システム分割法による大規模な システムの周波数応答解析†

大野 秀 嶺*・相 楽 和 男**

* 三菱重工業（株）技術本部技術管理部 東京都千代田区丸の内 2-5-1

** 三菱重工業（株）技術本部長崎研究所 長崎市飽ノ浦町 1-1

（昭和 45 年 5 月 19 日 受付）

A Method of Analysis of Large Scale Control Systems in Frequency Domain by System-Decomposition Technique

Hidemine OHNO* and Kazuo SAGARA**

(* Technical Administration Dept., Mitsubishi Heavy Industries Ltd., Chiyoda-ku, Tokyo)
(** Nagasaki Technical Institute, Mitsubishi Heavy Industries Ltd., Nagasaki)

(Received May 19, 1970)

A method of analysis of multi-variable control systems by digital computer is discussed in this paper. A large scale system can be decomposed into a suitable number of subsystems which are represented by signal flow graphs. In this method, the frequency characteristics of each subsystem is calculated by matrix elimination in the first place. Then, a matrix equation of a composed system is made up by rearranging elements of transfer matrices of the subsystems. From this equation, the transfer matrix of the higher class system can be evaluated by matrix elimination, just the same as in the signal flow graph reduction.

This method has much computational advantage to save the time for computation and to use the memory space effectively, as compared with the reduction without the system-decomposition technique. If we introduced a hierarchical system-decomposition or grouping, the computational advantage would be increased remarkably. Especially, a case study, in which parameters of the system are changed, can be carried out with little additional time for computation through an adequate system-decomposition. For this purpose the routine for reduction is proposed in this paper.

In the computer program developed by the authors, the input data can be made up directly in the tabular form from a block-connection diagram and signal flow graphs of a given system.

As an example, a steam temperature control system of reheat boiler is presented to demonstrate the computational advantage of this method.

1. は し が き

制御系の周波数応答の計算には Bode 線図や Nichol's 線図などによる図式的方法が広く使われてきたが、最近ではデジタル計算機も多く使われている。制御系の過渡応答計算には MIDAS¹⁾, MDAS²⁾, DDS³⁾, CSMP⁴⁾ などのように、システムの構成を示すブロッ

ク線図から直接インプットデータを作成するはん用プログラムが開発されているが、周波数応答計算についてもそれらと同じ形式のインプットデータ作成法をとったはん用プログラムがある^{5), 6)}。

筆者らは筆者らの 1 人が報告した周波数応答解析法⁵⁾を拡張し、大規模なシステムの解析を有利にした方法について報告する。

従来の方法^{5), 6)}では、システムを構成する各ブロックの出力を未知変数とする多元連立方程式を作り、伝

† IFAC Symposium on Multi-variable Control Systems, Duesseldorf で発表 (昭 43・10)

達関数の複素変数 s を $j\omega$ において、システムの入出力間の周波数トランスミッタンスを行列消去により求める。その場合、各ブロックの出力を未知変数として、ブロックの伝達関数が方程式の係数行列の対角線要素に含まれるようになる。そのためにシステムの規模が大きくなると係数行列の次元が大きくなり、演算時間、記憶容量、計算精度などの点で実用上の限界がでてくる。

筆者らが報告する方法では、まず、システムの構成を信号流れ線図 (signal flow graph) で示し、節点 (node) を変数として扱うことにより、方程式の係数行列の非対角線要素にも分枝 (branch) の伝達関数が多く含まれるようにする。そして、直列あるいは並列接続の分枝の中間の節点をあらかじめ消去した形で係数行列を構成する。これらの改良を加えることにより、消去すべき行列の次元はかなり小さくなる。この消去演算を信号流れ線図の消去 (signal flow graph reduction) と呼ぶことにする。

さらに、筆者らは“大規模なシステムを物理的現象の観点からいくつかのサブシステムに分割すると、それらはたがいに少数個の変数を媒介にして結合されている”ということに注目し、そのようなシステムの周波数応答解析に次の方法を採用した。

(1) 分割されたそれぞれのサブシステムに信号流れ線図の消去を行ない、媒介変数を入出力とする周波数伝達行列を求める。

(2) (1) で求めた周波数伝達行列の要素を組みかえて、もとのシステムの方程式を作り、それを解いて必要な入出力間の周波数伝達行列を求める。

これらの方法を使うと、大きなシステムに対しても、消去演算が行なわれる行列の次元は小さくなり、演算時間の短縮と記憶容量の有効利用が可能になる。

そこで、Fig. 1 のようにシステム L を中サイズ M_p 、小サイズ S_{pq} のサブシステムに分割して、変数の消去を段階的に行なうと、この方法の演算上の効果は倍加される。とくに、ケーススタディを行なう場合には、変更するパラメータを含む分枝を1つのサブシステムに集めるようにシステムを分割すると、それに直接関

