スイッチング関数のC形式とその応用

木 村 英 紀

東京大学大学院 東京都文京区本郷 7-3-1

(昭和 42 年 2 月 20 日受付)

C-Form of Switching Functions and Its Applications

Hidehori KIMURA

(Faculty of Engineering, University of Tokyo, Bunkyo-ku, Tokyo)

(Received February 20, 1967)

C-Form of switching function is one of the canonical forms of switching functions, which is defined by the three usual operations (addition, multiplication and scalar multiplication) over the real field, instead of the logical operations. In this paper, we consider some properties of switching functions by using this form and apply them to the analysis of the cycle structure of autonomous sequential circuits. These results are expressed in terms of linear algebra. A method of learning identification for sequential circuits is derived and discussed.

1. 緒 言

任意の n 変数スイッチング関数 $y=f(x_1, x_2, \dots, x_n)$ は、変数として 1 と -1 を用いることにより次のように表現されることがすでに知られている。

$$y = a_0 + a_1 x_1 + a_2 x_2 + a_3 x_3 + \dots + a_{1 \cdot n} x_1 x_2 \dots x_n \quad \dots\dots(1)$$

ここで 2^n 個の係数 $a_0, a_1, \dots, a_{1 \cdot n}$ は f の真理値表から決定される有理数で、 $x_i (i=1, \dots, n)$ および y は 1 または -1 をとる。1 は false, -1 は true に対応する。すなわち、(1) の形式を用いることにより、スイッチング関数が、論理和・論理積・否定といういわば「特殊な」演算を用いないで、通常の三つの演算、和・変数同士の積・スカラー倍、によって表現されている。

本稿では(1)の形式を便宜的に **C形式*** とよび、これを用いて組み合わせ回路や順序回路の種々の性質を研究する。C形式の第1の特徴は、論理演算や写像を行列やベクトルによって表現できるという点である。すなわち、C形式はブール代数をより具体的に取扱いやすい線形代数によって表現したものとみなすことができる。第2の特徴は、C形式が関関数の拡張として“flexible”な性質をもっているという点である。これは3や6における学習の問題に応用されている。

2. スイッチング関数のC形式

2.1 係数ベクトルの計算

すでに述べたように、任意の n 変数スイッチング関数は、(1) 式の 2^n 個の係数を適当に選ぶことにより決められる。たとえば $n=3$ の場合は

$$y = a_0 + a_1 x_1 + a_2 x_2 + a_{12} x_1 x_2 + a_3 x_3 + a_{13} x_2 x_1 + a_{23} x_2 x_3 + a_{123} x_1 x_2 x_3$$

となる。

(1) をベクトルの内積を用いて次のように表現する。

$$y = (a, \hat{x}) \quad \dots\dots(1)'$$

ここで $a, \hat{x}$ はともに 2^n 次元ベクトルで、 $a^T = (a_0, a_1, a_2, a_{12}, \dots, a_{1 \cdot n})$, $\hat{x}^T = (1, x_1, x_2, x_1 x_2, \dots, x_1 x_2 \dots x_n)$ である。 a を係数ベクトルとよぶことにする。係数ベクトルを真理値から決定するには次のようにする。

今、 $f(x_1, \dots, x_n)$ の $(x_1, \dots, x_n)$ に対抗する f の値を、 $(x_1, \dots, x_n)$ が表現する2進数の小さい順番に並べてできる 2^n 次元ベクトル $l(f)$ とすると、次の定理がすでに知られている。

$$\text{定理 1} \quad a = 2^{-n} H_n l(f) \quad \dots\dots(2)$$

ただし H_n は次のように逐次的に構成される $2^n \times 2^n$ 行列である。

$$H_1 = \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}, \quad H_2 = H_1 \times H_1 = \begin{pmatrix} H_1 & H_1 \\ H_1 & -H_1 \end{pmatrix},$$

$$\dots\dots\dots, \quad H_n = \begin{pmatrix} H_{n-1} & H_{n-1} \\ H_{n-1} & -H_{n-1} \end{pmatrix}.$$

(証明は付録1参照)

* (1) の 2^n 個の係数は、I. Ninomiya¹⁾ によるブール関数の coordinate representation と定数倍を除いて一致する。C形式のCは coordinate の頭文字である。

(例) Table 1 で与えられた関数のC形式をもとめる.

x_1	x_2	x_3	y
0	0	0	1
1	0	0	0
0	1	0	0
1	1	0	1
0	0	1	1
1	0	1	0
0	1	1	0
1	1	1	0

Table 1 An example of switching functions

x_1	x_2	x_3	y
1	1	1	-1
-1	1	1	1
1	-1	1	1
-1	-1	1	-1
1	1	-1	-1
-1	1	-1	1
1	-1	-1	1
-1	-1	-1	1

