論理関数を実現するのに必要な論理回路の段数について

ON THE DEPTH OF COMBINATIONAL CIRCUITS REQUIRED TO COMPUTE SWITCHING FUNCTIONS

安浦寛人・矢島脩三
HIROTO YASUURA AND SHUZO YAJIMA
（京都大学 工学部）
(FACULTY OF ENGINEERING, KYOTO UNIVERSITY)

1. まえがき

論理関数を論理回路で計算する場合，与えられた数種類の基本的な論理演算を実現する論理素子を組み合せて，適当な組合せ論理回路を構成して論理回路を実現する．この際に必要となる論理素子の数と計算時間に関する議論は，回路の複雑さや性能を評価する規準を与える．特に，大規模な回路においては，回路構成の違いによって，その複雑さや性能に大きな差が生じる．このため，論理関数を実現するときに必要となる論理素子の数や計算時間に関する議論は，単に理論的興味からだけでなく実用的見地からも重要である．与えられた数種の論理素子を組み合せて論理関数を実現する際に必要となる素子の数（回路の規模）の評価は，Shannon，Lupanov等の研究を始めとして古くから多くの研究がある〔SAVA76〕．一方，回路の計算時間は，回路の段数によって規定されると考えられるが，これに対しては，個々の算術演算回路等についての議論はなされているものの，一般の論理関数について統一的に議論したものは少ない〔SPIR6905〕．ここでは，n変数論理関数を与えられた素子集合を使って実現するときに必要となる段数が，入力変数の数nにどのように関係するかを考える．特に，よく知られた関数族や，実用上重要と思われる関数集合について詳しく議論する．

組合せ論理回路の計算時間は，回路の入力から出力へ至る各経路の最大遅延時間であり，各素子の遅延が同じであると仮定すると，回路の段数に比例すると考えられる．各種算術演算や処理を行うのに必要な回路の段数については，いくつかの理論的な研究がある〔SPIR7306〕．Ofman と Karatsuba は，mビットの2数の加算および乗算が，$\log_2 m$に比例する段数で計算できることを示した〔OFMA62〕，〔KARA062〕．さらに，Winograd は，加算，乗算に必要な計算時間の詳細な検討を行っている〔WINO6504〕〔WIN6710〕．Muller と Preparata は，n個の要素の Sorting を，$\log_2 n$に比例した段数で行う回路を与え，さらにブール表現中のリテラル数とその表現を実現する回路の段数の関係を与えた〔MULLP7504〕〔PREPM7705〕．一方，Unger は，順序回路により実現される関数を，組合せ回路で高速に計算する回路構成法を述べている〔UNGE7704〕．Spira は，あらゆるn変数論理関数を実現するためには，nに比例した段数の回路が必要となることを示した〔SPIR7101〕．しかし，加算，乗算，パリティ関数をはじめとして，多くの実用的な関数は，$O(\log n)$段の回路で実現できる．どのような関数が，$O(\log n)$段の回路で実現できるかという問題は，高速の組合せ回路を構成する立場から興味ある問題である．本稿では，この問題を中心に議論する．

このような議論をする際に，変数の数と無関係に，ある性質を共有するような論理関数の集合を考えることが必要となる．閾値関数族や対称関数族のような関数族はその好例である．しかし，実用的な面を考えると，n変数パリティ関数，n入力AND，m bit 加算器といった，もっと強い性質を共有する論理関数の集合が取り扱えることが望ましい．このような集合を取

り扱うために，ここでは，各nに対してちょうど1個のn変数関数をふくむような集合，「関数列」を導入した．さらに，この関数列が$\{0,1\}^*$上の形式言語と1対1に対応することに注目し，言語理論的な記述で関数列を指定する方法を導入した．

以下，2節で論理関数の計算時間を定義し，3節で$O(\log n)$段で実現できる論理関数集合の一般的な性質を調べる．4節では，いろいろな関数族中の関数の計算時間について述べる。最後に，新しく導入した関数列と計算時間の関係について5節で述べる．

2．論理関数の計算時間

論理関数の有限集合$\Omega=\{f_i \mid f_i:\{0,1\}^{r_i}\to\{0,1\}$, r_iは正整数$\}$を基底(basis)と呼ぶ．基底Ω上の組合せ回路Cとは，それぞれがΩに属する関数を実現する論理素子によって，次のように構成されるものである．