Fig. 1 A large scale system composed of a suitable number of subsystems

係のないサブシステムの消去演算は省略することができ、計算時間は著しく短縮される。

例として、再熱ボイラの蒸気温度制御系の例をあげ、この周波数応答計算法の演算上の効果について述べる。

2. 線形多変数システムの信号流れ線図の消去

2.1 原理

Fig. 2 Generalized partial flow graph

線形システムは例題 (Fig. 11) に示すように、詳細な構成を信号流れ線図で示すことができる。そこで一般的に考え、信号流れ線図で表わしたシステムの入力を $u_1, u_2, \dots, u_m$ 、出力節点を $z_1, z_2, \dots, z_{n_0}$ としよう。そ

して出力を含むシステムの未知変数を $x_1, x_2, \dots, x_n$ ($n \geq n_0$) とすると、任意の未知変数 x_i に注目した部分的信号流れ線図は Fig. 2 のように示される。

Fig. 2 の x_i は、

$$x_i(s) = \sum_{j=1}^n g_{ij}(s)x_j(s) + \sum_{j=1}^m f_{ij}(s)u_j(s) \quad (1)$$

で示され、これを書き換え、

$$\sum_{j=1}^n a_{ij}(s)x_j(s) = \sum_{j=1}^m f_{ij}(s)u_j(s) \quad (2)$$

と表わす。ただし、

$$a_{ij}(s) \triangleq \begin{cases} -g_{ij}(s) & \text{for } i \neq j \\ 1 - g_{ii}(s) & \text{for } i = j \end{cases} \quad (3)$$

(2) 式の関係は n 個の未知変数について考えられるので、行列で示すと、

$$[a_{ij}(s)][x_i(s)] = [f_{ij}(s)][u_j(s)] \quad (4)$$

ただし、 $[a_{ij}(s)]$: (3) 式で示す $a_{ij}(s)$ を要素とする行列 ($n \times n$)、 $[f_{ij}(s)]$: $f_{ij}(s)$ を要素とする行列 ($n \times m$)、 $[x_i(s)]$: $x_i(s)$ を成分とする n 元の列ベクトル、 $[u_j(s)]$: $u_j(s)$ を成分とする m 元の列ベクトル。

(4) 式から、

$$[x_i(s)] = [a_{ij}(s)]^{-1}[f_{ij}(s)][u_j(s)] \quad (5)$$

ここで、

$$[t_{ij}(s)] \triangleq [a_{ij}(s)]^{-1}[f_{ij}(s)] \quad (6)$$

を定義すれば、 $[t_{ij}(s)]$ は入力 $[u_j(s)]$ 、出力 $[x_i(s)]$ 間の伝達行列 ($n \times m$) であり、(5) 式は次のように書かれる。

$$[x_i(s)] = [t_{ij}(s)][u_j(s)] \quad (7)$$

ここで、システムの真の出力は $z_1, z_2, \dots, z_{n_0}$ で、それらは $x_1, x_2, \dots, x_n$ の一部であるから、真の出力に

関する伝達関数 $t_{zij}(s)$ を取り出すと、次式が得られる。

$$[z_i(s)] = [t_{zij}(s)][u_j(s)] \quad (8)$$

(6) 式から入力 $u_j(s)$ 、出力 $x_i(s)$ 間のトランスミッタンス $t_{ij}(s)$ が求められるわけであるが、一般に分枝の伝達関数 $g_{ij}(s)$ 、 $f_{ij}(s)$ は複素変数 s に関する有理式となる場合が多いが、超越関数 e^{-Ls} (むだ時間要素) を含む場合もあるので、直接的に (6) 式から $t_{ij}(s)$ を求める演算はやっかいである。そこで複素変数 s を $j\omega$ とおいて、

$$[t_{ij}(j\omega)] = [a_{ij}(j\omega)]^{-1} [f_{ij}(j\omega)] \quad (6')$$

として、周波数伝達行列を求めることを考える。 $[t_{ij}(j\omega)]$ は、

$$[a_{ij}(j\omega)][t_{ij}(j\omega)] = [f_{ij}(j\omega)] \quad (9)$$

を代数方程式とし、行列消去により求められる。

そして、必要な入力 $[u_j(j\omega)]$ 出力 $[z_i(j\omega)]$ 間の伝達行列 $[t_{zij}(j\omega)]$ は z_i と x_i の対応をつけ、 $[t_{ij}(j\omega)]$ から、 z_i 以外の行を捨てて残った行列として求められる*1。そこで、