Table 2 Transformation of variables from (0,1) to (1,-1)

(2)式を用いると

$$\begin{pmatrix} a_0 \\ a_1 \\ a_2 \\ a_{12} \\ a_3 \\ a_{13} \\ a_{23} \\ a_{123} \end{pmatrix} = 2^{-3} \begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 & 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 \\ 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 \\ 1 & -1 & 1 & -1 & -1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 & -1 & -1 & 1 & 1 \\ 1 & -1 & -1 & 1 & -1 & 1 & 1 & -1 \end{pmatrix} \begin{pmatrix} -1 \\ 1 \\ 1 \\ -1 \\ -1 \\ 1 \\ 1 \\ 1 \end{pmatrix} = \begin{pmatrix} 1/4 \\ -1/4 \\ -1/4 \\ -3/4 \\ -1/4 \\ 1/4 \\ 1/4 \\ -1/4 \end{pmatrix}$$

したがって

$$y = \frac{1}{4} (1 - x_1 - x_2 - 3x_1x_2 - x_3 + x_1x_3 + x_2x_3 - x_1x_2x_3)$$

2.2 係数ベクトルの性質

二つの n 変数スイッチング関数 $y_1(x) = (a, \hat{x})$ と $y_2(x) = (b, \hat{x})$ の積 $y = y_1(x) \cdot y_2(x)$ は、変数 (0,1) で考えれば mod-2 和 $y = y_1(x) \oplus y_2(x)$ に対応する. これを

$$y = (C, \hat{x})$$

と表わすと, C は a および b から次のように計算される.

まず以下のように逐次的に定義される $2^i \times 2^i$ 行列 $S_i(a)$ ($i=1, 2, \dots, n$) および $S_{i+1}(a)$ ($i=1, 2, \dots, n-1$) を導入する.

$$\begin{aligned} S_1(a) &= \begin{pmatrix} a_0 & a_1 \\ a_1 & a_0 \end{pmatrix}, S_{12}(a) = \begin{pmatrix} a_2 & a_{12} \\ a_{12} & a_2 \end{pmatrix}, \\ S_2(a) &= \begin{pmatrix} S_1(a) & S_{12}(a) \\ S_{12}(a) & S_1(a) \end{pmatrix} \\ &= \begin{pmatrix} a_0 & a_1 & a_2 & a_{12} \\ a_1 & a_0 & a_{12} & a_2 \\ a_2 & a_{12} & a_0 & a_1 \\ a_{12} & a_2 & a_1 & a_0 \end{pmatrix}, S_{23}(a) = \begin{pmatrix} a_3 & a_{13} & a_{23} & a_{123} \\ a_{13} & a_3 & a_{123} & a_{23} \\ a_{23} & a_{123} & a_3 & a_{13} \\ a_{123} & a_{23} & a_{13} & a_3 \end{pmatrix} \\ &\dots\dots\dots \dots\dots\dots \\ S_n(a) &= \begin{pmatrix} S_{n-1}(a) & S_{n-1n}(a) \\ S_{n-1n}(a) & S_{n-1}(a) \end{pmatrix} \end{aligned}$$

上の過程で $S_{i-1}(a)$ は $S_{i-1}(a)$ の各要素の添数の右側に新しい添数 i を加え添数 0 を除いたものを要素

としてもつ行列である. $S_n(a)$ を用いて C は

$$c = S_n(a)b \quad \dots\dots(3)$$

$$= S_n(b)a \quad \dots\dots(4)$$

と表わされる.

(3), (4) 式を記号 $\times$ を用いて次のように書く.

$$c = a \times b \quad \dots\dots(3)'$$

$$= b \times a \quad \dots\dots(4)'$$

スイッチング関数の mod-2 和の性質から次の関係が成立する.

$$1) \quad a \times b = b \times a$$

$$2) \quad a \times (b \times C) = (a \times b) \times C$$

$$3) \quad a \times C = a$$

$$4) \quad a \times a = e$$

ここで e は最初の要素が 1 でほかはすべて 0 である 2^n 次元ベクトルである. すなわち $e = (1, 0, 0, \dots, 0)$. 特に重要なのは 4) の性質で, これは係数ベクトルの各要素の間にある関係が成立していなければならないことを示している. たとえば $n=2$ の場合 4) は

$$\begin{cases} a_0^2 + a_1^2 + a_2^2 + a_{12}^2 = 1 \\ a_0 a_1 + a_2 a_{12} = 0 \\ a_0 a_2 + a_1 a_{12} = 0 \\ a_0 a_{12} + a_1 a_2 = 0 \end{cases}$$

と等価である.