1．回路Cは，外部入力と1つの外部出力を持つ．外部入力は，論理関数の入力変数か，もしくは定数（0または1）に対応する。

2．各素子の入力は，外部入力か，もしくは回路中の他の素子の出力である。

3．外部出力は，1つの外部入力か，もしくは回路中の1つの素子の出力である．

4．任意の外部入力から外部出力へ至る経路の中で，同一素子を2回以上含むものは無い．ここに，経路とは，素子の列$e_{i_1}, e_{i_2}, \cdots, e_{i_n}$(1)で，$e_{i_1}$の入力の1つ以上が入力変数に対応する外部入力で，$e_{i_k}$の出力が$e_{i_{k+1}}$の入力の1つになっており（$k=1,2,\cdots,n-1$）さらに，$e_{i_n}$の出力が外部出力となっているものである。

すなわち，ここで考える組合せ回路は，フィードバックループを含まない1出力の回路である．

組合せ回路の段数(depth)とは，外部入力から外部出力に至るすべての経路のうち，最も多くの素子を含む経路の素子数である．論理関数fの基底Ω上での計算時間$D_\Omega(f)$とは，fを実現する基底Ω上の組合せ回路のうちで，最も段数の小さな回路の段数である．

(1) ここで $i_j = i_l$ $(j \neq l)$も許される。

任意の論理関数をΩ上の組合せ回路で実現できるとき，基底Ωは完全(complete)であるという．すべてのr変数関数の集合からなる基底$\{f \mid f:\{0,1\}^r\to\{0,1\}\}$を$\Omega_r$で表わす．$r \geqq 2$のとき，基底$\Omega_r$は完全である．

論理関数fの計算時間$D_\Omega(f)$は，基底Ωの選び方によって異る．しかし，完全基底の上では，その違いは高々定数倍程度であることが次の定理からわかる．

〔定理1〕　ΩとΩ'をともに完全基底とする．任意の論理関数fに対して，

$$D_{\Omega'}(f) \leqq D_0\, D_\Omega(f)$$

が成立する．但し，$D_0=\max_{g\in\Omega} D_{\Omega'}(g)$．

（証明）　fはΩ上で$D_\Omega(f)$段の回路Cで実現できる．今，Ω中の各関数gを，Ω'上で$D_{\Omega'}(g)$段の回路で構成し，C中の各素子を対応するΩ'上の回路で置き換えた回路C'を考える．C'の段数は高々$(\max_{g\in\Omega} D_{\Omega'}(g))D_\Omega(f)$である．□

3．$O(\log n)$段実現可能な関数集合

真にn変数に依存する関数の計算時間は，Ω_r上で$\lceil \log_r n \rceil$(2)以上となることは，木状に回路を構成することを考えれば容易にわかる．一方，F_n $(n=1,2,\cdots)$ですべてのn変数関数からなる集合を表わすと，任意のkに対し，ある定数N_kがあって，$n>N_k$なるすべてのnに対して，

$$\lceil n\log_r 2\rceil-\log_r\log_r n \leqq \max_{f\in F_n} D_{\Omega_r}(f) \leqq \lceil n\log_r 2\rceil + 2\underbrace{\log_r\cdots\log_r}_{k回} n$$

となることをSpiraが示した[SPIR7101]．すなわち，n変数関数の計算時間$D_{\Omega_r}(f)$の最大値は，nに比例する程度の増え方をする．しかし，論理関数の部分集合を考えた時，その中のすべてのn変数関数の計算時間の最大値が，Ω_r上で$\log_r n$に比例するような集合を考えることができる．

〔定義〕　ある論理関数の集合Aが$O(\log n)$段実現可能とは，ある正の定数αとCが存在して，任意のA中のn変数関数fについて，

$$D_{\Omega_r}(f) \leqq \alpha \log_r n + C$$

がすべての自然数nに対して成立することであ

(2) $\lceil x\rceil$は，xに等しいかより大きな最小の整数を表わす．

る．

Preparata と Muller は，k個のリテラルを持つブール表現で表わすことのできる論理関数の計算時間は，完全基底$\Omega=\{xy, x+y, \bar{x}\}$上で，高々 $2.81\lceil \log_2 k \rceil$ であることを示した〔PREPM 7705〕これより，論理関数の集合が $O(\log n)$ 段実現可能であるための必要十分条件が得られる．