$$[z_i(j\omega)] = [t_{zij}(j\omega)][u_j(j\omega)] \quad (8')$$

が変数消去したシステムの入出力の関係となる。

(9) 式から $[t_{ij}(j\omega)]$ を求める行列消去をデジタル計算機で行なう場合の演算時間は行列の次元 n が大きくなると相乗的に長くなる。したがって、 n をできるだけ減らした形で行列消去を行えば演算時間は短縮される。また、計算機の記憶容量を有効に利用するという点でも有利である。そこで、筆者らは次に述べる 2 つの方法を採用し、 n を小さくするようにした。

Fig. 3 Partial flow graph with a direct path g_{ii}

を組むことにした。

Fig. 3 の x_i は、

$$x_i(s) = g_{si}(s) \left\{ \sum_{j=1}^{n'} t_{ij}(s) x_j(s) + \sum_{j=1}^m f_{ij}(s) u_j(s) \right\} \quad (1')$$

となる。ここで、(3) 式の $a_{ij}(s)$ の代わりに、

$$a_{ij}(s) \triangleq \begin{cases} -g_{ij}(s) & \text{for } i \neq j \\ g_{si}^{-1}(s) - g_{ii}(s) & \text{for } i = j \end{cases} \quad (3')$$

*1 数式的には $[x_i]$ と $[z_i]$ の変換行列を掛けることになり、その取り扱いについては 3・1 で述べる。

を用いれば、(4) 式と同じ形式の関係を得る。(1') 式の n' は (1) 式の n から x_i' のような節点の個数を引いたものである。したがって、伝達行列を求めるための方程式 ((9) 式) の行列は $[a_{ij}(j\omega)] (n' \times n')$ 、 $[f_{ij}(j\omega)] (n' \times m)$ と小さくなる。また、この方法をとれば、 $g_{si}(j\omega)$ について $f_{ij}(j\omega) g_{si}(j\omega)$ と $g_{ij}(j\omega) g_{si}(j\omega)$ の $(n+m)$ 回の乗算をせずに、 $g_{si}(j\omega)$ の逆数 $g_{si}^{-1}(j\omega)$ を求め、(3) 式の $a_{ii}(s)$ の代わりに (3') 式の $a_{ii}(s)$ の算式を用いればよいことになる。

さらに、Fig. 4 のような分枝の並列接続や直列接

(a) $g_{ij} = g_1 + g_2 (NC=1)$ (b) $g_{ij} = g_1 g_2 (NC=2)$

Fig. 4 Reduction of intermediate node

続についても中間変数 x_i' があらかじめ消去される。すなわち、デジタル計算機を使えば、はじめに、

$$g_{ij}(j\omega) = g_1(j\omega) \quad (10.1)$$

とおき、並列接続の場合には、

$$g_{ij}(j\omega) = g_{1j}(j\omega) + g_2(j\omega) \quad (10.2)$$

とし、直列接続の場合には、

$$g_{ij}(j\omega) = g_{1j}(j\omega) g_2(j\omega) \quad (10.2')$$

と処理することができる。Fig. 5 の例では $g_1, g_2, \dots, g_6$ の順序と次々の演算が加算か乗算かの区別を指示

$$g_{ij} = [(g_1 + g_2)g_3 + g_4]g_5g_6$$

Fig. 5 An example of reduction of intermediate nodes

(インプットデータ NC による) すれば、中間変数 x_1', x_2', x_3' は $g_{ij}(j\omega)$ の計算が完了したときに消去されている。

2.2 計算方法

与えられた線形システムを信号流れ線図で表わし、入力の節点 (source) を $u_1, u_2, \dots, u_m$ 、未知変数の節点を $x_1, x_2, \dots, x_n$ と符号化する。そこで、筆者らは分枝の伝達関数の標準的タイプとして Table 1 のようなものを考え、コードにより $g(s)|_{s=j\omega}$ の算式を区別するようにした。その算式はサブプログラム (BRANCH) の中に記憶され、新形式の伝達関数を必要とするときはその算式はサブプログラムの中に簡単に追加される。