3. C形式による学習的なパターン認識

C形式を用いて n 次元パターン $x = (x_1, x_2, \dots, x_n)$ を二つのクラスに類別する問題を考える. 識別関数として(1)式の形をとる. これは次のような条件を満たすスイッチング関数の係数ベクトルを求める問題に帰着する.

$$y(x) = (a, \hat{x}) = \begin{cases} 1 & x \text{ がクラス 1 に属するとき} \\ -1 & x \text{ がクラス 2 に属するとき} \end{cases}$$

誤差の損失関数 $L(a)$ を

$$L(a) = E_x (y - (a, \hat{x}))^2$$

ととる. E_x は入力パターン x に関する期待値をとることを意味する. 確率近似法 (stochastic approximations)⁴⁾によれば $L(a)$ を最小にする a の値は, 逐次式

$$\begin{aligned} a(t+1) &= a(t) + \gamma(t) \frac{\partial L(a)}{\partial a} \Big|_{a=a(t)} \\ &= a(t) + \gamma(t) (y(t+1) - (a(t), \hat{x}(t+1))) \hat{x}(t+1) \end{aligned}$$

において修正幅 $\gamma(t)$ を適当に選び t を十分大とした極限值である. 今, $\gamma(t)$ として修正の各段階においてそのときの入力パターンに対して正解を与えるように, すなわち $(a(t+1), \hat{x}(t+1)) = y(t+1)$ を満たすような値を選ぶ. 代入してみれば明らかのように, その場合は $\gamma(t) = 2^{-n}$ となる. 結局, 学習の基本式は

$$a(t+1) = a(t) + 2^{-n}cy(t+1) - (a(t), \hat{x}(t+1))\hat{x}(t+1) \quad \dots\dots(5)$$

となる。

(5)式は定理 1 で述べた係数ベクトルの計算法を逐次式の形で表現したものにはかならない。

Fig. 1 Block diagram of learning identifier of switching circuits

(5)式の性質を調べるために必要な補題を次に述べる。

補題 1 $x^i \neq x^j$ ならば $(\hat{x}^i, \hat{x}^j) = 0$ である。

証明は付録 2 を参照されたい。

これを用いると、 $x(t) \neq x(t-1)$ ならば $(a(t+1), \hat{x}(t)) = (a(t), \hat{x}(t))$ となる。この関係を逐次利用すると (5)式は次のような性質をもつことがわかる。(初期値はすべて 0 とする。)

1) l 番目のパターン x^l がはじめて提示された時刻を $t=i$ とすると $(a(i-1), \hat{x}(i)) = 0$

2) $t > i$ で x^l が提示された時刻を $t=j_1, j_2, \dots, j_s, \dots$ とすると $y(j_s) = (a(j_s-1), \hat{x}(j_s))$ 。

性質 1) および 2) は、はじめてのパターンが提示されたときのみ修正が行なわれ、以後そのパターンが提示されたときは必ず正解を示し、しかもその修正はほかのパターンの学習には影響を与えないという、いわゆる学習の直交性を示している。したがって、出現する可能性のあるすべてのパターンが提示されると同時に学習は終了する。これは、閾関数による学習と比して各パターンに対する学習が完全に独立し記憶のかく乱が全くない、いわば「完全な学習」ともいうべきで、 2^n 個もの多くの学習素子を必要とすることの当然の帰結でもある。

(5)式を実際に応用した例については Table 6 を参照。

4. スイッチング関数系と C 形式

4.1 拡大行列と T_f 行列

m 個の n 変数スイッチング関数によるスイッチング関数系を考える。

$$\begin{cases} y_1(x) = (a^1, \hat{x}) \\ y_2(x) = (a^2, \hat{x}) \\ \dots\dots\dots \\ y_m(x) = (a^m, \hat{x}) \end{cases} \quad \dots\dots(6)$$

(6)式は行列 $A^T = (a^1, a^2, \dots, a^m)$ を用いると次のように表現できる。

$$y = \hat{A}\hat{x} \quad \dots\dots(6)'$$

ただし $y^T = (y_1, y_2, \dots, y_m)$ で A は $m \times 2^n$ 行列である。(3)'式または(4)'式を用いると(6)式から

$$\begin{cases} 1 = (e, \hat{x}) \\ y_1 = (a^1, \hat{x}) \\ y_2 = (a^2, \hat{x}) \\ y_1 y_2 = (a^1 \times a^2, \hat{x}) \\ \dots\dots\dots \\ y_1 y_2 \dots y_m = (a^1 \times a^2 \times \dots \times a^m, \hat{x}) \end{cases} \quad \dots\dots(7)$$