〔定理２〕 論理関数の集合 A が $O(\log n)$ 段実現可能であるための必要十分条件は，ある多項式 $P(x)$ が存在し，A 中のすべての n 変数関数が $P(n)$ 以下のリテラル数を持つブール表現で表わされることである．

（証明） 十分性．A 中の n 変数論理関数は，$P(n)$以下のリテラルを持つブール表現で表わせるから，Preparata 等の結果より，$2.81\lceil \log_2 P(n) \rceil$段以下の$\Omega=\{xy, x+y, \bar{x}\}$上の回路で実現できる．$P(n)$ は多項式であり，Ωは完全基底であるから，定理１より A は $O(\log n)$段実現可能である．

必要性．A 中の各 n 変数論理関数 f を実現するΩ上の $\alpha \log_2 n + C$ 段以下の論理回路 C_f があるとする．ここに，α と C は正の定数である．回路 C_f に対し，同じ関数を実現し，しかも段数も等しく，回路中のすべての素子の出力が，ただ１つの素子の入力となっているような回路 C'_f が存在する．（すなわち C'_f は Fan-out 1 の回路である.）この C'_f 中の各素子は，f のブール表現の演算子（・，＋，¯）に１対１に対応し，その個数は高々 $n^{\alpha}+2^{C}-1$ である．すなわち，ブール表現は高々 $n^{\alpha}+2^{C}$個のリテラルしか含まない．□

次に，$O(\log n)$ 段実現可能な関数集合についての性質を述べる．

〔性質１〕 A が $O(\log n)$ 段実現可能な関数集合とすると

$$B=\{g \mid {}^{\exists}f \text{ s.t. } g \overset{NPN}{\sim} f,\ f\in A\}$$

なる集合 B も $O(\log n)$ 段実現可能である．ここに，$g \overset{NPN}{\sim} f$ とは，g が f に次のような操作を０回以上施して得られることを表わす．1)関数の否定，2)変数の置換，3)変数の否定．

（証明） NPN 変換はブール表現のリテラル数を変化させないから，定理２より明らか．□

〔性質２〕 ある多項式 $P(x)$ があって，関数集合 A において，任意の $f\in A\cap F_n$ なる関数 f について

$$|\{f^{-1}(1)\}|<P(n) \quad \text{または} \quad |\{f^{-1}(0)\}|<P(n)$$

が成立するとき，A は $O(\log n)$ 段実現可能である．

（証明） A 中の各関数の標準加法形または標準乗法形によるブール表現は、高々 $nP(n)$個のリテラルを持つから，定理２より明らか． □

〔性質３〕 A を $O(\log n)$段実現可能な関数集合とする．$P(x)$ を多項式とする．すべての自然数 n について $f\in A\cap F_n$ なる f に対し，

$$|\{f^{-1}(1)\}\ominus\{g^{-1}(1)\}|<P(n) \quad {}^{(1)}$$

なる関係をもつ関数 g からなる関数集合は，$O(\log n)$ 段実現可能である．

（証明） $h_1=\bar{f}g$，$h_2=f\bar{g}$ とすると，$g=h_1+\bar{h}_2 f$ と表わせる．

$$\{h_1^{-1}(1)\}\cup\{h_2^{-1}(1)\}=\{f^{-1}(1)\}\ominus\{g^{-1}(1)\}$$

であるから，

$$|\{h_1^{-1}(1)\}|<P(n),$$
$$|\{h_2^{-1}(1)\}|<P(n),$$

となる．これより，$g=h_1+\bar{h}_2 f$ を単純に実現し，

$$D_{\Omega_r}(g)\leq \max(D_{\Omega_r}(f), \lceil \log_r n\rceil+\lceil \log_r P(n)\rceil)+2$$