筆者らのプログラムでは、信号流れ線図の消去のため、次のようなデータを数表形式で与える。

a. 未知変数 x_i 、入力変数 u_j 、出力 z_i などの個

Table 1 Standard types of transfer functions

code	formula of $g(s)$
1	d_1
2	$\frac{d_1+d_2s}{d_3+d_4s+d_5s^2}$
3	$d_1 \frac{1+d_2s+d_3s^2}{1+d_4s+d_5s^2}$
4	$\frac{d_1(1+d_2s)}{1+d_3s+d_4s^2+d_5s^3}$
5	$\frac{d_1}{s^2+2d_2d_3s+d_3^2}$
8	$\frac{d_1 \exp(-d_2s)}{1+d_3s}$
9*	$d_1 \exp\{-\psi(s)\}$
10*	$d_1 [1-\exp\{-\psi(s)\}]/\psi(s)$
11**	$d_1 \exp\{-\psi'(s)\}$
12**	$d_1 [1-\exp\{-\psi'(s)\}]/\psi'(s)$
0	skip over the subroutine BRANCH and set the value at the last one of $g(s)$
-1	skip over the subroutine BRANCH and multiply the last value of $g(s)$ by -1.0

note : * $\psi(s) = d_2s + \frac{d_3s}{d_3s + d_5}$

** $\psi'(s) = d_2s + d_3s + \frac{d_3(d_4 \cdot d_5)s}{d_5s + 1}$

数 n, m, n_0 , 分枝 g_{ij}, f_{ij}, g_{si} などの個数 N, M, N_s , 計算すべき周波数 ω の数 K .

b. 分枝 g_{ij}, f_{ij} について, 行先と出発の符番 i, j , 伝達関数のコード, その定数 $d_1, d_2, \dots, d_5$. 分枝 g_{si} については, 行先節点の i , 伝達関数のコードと定数.

c. 出力変数 z_i と未知変数 x_i の対応関数.

d. 計算すべき周波数 ω の指定.

以上のデータを与え, 各周波数 ω ごとに伝達行列 $[t_{ij}(j\omega)]$ が次の順序で計算される.

1) (9) 式の行列 $[a_{ij}(j\omega)]$ ($n \times n$), $[f_{ij}(j\omega)]$ ($n \times m$) の要素をすべて 0 にセットする.

2) $g_{ij}(s)$ の伝達関数のコードにより, $g(j\omega)$ の算式を選択し, 定数 $d_1, d_2, \dots, d_5$ を用いて $g(j\omega)$ を計算し, $a_{ij}(j\omega) = -g(j\omega)$ とおく. 同様に $f_{ij}(s)$ について $g(j\omega)$ を計算し, $f_{ij}(j\omega) = g(j\omega)$ とする. 次に $g_{si}(s)$ について $g(j\omega)$ を計算し, $g_{si}(j\omega) = g(j\omega)$, $a_{ii}(j\omega) = a_{ii}(j\omega) + g_{si}^{-1}(j\omega)$ とする. 該当する $g_{si}(s)$ のないとき $g_{si}(j\omega) = 1$.

3) (9) 式を行列消去して $[t_{ij}(j\omega)]$ を求める.

4) c. のデータを使って $[t_{ij}(j\omega)]$ から $[t_{zij}(j\omega)]$ を編成.

以上の説明では Fig. 4 の中間変数の消去について省略しているが, それを行なう場合には, b. のデータに次の b'. を追加する.

b'. 並列接続の場合 $NC=1$, 直列接続の場合 $NC=2$. これにより, (10.2) あるいは (10.2') 式と同様な演算が行なわれる.

また, Table 1 の伝達関数のコード 0, -1 は次のように用いられ, 上述 2) において異なる分枝で伝達関数の内容が同じ場合の $g(s)|_{s=j\omega}$ の重複演算を省略する.

コード 0 : 1つ前で計算した分枝の $g(j\omega)$ と同じ値とする.

コード -1 : 1つ前で計算した分枝の $g(j\omega)$ の値に -1 を掛ける.

このコード 0 と -1 を用いると, Table 1 のコード 8~12 のように, e^{-Ls} , $e^{-\psi(s)}$ などを含む伝達関数の重複計算を省き演算時間の短縮に役立つ.

3. サブシステムを結合して作られるシステムの消去

3.1 原理*2

ここでは Fig. 6 のようにサブシステム S_q を数個結合して作られるシステム M の消去について考える.

Fig. 6 An example of a system (M) composed of three subsystems (S_1, S_2, S_3)

サブシステム S_q の入力 $[u_{qj}]$ と出力 $[z_{qi}]$ との関係,

$$[z_{qi}] = [t_{qij}] [u_{qj}] \quad (11)$$

と示す. この関係を信号流れ線図の消去で与えられる (8) 式の関係と同じもので, t_{zij} の添字 z を省略し, サブシステム S_q に関するものには添字 q を付けて示す. また, Fig. 6 の変数消去を説明していく便宜上, $[z_{qi}]$, $[u_{qj}]$, $[t_{qij}]$ を z_q , u_q , t_q と表わすことにする. すなわち, (11) 式は次のように示す.

