法では、前者については、ネットワークのビジュアルな表示を参考にして設計者（ユーザ）が判断するようになっている。後者についても、設計項目を入れ換えながら徐々に修正を加えていくなどの方法によらざるをえない。しかし、一般に設計過程では、過去の設計事例を参考にすることで初期設計解を設定し、それに修正を加えていくため、設計項目の修正量は比較的小さい。したがって、上記のいずれの場合においても、本手法における各項目間の関係のビジュアルな表示を参考にすることによって、従来に比べてより有効、かつ確実に、設計者が設計解に修正を加えていくことが可能である。