であることがわかる． □

性質２は，集合中の関数の on set（off set）の数が多項式 $P(n)$で押えられる場合，集合は $O(\log n)$ 段実現可能であることを示している．性質３は，$O(\log n)$ 段実現可能な関数集合 A に，A 中の関数と高々 $P(n)$ 個の入力に対してのみ出力値が異なるような関数 g を加えても，$O(\log n)$ 実現可能であることは変わらないことを主張している．

最後に，関数集合の要素数と $O(\log n)$ 段実現可能性の関係について述べる．

〔性質４〕 関数集合 A が $O(\log n)$ 段実現可能であるための必要条件は，任意の自然数 m に対して，正の定数 k と D があって，

$$|F_n\cap A|<(mn)^{Dn^k}/m$$

が成立することである．

(1) $A\ominus B$ は集合 A と集合 B の対称差を表わす．

(証明)　今，基底$\Omega = \{f_i \mid f_i : \{0,1\}^{r_i} \to \{0,1\},$ $i = 1, 2, \cdots, m\}$上で考える。ここに，$m = |\Omega|$である．$r = \max r_i$とすると，Ω上で$k\lceil \log_r n \rceil + c$段で構成できる関数は高々$(n\,m)^{n^k r^c}/m$個である．　□

4．論理関数族の計算時間

ここでは，いろいろな論理関数族の実現に必要な計算時間を調べる．

4.1　線形関数族

$f(x_1, x_2, \cdots, x_n) = a_0 \oplus a_1x_1 \oplus a_2x_2 \oplus \cdots \oplus a_nx_n$ ($a_i = 0$または1，$i = 0, 1, 2, \cdots, n$．$\oplus$は排他的論理和．nは自然数．)と表わされる論理関数fを線形関数とよぶ．すべての線形関数からなる集合を線形関数族とよぶ．

〔定理3〕　線形関数族は$O(\log n)$段実現可能である。

(証明)　n変数線形関数は、基底$\Omega = \{x \oplus y\}$上で$\lceil \log_2 n \rceil$段以下で構成できる〔MCCO 76〕。□

4.2　対称関数族

関数$f(x_1, x_2, \cdots, x_n)$が，変数$(x_1, x_2, \cdots, x_n)$の置換に関して不変であるとき，fを対称関数という．すべての対称関数からなる集合を対称関数族という．対称関数は，ある整数の組$a_1, a_2, \cdots, a_k$があって，入力変数のうちその値が1であるものの数がa_iに等しいとき1，等しくないとき0を出力する関数と考えられるから，このような対称関数を$S^n_{a_1a_2\cdots a_k}$と記す．

〔補題〕〔MULLP 7504〕　入力$(x_1, x_2, \cdots, x_n)$に対し，この中の値が1に等しいものの数の2進表示$(y_1\ \ y_2\ \ \cdots\ \ y_{\lceil \log_2 n \rceil})$を，基底$\Omega = \{xy + yz + zx,\ x \oplus y \oplus z\}$上で，$2\lceil \log_2 n \rceil$段で構成できる．

この補題を用いて，対称関数族が$O(\log n)$段実現可能であることが示される．基底として，$\Omega = \{xy + yz + zx,\ x \oplus y \oplus z,\ x + y,\ xy\}$を考える．任意の$n$変数対称関数$S^n_{a_1a_2\cdots a_k}$に対して，図1のような回路を構成する．1の数

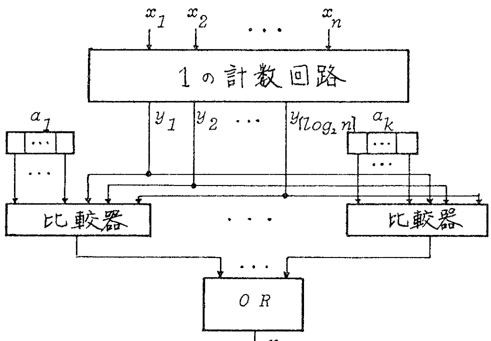

図1．対称関数の実現

の2進表現は，補題より$2\lceil \log_2 n \rceil$段で計算できる．この結果を，$a_1, a_2, \cdots, a_k$の2進表示と比較するのは，$1 + \lceil \log_2 \lceil \log_2 n \rceil \rceil$段のANDでできる．最後に，この$k$個の比較の結果のORをとるのに$\lceil \log_2 k \rceil$段必要であるから，全体で

$$2\lceil \log_2 n \rceil + \lceil \log_2 \lceil \log_2 n \rceil \rceil + \lceil \log_2 k \rceil + 1$$