*2 ここでは周波数応答を扱うことは明らかであるので, 複素変数 s と区別するための $j\omega$ を省略する.

$$z_q = t_q u_q \quad (11')$$

ここで、次のベクトルを考える。

$$z \triangleq \begin{bmatrix} z_1 \\ z_2 \\ z_3 \end{bmatrix}, \quad u \triangleq \begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix} \quad (12)$$

すなわち、(11'), (12) 式から、

$$\begin{bmatrix} z_1 \\ z_2 \\ z_3 \end{bmatrix} = \begin{bmatrix} t_1 u_1 \\ t_2 u_2 \\ t_3 u_3 \end{bmatrix} = \begin{bmatrix} t_1 & 0 & 0 \\ 0 & t_2 & 0 \\ 0 & 0 & t_3 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix} \quad (13)$$

この関係は Fig. 6 でサブシステム S_1, S_2, S_3 を結合する前の状態を示し、Fig. 7 (a) のようなブロック図で示される。

Fig. 7 Reduction of composed system

そこで、

$$t \triangleq \begin{bmatrix} t_1 & 0 & 0 \\ 0 & t_2 & 0 \\ 0 & 0 & t_3 \end{bmatrix} \quad (14)$$

を定義すれば、(13) 式は次のように書き換えられる。

$$z = tu \quad (13')$$

ここで、Fig. 6 のようにサブシステム S_q の出力 z_{qi} をシステム M の未知数 X_v と対応づけることは、変数変換、

$$X = C_{Xz} z \quad (15)$$

で示される。ここで、 X は X_v を成分とするベクトルで、 C_{Xz} は $z \rightarrow X$ への変換行列である。そして、 z_{qi} は X_v とは一对一の対応関係があるので、 C_{Xz} は要素が 0 または 1 となる正方行列で、各行、各列に 1 がただ 1 個あり $\det C_{Xz} = 1$ となる。それゆえ (15) 式は、

$$z = D_{Xz}^{-1} X \quad (15')$$

とも表わされる。

いっぽう、サブシステム S_q の入力 u_{qj} はシステム M の入力 V_μ あるいは未知変数 X_v のいずれかに対応する。 V_μ を成分とするベクトル V を考えると、

$$u = C_{uv} V + C_{ux} X \quad (16)$$

と示される。ここで、 C_{uv}, C_{ux} は u_{qj} と V_μ, u_{qj} と X_v の対応関係を示す変換行列である。

(13), (15'), (16) 式の関係は Fig. 7 (b) のように示される。したがって、Fig. 7 (b) で、

$$(I - C_{Xz} t C_{ux}) X = C_{Xz} t C_{uv} V \quad (17)$$

なる関係が成立する。ただし、 I は単位行列とする。

そこで、

$$A \triangleq I - C_{Xz} t C_{ux}, \quad F \triangleq C_{Xz} t C_{uv} \quad (18)$$

を定義すると、(17) 式は、

$$AX = FV \quad (19)$$

と書かれ、これを解けば、

$$X = A^{-1} F V = T V \quad (20)$$

ただし、 T は入力 V 、出力 X としたシステム M の伝達行列で、次のように定義される。

$$T \triangleq A^{-1} F \quad (21)$$

すなわち、伝達行列 T は、

$$AT = F \quad (22)$$

の行列消去により求められる。この A と F は (18) 式に示すもので、サブシステムの伝達行列 t_q および変数の変換行列 C_{Xz}, C_{uv}, C_{ux} が与えられるとその行列は決まる。

さらに、Fig. 7 で未知変数 X_v の中から出力 Z_i を選択することは、 $X \rightarrow Z$ の変換行列 C_{ZX} を使い、

$$Z = C_{ZX} X = C_{ZX} T V \quad (23)$$

と示され、

$$T_Z = C_{ZX} T \quad (24)$$

が、入力 $V = [V_\mu]$ 、出力 $Z = [Z_i]$ 間の伝達行列である。

3.2 計算方法

この消去の計算は (22) 式の行列 A, F の要素をセットし、それを行列消去法で解き T を求め、さらに (24) 式の変換行列 C_{ZX} を掛けて T_Z を求めることである。