となる。 $\hat{A}^T = (e, a^1, a^2, a^1 \times a^2, \dots, a^1 \times a^2 \times \dots \times a^m)$ とおくと(7)式は

$$\hat{y} = \hat{A}\hat{x} \quad \dots\dots(7)'$$

と表わされる。 $\hat{A}$ は $2^m \times 2^n$ 行列で、 A が $\hat{x}$ から y への写像を与えるのに対し $\hat{A}$ は $\hat{x}$ から $\hat{y}$ への写像を与える。 $\hat{A}$ を A の拡大行列とよぶ。

(6)式は 2^n 個の n -tuple $(x_1, x_2, \dots, x_n)$ から 2^m 個の m -tuple $(y_1, y_2, \dots, y_m)$ への写像 f を表現しているが、このことに着目して f をより直接的な方法で表現することができる。

$$\text{変換 } v_i = \frac{1}{2}(1 - x_i) \quad (i=1, 2, \dots, n), w_j = \frac{1}{2}(1 - y_j) \quad (j=1, \dots, m)$$

により変数を $(1, -1)$ から $(0, 1)$ へ戻し、数 $\alpha = 1 + \sum_{i=1}^n v_i 2^{i-1}$ と数 $\beta = 1 + \sum_{j=1}^m w_j 2^{j-1}$ を定義する。すると f は $\alpha = 1, 2, \dots, 2^n$ と $\beta = 1, 2, \dots, 2^m$ との対応関係として表現される。いま、 $2^n \times 2^m$ 行列 T_f を次のように定義する。

$$(T_f)_{\alpha\beta} = \begin{cases} 1, & \alpha=i \text{ が } \beta=j \text{ に対応しているとき} \\ 0, & \text{それ以外のとき} \end{cases}$$

たとえば Table 3 で表わされるスイッチング関数系 f に対して

$T_f = \begin{pmatrix} 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \end{pmatrix}$	x_1	x_2	x_3	y_1	y_2
	1	1	1	1	-1
	-1	1	1	1	1
	1	-1	1	1	-1
	-1	-1	1	-1	-1
	1	1	-1	1	1
	-1	1	-1	-1	1
	1	-1	-1	-1	-1
	-1	-1	-1	1	1

Table 3. An example of switching functions

となる。

こうして定義された T_f と $\hat{A}$ とは次のような双対的な関係でむすばれている。

定理 2 T_f と $\hat{A}$ との間には次のような関係がある.

$$T_f = 2^{-m} H_n \hat{A}^T H_m \quad \dots\dots (8)$$

$$\hat{A} = 2^{-n} H_m T_f^T H_n \quad \dots\dots (9)$$

ただし H_n, H_m は定理 1 で定義された行列である.

(証明) $\hat{A}$ の定義と H_n の構造から $\hat{A} H_n$ の第 i 列は $\alpha=i$ に対応する像 $\hat{y}$ に等しい. 一方 H_m の第 j 行は $\beta=j$ に対応する y であるから $H_m \hat{A} H_n$ の (j, i) 要素は $\alpha=i$ が $\beta=j$ に対応しているとき $(\hat{y}, y) = 2^m$ となり, そうでない場合は補題 1 で示した直交性により 0 となる. すなわち

$$H_m \hat{A} H_n = 2^m T_f^T$$

H_m, H_n の対称性を考慮すれば (8) 式が得られる. また, $H_n H_n^T = 2^n I_{2^n}$ となることを用いれば (8) 式から (9) 式が直ちに得られる. (証明終)

4.2 写像の性質と逆関数系

定理 3 $m \geq n$ の場合写像 (6) が 1 対 1 であるための必要十分条件は

$$\hat{A}^T \hat{A} = 2^{m-n} I_{2^n} \quad \dots\dots (10)$$

となることである. ただし I_{2^n} は $2^n \times 2^n$ 単位行列である.

(証明) 写像 f が 1 対 1 であるためには $T_f T_f^T = I_{2^n}$ となる必要十分であることが容易にわかる. したがって (9) 式から

$$\begin{aligned} \hat{A}^T \hat{A} &= 2^{-2n} H_n^T T_f H_m^T H_m T_f^T H_n \\ &= 2^{-2n+m} H_n^T T_f T_f^T H_n \\ &= 2^{m-n} I_{2^n} \end{aligned}$$

逆に (10) 式が成立すれば (8) 式から

$$\begin{aligned} T_f T_f^T &= 2^{-2m} H_n \hat{A}^T H_m H_m^T \hat{A} H_n \\ &= I_{2^n} \end{aligned} \quad (\text{証明終})$$

系 $m=n$ の場合写像 (6) の逆写像が存在するための必要十分条件は

$$\hat{A}^T \hat{A} = I_{2^n} \quad \dots\dots (11)$$

となることで, その逆写像は

$$\hat{x} = \hat{A}^T \hat{y}$$

で与えられる.