ここで、行列 A と F は (18) 式で示され、その中の変換行列 C_{Xz}, C_{ux}, C_{uv} の要素は 0 または 1 であるから、 $C_{Xz} t C_{ux}$ と $C_{Xz} t C_{uv}$ は伝達行列 t の要素 t_{qij} の入れ換えとして実行される。また、(24) 式の変換も同じことである。

そこで、次のデータを与える。

- C_{Xz} : S_q の出力 z_{qi} と M の未知変数 X_v の対応関係
- C_{ux} : S_q の入力 u_{qj} と M の未知変数 X_v の対応関係
- C_{uv} : S_q の入力 u_{qj} と M の入力 V_μ の対応関係
- C_{ZX} : M の未知変数 X_i と M の出力 Z_i の対応関係

この a. と b., b.' のデータの記入例を Table 2 の $i_X(i), j_u(j)$ に示した。b., b.' において、 u_{qj} は X_j あるいは V_μ のいずれか一方にのみ対応するので、b. の $j_u(j)$ にはマイナス符号をつけ、b.' と区別する。

このようなデータを与えてサブシステム M の変数消去は Fig. 8 のように行なわれる。ただし、Fig. 8 と

Fig. 8 Schematic diagram of transfer matrix reduction

Table 2 An example of input data for transfer matrix reduction

7	5	1	4							$i_x(i)$
1	2	3	6							$i_x(i)$
2	3	-7	-5	-4						$j_u(j)$

Table 2 の $i_x(i)$, $j_u(j)$, $T(q, i, j)$ の i, j は S_q の z_{qi} , u_{qj} , $t_{qi,j}$ の添字 i, j を示し, ν と μ は X_ν と V_μ の添字を示す。

Table 2 のデータ記入例は Fig. 6 のサブシステム S_1 に関するもので, $i_x(i)$ の 1, 2, 3, 6 は $z_1 \sim z_4$ が X_1, X_2, X_3, X_6 となり, $j_u(j)$ の 2, 3, -7, -5, -4 は $u_1 \sim u_5$ が V_2, V_3, X_7, X_5, X_4 となることを示す。Table 2 の $i_x(i)$ は消去前のサブシステム S_q の未知変数 x_i と出力 z_i との関係を示している。すなわち, $i_x(i)$ と $i_x(i)$ との対応はサブシステム S_q の x_i とシステム M の X_i とを関係づけている。システム M の出力 Z_i を指定するインプットデータもこの $i_x(i)$ と同様の形式で与える。

3.3 システムの段階的消去による伝達行列の計算

Fig. 1 のように1つのシステム L をサブシステム M_p, S_{pq} に分割した場合には, 3.1, 3.2 に述べた方法を S_{pq} 結合して作った M_p に適用し, さらに M_p を結合して作った L についても適用する段階的なシステムの消去ができる。その段階的な消去の手順は概念的に Fig. 9 で示される。ここで, Fig. 9 の S_{pq} では 2.2 に述べた計算手順をサブプログラムとして使い, M_p と L では Fig. 8 の計算をサブプログラムとする。

一部のパラメータを逐次変更し, 同じような計算をくり返すケーススタディを行なうことが多い。その場合, 変更されるパラメータを含む部分が1つの小サブシステム S_{ch} に集中するようにシステムを分割しておく, それ以外のサブシステム S_{pq} と M_p はパラメータの組み合わせを変えても同一の周波数では伝達行

Fig. 9 Skelton diagram of main program

列は変わらない。それゆえ, パラメータ変更1回につき, サブシステム S_{ch} とシステム L の計算を追加すればよいことになる。

このケーススタディのために, 筆者らのプログラムでは, 分枝の伝達関数のタイプと定数とを変更可能とし, その変更に必要な分枝のデータだけを全体のデータの最後に付け加えるという方法をとっている。

このシステム分割法には, 以上のような段階的消去とケーススタディの合理的方法が容易に採用できる。

4. 応 用

例として, 再熱ボイラの蒸気温度制御系の解析について述べる。そのシステムのブロック接続図は Fig. 10 のようになり, そのうちの1つの小サブシステム S_{31} の信号流れ線図を Fig. 11 に示す。

この例では蒸気温度制御のコントローラを除いた制御対象の周波数特性を求めようとし, システムへの入力と出力を次のとおりとする。

入力: 蒸気流量 (f), ガス再循環ダンパ (φ_R), 減温器スプレー弁開度 (φ_{sp})

出力: 減温器後蒸気温度 (θ_{SH1}), 過熱器出口蒸気温度 (θ_{SH2}), 再熱器出口蒸気温度 (θ_{RH})