(証明) 逆写像が存在するためには (6) 式が 1 対 1 かつ onto であることが必要十分条件であるが, $m=n$ の場合は 1 対 1 であれば必ず onto となる. したがって (10) 式で $m=n$ とおいた式が求める条件である. 定理の後半は前半から自明である. (証明終)

(例) Table 4 で与えられたスイッチング関数系を考える.

これを C 形式で表現すると,

$$\begin{cases} y_1 = 1/2(-x_1 - x_1 x_2 + x_3 - x_2 x_3) \\ y_2 = 1/2(-x_2 - x_1 x_2 + x_2 x_3 - x_1 x_2 x_3) \\ y_3 = x_1 x_3 \end{cases}$$

x_1	x_2	x_3	y_1	y_2	y_3
1	1	1	-1	-1	1
-1	1	1	1	1	-1
1	-1	1	1	1	1
-1	-1	1	1	-1	-1
1	1	-1	-1	-1	-1
-1	1	-1	1	-1	1
1	-1	-1	-1	1	-1
-1	-1	-1	-1	1	1

Table 4 An example of switching functions

したがって

$$A = \begin{pmatrix} 0 & -1/2 & 0 & -1/2 & 1/2 & 0 & -1/2 & 0 \\ 0 & 0 & -1/2 & -1/2 & 0 & 0 & 1/2 & -1/2 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \end{pmatrix}$$

$\hat{A}$ を計算すると

$$\hat{A} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1/2 & 0 & -1/2 & 1/2 & 0 & -1/2 & 0 \\ 0 & 0 & -1/2 & -1/2 & 0 & 0 & 1/2 & -1/2 \\ 0 & 1/2 & 1/2 & 0 & 1/2 & 0 & 0 & -1/2 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1/2 & 0 & -1/2 & -1/2 & 0 & -1/2 & 0 \\ 0 & 0 & -1/2 & -1/2 & 0 & 0 & -1/2 & -1/2 \\ 0 & 1/2 & -1/2 & 0 & 1/2 & 0 & 0 & 1/2 \end{pmatrix}$$

$\begin{matrix} & \uparrow & \uparrow & \uparrow & & & & \end{matrix}$

これより逆関数系は

$$\begin{cases} x_1 = -1/2(y_1 - y_1 y_2 - y_1 y_3 - y_1 y_2 y_3) \\ x_2 = -1/2(y_1 - y_1 y_2 + y_2 y_3 + y_1 y_2 y_3) \\ x_3 = 1/2(y_1 + y_1 y_2 - y_1 y_3 + y_1 y_2 y_3) \end{cases}$$

で与えられる.

注) 逆関数系は連立論理方程式の解とみなすことができる.

5. オートノーマスな順序回路の周期構造

オートノーマスな順序回路を C 形式で表現すれば

$$x(t+1) = \hat{A}x(t) \quad \dots\dots (12)$$

となる. $x(t+1), x(t)$ はおのおの時刻 $t+1, t$ における回路状態をあらわす n 次元ベクトルで A は $n \times 2^n$ 行列である. $x(t+1)$ を拡大すれば

$$\hat{x}(t+1) = \hat{A}\hat{x}(t) \quad \dots\dots (13)$$

となり見かけ上は $\hat{x}(t)$ と $\hat{x}(t+1)$ との間の線形変換の形をとる. (12) 式の最大周期は 2^n で, 一般にはいくつかの孤立した周期列をもつが, この周期構造を $[m_1(l_1), m_2(l_2), \dots, m_k(l_k); m_0]$ とあらわす. $m_i(l_i)$ は長さ l_i の周期を, m_i 個もつことを意味し m_0 は過渡的な状態 (一度その状態を離れると再びそこにもどることのないような状態) の数である. 明らかに $\sum_i m_i l_i + m_0 = 2^n$ である.

(12) 式の表現とその周期構造の間には次の定理で

示されるような直接的な関係が存在する。

定理 4 (12) 式が $[m_1(l_1), m_2(l_2), \dots, m_k(l_k); m_0]$ なる周期構造をもつためには, $\hat{A}$ の固有多項式 $\varphi_{\hat{A}}(\lambda)$ が $\varphi_{\hat{A}}(\lambda) = (\lambda^{l_1} - 1)^{m_1} (\lambda^{l_2} - 1)^{m_2} \dots (\lambda^{l_k} - 1)^{m_k} \lambda^{m_0} \dots$ (14) のように因数分解されることが必要十分条件である。

(証明) (12) 式に対し (8) 式で定義した $T_f = 2^{-n} H_n \hat{A} H_n$ を考える. 適当に内部状態の符号化を変えると孤立した各周期列は cyclic な構造で表現できる. たとえば長さ l_i の周期列は Table 6 のような遷移表で与えられる.

t	S_i	S_{i+1}	S_{i+2}	$\dots$	S_{i+l_i}
$t+1$	S_{i+1}	S_{i+2}	S_{i+3}	$\dots$	S_i

Table 5 A cyclic structure of state transitions

これに対応する T_f の部分行列は $2^{l_i} \times 2^{l_i}$ 行列

$$L_i = \begin{pmatrix} 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 1 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots \\ 1 & 0 & 0 & \dots & 0 \end{pmatrix}$$

であらわれ, 固有方程式は $\varphi_{L_i}(\lambda) = \lambda^{l_i} - 1$ となる. これを考慮すると

$$T_f = S^T \begin{pmatrix} A_1 & 0 & 0 \\ & A_2 & \vdots \\ 0 & \dots & A_k \\ * & & 0 \end{pmatrix} S, \quad A_i = \begin{pmatrix} L_i & 0 \\ & L_i \\ 0 & \dots & L_i \end{pmatrix}$$

となる. S は 1 対 1 の符号変換をあらわす行列で定理 3 の系から直交行列である. また, * の部分は過渡状態を表わす部分で各列はただ一つの要素が 1 でほかはすべて 0 である.

これらを用いて (9) 式から実際に $\hat{A}$ の固有方程式を計算すると (14) 式が得られる.

6. 順序回路の入出力関係の学習的同定

ある未知の順序回路に対し, その実際の入出力系列がデータとして与えられたとき, その中にあらわれる法則性を抽象することによってその順序回路の機能を再構成することはいかにして可能であろうか? これは, ブラックボックスとして与えられた順序回路のモデル構成あるいは同定 (identification) の問題である (Fig. 2). この問題に対して今まで考察してきた C 形式を応用する.

順序回路が上に述べた意味で同定可能であるためには一意的な入出力関係をもたねばならない. ところが A. Gill⁵⁾ が反例をあげたように, すべての順序回路は一意的な入出力関係をもつとは限らない. この場合,

Fig. 2 Identification of sequential circuit

回路の初期状態が無限の未来に対して入出力関係に影響を与える. そこで, 一意的な入出力関係をもつ順序回路のクラスを A. Gill⁵⁾ に従って有限記憶回路 (finite memory sequential circuit) と名付け次のように定義する.

定義 時刻 t における出力 $y(t)$ が, 過去の μ 個の入力 $u(t-1), u(t-2), \dots, u(t-\mu)$ と過去の μ 個の出力 $y(t-1), \dots, y(t-\mu)$ から一意的に決定できる場合, すなわち

$$y(t) = f(y(t-1), y(t-2), \dots, y(t-\mu), u(t-1), \dots, u(t-\mu)) \quad \dots (15)$$

なる関数関係が存在するとき, この回路は有限記憶回路であるといい, μ の最小値をこの回路の記憶とよぶ.

一般の順序回路は C 形式を用いると

$$\begin{cases} x(t+1) = A(\hat{x}(t) \times \hat{u}(t)) \\ y(t) = B(\hat{x}(t) \times \hat{u}(t)) \end{cases} \quad \dots (16)$$

と表わされる. ここで $u(t), y(t)$ は時刻 t における入力出力を表わす r, m 次元ベクトル, A, B はおのおの $n \times 2^{n+r}, m \times 2^{n+r}$ 行列である.

ここでは簡単のため出力が状態のみに依存する [入力]出力回路

$$\begin{cases} x(t+1) = A(\hat{x}(t) \times \hat{u}(t)) \\ y(t) = (b, \hat{x}(t)) \end{cases} \quad \dots (17)$$

を考える. 以下の解析は, 一般の (16) 式の場合にも容易に拡張できる.

C 形式を用いれば (17) 式が有限記憶であるための必要十分条件をもとめることができるが, 非常に複雑になるので省略する.

今, 与えられた未知の有限記憶回路の入出力関係が

$$y(t) = (a, \hat{w}[t-1; \mu]) \quad \dots (18)$$

と表現されるものとする. ここで $\hat{w}[t-1; \mu] = (y(t-1), y(t-2), \dots, y(t-\mu), u(t-1), \dots, u(t-\mu))$ とする. (5) 式により, 未知の係数ベクトル a は次のような逐次式で決定することができる.

$$a(t+1) = a(t) + 2^{-2\mu} (y(t+1) - a(t), \hat{w}[t; \mu]) \hat{w}[t; \mu] \quad \dots (19)$$