そして, 過熱器 (SH) と再熱器 (RH) とを詳細なモデルで扱うことを試み, 前者をパネル (S_{21}), プラトン (S_{22}), 中間 (S_{23}), 最終 (S_{31}) の各過熱器の4セクションに分割し, 後者を横置 (S_{33}), 最終 (S_{32}) の各再熱器のセクションに分割した。ここで, S_{21} , S_{22} , S_{23} には火炉のふく射による熱入力 (q_r) があり, それらの信号流れ線図は類似しているので中サブシステム M_2 にまとめた。 S_{31} と S_{32} とはまったく同じ信号流れ線図 (定数は異なる) となるのでそれらに S_{33} を加えて M_3 にまとめた。また, $M_1(S_{11})$ はドラム, 降水管, 蒸発管, 火炉 (燃焼制御系も含める) などの部分で簡単なモデルで扱い, こまかく分割しな

Fig. 10 Block-connection diagram of STC system of Reheat boiler (controller removed)

◎ u_j : input signals ● x_i : unknown variables
○: intermediate variables ○: output signals

$G_1 = (1 - e^{-\phi'(s)})/\phi'(s)$, $G_2 = a_1/(1 + T_1 S)(1 + T_2 S)$,
 $G_3 = b_1(b_2 - e^{-\phi(s)})/(\phi(s) - b_3)$, $G_4 = (1 + T_3 S)/(1 + T_1 S)$,
 $G_5 = C_1 S/(1 + T_3 S)$, $G_6 = C_2(1 + T_4 S)$,
 $G_7 = (1 + T_5 S)/(1 + T_2 S)$, $G_8 = 1/(1 + T_1 S)$,
 $G_9 = a_1'/(1 + T_2 S)$, $G_{10} = (1 + T_5 S)/(1 + T_2 S)$,
 $G_{11} = e^{-\phi'(s)}$, $G_{12} = a_1'/(1 + T_2 S)$

Fig. 11 Signal flow graph of FIN. SH (S_{31}) in Fig. 10

かった。ドラム出口から再熱器出口の間には過熱器の減温器と主蒸気管・高圧タービンが含まれるが、前者を S_{22} に、後者を S_{33} に、それぞれ簡単化したモデルにして含ませた。

以上のようにシステムの分割を行ない、Fig. 10 のブロック接続図から Table 2 の形成でサブシステムの結合関係を規定するインプットデータを与える。また、Fig. 11 のような信号流れ線図を各小サブシステムに描き、2・2 の (1)～(3) 項のデータを作表する。この例では、 S_{31} と S_{32} の信号流れ線図はまったく同形になり、そのほかの S_{21} , S_{22} , S_{23} , S_{33} について類似の形になる。このことは信号流れ線図の作図が容易なばかりでなく、計算のためのインプットデータをチェックするのも容易になる。

2・1 において、信号流れ線図からの縮体で演算時間短縮のために中間変数を消去していくことを述べたが、この例では Fig. 4 による方法がおおいに活用される。

Fig. 11 で x_i の記号を付けた節点の個数 n は 8 であるが、 x_i の符号のない節点 (白丸) は 15 個ある。したがって、Fig. 4 の方法で中間変数を消去して行かないと $n=23$ となって、その中間変数消去の有効なことがわかる。

また、2・2 において、分枝の伝達関数のタイプを指定する code に 0 と -1 を設けたことを述べたが、それもこの例題では大いに有効である。すなわち、Fig. 11 で分枝の整理番号として、○印で囲んで示した数字を用いたが、①, ②, ③, ④ はまったく同じ伝達関数 G_1 となり、 G_1 は Table 1 の code 12 のタイプでかなり複雑な形をしている。そこで分枝 ②, ③, ④ の code を 0 と記入すれば、分枝 ① について計算した結果がそのまま ②, ③, ④ に対して使われる。また、分枝 ④ と ⑤, ⑥ と ⑦ は正負の符号を反転しただけの違いであるから、code を -1 とするだけで分枝 ⑤ と ⑦ のトランスミッタンスが求められる。

この蒸気温度制御系の解析で縮体されるシステムの規模を表わす目安として、サブシステム S_{pq} , M_p とシステム L の変数の個数 n , m , 分枝の個数 N , M を一括して Table 3 に示した。行列消去に要する演算時間と行列のサイズとの関係は消去演算のサブプログラムの作り方に依存し複雑であるので、厳密な検討をしていないが、システム分割法が演算時間の短縮にどれだけ効果があるか、次のような簡単な試算を行なった。ここで、行列消去のための演算時間は未知変数 n の 2 乗に比例すると仮定すれば、