(19) 式は (5) 式に対して 3 で述べたと全く同様の性

Fig 3 Block diagram of learning identifier of sequential circuit

質をもつ。すなわち、ある入出力関係に対する学習が一度行なわれれば、その後その入出力関係に対しては必ず正しい解答を与える。

(例) 次のような回路(20)式をモデルにして実際に同定を行なった結果を Table 8 に示す。

$$\begin{cases} x_1(t+1) = 1/2(-x_1(t) - x_2(t) \\ \quad - x_1(t)u(t) + x_2(t)u(t)) \\ x_2(t+1) = x_1(t)x_2(t)u(t) \\ y(t) = x_2(t) \end{cases} \quad \dots\dots (20)$$

入力は1と-1をランダムに入れた。表の各係数は実際の値を $2^4=16$ 倍したものである。時刻を示す欄で欠けているのはすでに前に一度出現したパターンが再び現われた場合で、学習は行なわれず係数は不変であるから除外した。なお初期値 $a(0)$ はすべて0とした。

この結果から(20)式の入出力関係は

$$y(t) = 1/2[u(t-1)y(t-2)(1+u(t-2)) + u(t-1)y(t-2)y(t-1)(1-u(t-2))]$$

と表わされることがわかる。

この例は有限記憶回路であるが、有限記憶でない場合もある入出力パターンに対しては誤りを許容すればやはり同様に同定を行なうことができる。

7. 結 言

スイッチング関数に対して、C形式という表現法を統一的に考察した。この表現法は、外見上スイッチング関数を線形空間における写像として取り扱うことができる。このことから、逆写像や順序回路の周期構造、などの性質を行列とベクトルを用いて表現できた。また、この表現法がパラメータを変えることによってすべてのスイッチング関数を表現できるので、スイッチング関数を「学習的」に決定するのに好都合であり、しかもその学習法則は直交性という便利な性質をもっている。欠点は多くの素子(すなわち入力の数と同じ 2^n 個)を必要とすることで、この点で threshold logic や高次の(2個以上の変数の積を項としてもつ) threshold logic との関連は興味ある問題である。また、この形式を用いて順序回路のほかのいろいろな問題(たとえば可制御性、情報無損失回路や符号化回路、状態割当問題など)を解析することが可能であろう。

おわりに、日ごろからご指導、ご激励をいただいている南雲仁一先生に深謝いたします。

参 考 文 献

- 1) I Ninomiya: "A Study of the Structure of Boolean Functions and its Applications to the Synthesis of Switching Circuits", Memoirs of the Fac. of Eng., Univ. of Nagoya, 13-2, pp.149-363 (1961)
- 2) R.P.Coleman: "Orthogonal Functions for Logical Design of Switching Circuits", IRE Trans. on Electronic Computers, 10-EC-10, pp. 379-383, September (1961)
- 3) Earl E. Goss: "An Adaptive Network for Producing Real Functions of Binary Inputs", Information and Control, 8-2, pp.111-123 (1965)
- 4) Я.З.Цыпкин: "Адаптация, Обучение и Самообучение в Автоматических Системах", Автоматика и Телемеханка, 27-1, pp. 23~60 (1966)

t	u _{t-2}	u _{t-1}	y _{t-2}	y _{t-1}	error	a ₀	a ₁	a ₂	a ₁₂	a ₃	a ₁₃	a ₂₃	a ₁₂₃	a ₄	a ₁₄	a ₂₄	a ₁₂₄	a ₃₄	a ₁₃₄	a ₂₃₄	a ₁₂₃₄
1	-1	1	1	-1	1	1	-1	1	-1	1	-1	1	-1	-1	1	-1	1	-1	1	-1	1
2	1	1	-1	1	1	2	0	2	0	0	-2	0	-2	0	2	0	2	-2	0	-2	0
3	1	-1	1	1	-1	3	1	1	-1	1	-1	-1	-3	1	3	-1	1	-1	1	-3	-1
4	-1	1	1	1	-1	2	2	0	0	0	0	-2	-2	0	4	-2	2	-2	2	-4	0
5	1	-1	1	-1	1	3	3	-1	-1	1	1	-3	-3	-1	3	-1	3	-3	2	-3	1
6	-1	-1	-1	1	-1	2	4	0	-2	2	0	-4	-2	-2	4	0	2	-2	0	-4	2
7	1	1	1	1	-1	1	3	-1	-3	1	-1	-5	-3	-3	3	-1	1	-3	-1	-5	1
13	1	-1	-1	1	-1	0	2	0	-2	2	0	-6	-4	-4	2	0	2	-2	0	-6	0
19	1	1	1	-1	-1	-1	1	-1	-3	1	-1	-7	-5	-3	3	1	3	-1	1	-5	1
20	1	1	1	-1	1	0	2	0	-2	0	-2	-8	-6	-4	2	0	2	0	2	-4	2
24	-1	-1	1	-1	-1	-1	3	1	-3	-1	-1	-7	-7	-3	1	-1	3	1	1	-5	3
25	-1	1	-1	-1	-1	-2	4	0	-2	0	-2	-6	-8	-2	0	0	2	0	2	-6	4
32	1	-1	-1	-1	-1	-3	3	1	-1	1	-1	-7	-9	-1	1	-1	1	-1	1	-5	5
33	-1	-1	-1	-1	1	-2	2	0	0	0	0	-6	-10	-2	2	0	0	0	0	-6	6
34	-1	1	-1	1	1	-1	1	1	-1	-1	1	-7	-9	-1	1	1	-1	-1	1	-7	7
160	-1	-1	1	1	1	0	0	0	0	0	0	-8	-8	0	0	0	0	0	0	-8	8