(1) システム分割を行なう場合

$$\sum (n \text{ in } S_{pq})^2 + \sum (n \text{ in } M_p)^2 + \sum (n \text{ in } L)^2 = 806$$

(2) システム分割をしない場合

$$\{\sum (n \text{ in } S_{pq})\}^2 = 4356$$

となる。これはひじょうに大まかな目安であるが、システム分割法が演算時間の短縮に有効であることは明らかである。この行列消去の演算時間が n^3 に比例す

Table 3 Size of matrices to be reduced for the system in Fig. 10

	n	m	N	M	n_0
$M_1(S_{11})$	9	3	12	7	5
S_{21}	11	5	40	5	2
S_{22}	11	5	40	5	2
S_{23}	12	6	41	7	3
M_2	7	6	—	—	3
S_{31}	8	4	35	4	2
S_{32}	8	4	35	4	2
S_{33}	7	4	21	7	1
M_3	5	4	—	—	2
L	8	3	—	—	3
$\Sigma(n \text{ in } S_{pq})=66, \Sigma(N \text{ in } S_{pq})=224$					

るならば、上記 (1), (2) の差はさらに著しくなる*3。

この例題について、周波数 ω を 24 点とし、周波数応答を IBM 7044 を用いて計算した結果、純粹の演算（データ読み込み開始から印字開始まで）は 148 秒で行なわれた。

5. む す び

システムを適当な個数のサブシステムに分割し、段階的に消去し、規模の大きい線形システムの周波数応答を求める方法について述べた。例として、再熱ボイラの蒸気温度制御系をあげ、システム分割法と信号流れ線図の消去における演算上の工夫が大規模なシステムの解析で演算時間短縮に効果のあることを示した。筆者らのプログラムでは他の動特性解析のはん用プログラムと同様に、システムの構成を示すブロック接続図と信号流れ線図から直接的にインプットデータが作表される。また、システムの部分的修正の場合にはその修正個所だけのデータをさしかえる方法をとっている。一般に、システムの動特性解析では一部のパラメータを変更してケーススタディをくり返し行なうことが多いが、このシステム分割法を使うと変更部分のデ

ータをデータリストの終わりに付け加えると、わずかな演算時間の延長でそのケーススタディが行なえる。

この方法により、従来、自動制御系のシンセシスや動特性解析に広く使われてきた周波数応答法をシステムの規模が大きい場合にも適用することが容易になった。とくに、筆者らが行なったボイラタービンプラントの動特性解析⁹⁾のように、外乱の周波数特性が知られており、そのシステムの運転性能を推定しようという場合には有効な方法である。また、周波数応答法は統計的制御理論との結びつきが容易であるので、その分野への応用も期待できる。しかし、大規模なシステムは一般に操作量と制御量の相互干渉がある多変数制御系となるので、この計算法をシンセシスに利用するには多変数制御系を周波数領域で扱うユニークな手法の検討がなお必要であろう。その準備の1つとして、パラメータ感度法的应用を検討しているが⁹⁾、それに関しては改めて報告することにした。

参 考 文 献

- 1) R. T. Harnett ほか: MIDAS, An Analog Approach to Digital Computation, Simulation, 17/43 (Sept.-1964)
- 2) 的場: アナログむき入力言語によるディジタルシミュレータ (MDAS), 電通学誌, 49-7, 1351/1356 (昭 41-7)
- 3) 三巻: ダイナミック・システムのディジタルシミュレーション, 計測と制御, 7-4, 258/268 (昭 43-4)
- 4) R. D. Brennan ほか: Tow Continuous System Modeling Program (CSMP), IBM Systems Journal, 6-4, 242/266 (1967)
- 5) R. G. Abraham ほか: Digital Computers in Control-System Design, Electro-Technology, 75/86 (June-1966)
- 6) 大野: ディジタル計算機による多変数制御系の一方法, 計測と制御, 6-10, 699/709 (昭 42-10)
- 7) H. Ohno: An Analyzing Method of Linear Multivariable Control System Using Digital Computer, IFAC Symposium on Multivariable Control Systems, Duesseldorf (Oct.-1968)
- 8) K. Itoh ほか: Comparison of Dynamics Between Natural Circulation Boiler and Forced Circulation Boiler, 4-th IFAC Congress (June-1969)
- 9) 大野: 多重ループ系のパラメータ感度解析, 第8回計測自動制御学会学術講演会論文集, 95/96 (昭 44-8)

*3 文献 7) では高圧タービン, 再熱器, 中低圧タービンのカスケード系で2段階の消去を行ない, 分割して扱うことにより演算時間が1/3になることを報告している。