Table 6 Identification process of the circuit (26)

- 5) A. Gill: Introduction to the Theory of Finite-State Machine, New York, McGraw-Hill (1962)
 6) R. E. Kalman: "Mathematical Description of Linear Dynamical Systems", J. of SIAM Control, ser, A, 1, pp. 152~192 (1963)

付 録 1

(定理 1 の証明) $n=1$ のとき $y=a_0+a_1x_1$

これより $y_1=y(x_1=1)=a_0+a_1$

$$y_1=y(x_1=-1)=a_0-a_1$$

したがって

$$\begin{pmatrix} a_0 \\ a_1 \end{pmatrix} = 2^{-1} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \begin{pmatrix} y_1 \\ y_{-1} \end{pmatrix} \\ = 2^{-1} H_1 t(f)$$

$n=N-1$ のとき定理は成立しているとする. このとき, $n=N$ に対して $x_N=1$ のときの $(x_1, \dots, x_{N-1}) = x^{(N-1)}$ がつくる 2^{N-1} 次元の真理値ベクトルを $\bar{t}(f)$, $x_N=-1$ のときのそれを $\bar{\bar{t}}(f)$ とする. また係数ベクトル a を二つの 2^{N-1} 次元ベクトルに分割し $a^T = (\bar{a}^T, \bar{\bar{a}}^T)$ とすると帰納法の仮定と $\bar{t}(f), \bar{\bar{t}}(f)$ の定義から

$$y = \bar{a}^T \hat{x}^{(N-1)} + \bar{\bar{a}}^T \hat{x}^{(N-1)} x_N = x^{(N-1)} = (x_1, x_2, \dots, x_{N-1})$$

であるから

$$\bar{a} + \bar{\bar{a}} = 2^{-(N-1)} H_{N-1} \bar{t}(f)$$

$$\bar{a} - \bar{\bar{a}} = 2^{-(N-1)} H_{N-1} \bar{\bar{t}}(f)$$

これをとくと

$$a = \begin{pmatrix} \bar{a} \\ \bar{\bar{a}} \end{pmatrix} = 2^{-N} \begin{pmatrix} H_{N-1} & H_{N-1} \\ H_{N-1} & -H_{N-1} \end{pmatrix} \begin{pmatrix} \bar{t}(f) \\ \bar{\bar{t}}(f) \end{pmatrix} \\ = 2^{-N} H_N t(f)$$

よって証明された.

付 録 2

(補題 1 の証明)

$$n=1 \text{ のとき } (\hat{x}^i, \hat{x}^j) = 1 + x_1^i x_1^j$$

仮定より $x_1^j \neq x_1^j$ であるから $x_1^i x_1^j = -1$ よって $n=1$ のときは確かに成立する.

$n=N-1$ のとき成立するとする. $\hat{x}$ の定義から $x^{(N-1)} = (x_1, x_2, \dots, x_{N-1})$ とすれば

$$\hat{x} = \begin{pmatrix} \hat{x}^{N-1} \\ x_N \hat{x}^{N-1} \end{pmatrix}$$

であるから

$$\begin{aligned} (\hat{x}^i, \hat{x}^j) &= (\hat{x}^{i(N-1)}, \hat{x}^{j(N-1)}) + (x_N^i \hat{x}^{i(N-1)}, x_N^j \hat{x}^{j(N-1)}) \\ &= (1 + x_N^i x_N^j) (\hat{x}^{i(N-1)}, \hat{x}^{j(N-1)}) \end{aligned}$$

$x^{i(N-1)} \neq x^{j(N-1)}$ のときは帰納法の仮定より $(\hat{x}^{i(N-1)}, \hat{x}^{j(N-1)}) = 0$, $x^{i(N-1)} = x^{j(N-1)}$ のときは $x^i \neq x^j$ の仮定から $x_N^i \neq x_N^j$ でなければならない. したがって $x_N^i x_N^j = -1$.

いずれの場合にも

$$(\hat{x}^i, \hat{x}^j) = 0$$

となる. よって証明された.