段で構成できる．

〔定理4〕　対称関数族は$O(\log n)$段実現可能である。

4.3　閾値関数族

$n + 1$個の実数$w_1, w_2, \cdots, w_n, t$に対して，

$$f(x_1, x_2, \cdots, x_n) = \begin{cases} 0 & \sum_{i=1}^{n} w_i x_i < t \text{ のとき} \\ 1 & \sum_{i=1}^{n} w_i x_i \geqq t \text{ のとき} \end{cases}$$

(ここに乗算，加算は実数として行なう.)の形で与えられる関数fを閾値関数という．w_iを重み，tを閾値とよぶ．すべての閾値関数からなる集合を閾値関数族という．

〔定理5〕　閾値関数族は$O(\log n)$段実現可能である．

(証明)　任意の閾値関数は，NPN等価な正関数の閾値関数を持つ．任意のn変数の正関数で閾値関数であるものは，次式を満す整数重みw_iで表わすことができる〔MUROI 7606〕．

$$0 \leqq w_i \leqq 2\left(\frac{n+1}{4}\right)^{(n+1)/2} \qquad (i = 1, 2, \cdots, n)$$

よって重みw_iは，高々$\left(\frac{n+1}{2}\right)\log_2(n+1) - n$ビットで2進数として表現できる．$n$個の$m$ビットの数の加算は，$\log_2 mn$に比例した段数

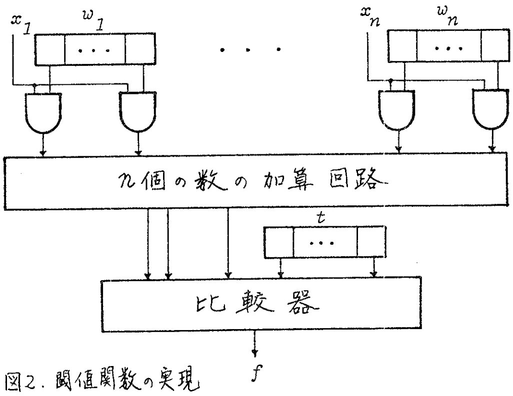

図2. 閾値関数の実現

で行えることが知られている〔OFMA 62〕. その結果を閾値 t と比較するのは, $\log_2(\frac{n+1}{2}(n+1))$ に比例した段数で行える. すなわち, 図2に示すような回路構成により, 正関数の閾値関数は $O(\log n)$ 段実現可能であることがわかる. 性質1より, 閾値関数族は $O(\log n)$ 段実現可能である. □

4.4 $O(\log n)$ 段実現可能でない関数族

〔定理6〕 単調増大(減少)関数族は $O(\log n)$ 段実現可能ではない〔SPIR6905〕.

これは、n 変数の単調関数の数が性質4を満足しないことから示される.

〔系〕 ユネイト関数族は $O(\log n)$ 段実現可能ではない〔SPIR 6905〕.

関数 $f(x_1, x_2, \cdots, x_n)$ が,

$$f(x_1, x_2, \cdots, x_n) = \bar{f}(\bar{x}_1, \bar{x}_2, \cdots, \bar{x}_n)$$

をみたすとき, f を自己双対関数という. すべての自己双対関数からなる集合を自己双対関数族という.

〔定理7〕 自己双対関数族は $O(\log n)$ 段実現可能ではない.

(証明) n 変数自己双対関数は $2^{2^{n-1}}$ 個あり, 性質4の $O(\log n)$ 段実現可能であるための必要条件を満たさない.

5. $O(\log n)$ 段実現可能な関数列

5.1 関数列と形式言語〔YASUY7807〕〔YASUY7810〕

ANDやOR, パリティ関数, 加算, 乗算等, 多くの有用な論理関数は, 任意の入力数に対して定義されるが, それらはある規則に従っており, 規則的な回路構成法も知られている. このような入力変数の数に依存しないような一つの性質を共有する関数の集合を, より一般的に取扱うために, ここでは特に各 n に対しちょうど1個の n 変数関数を含む関数列を考える.

関数列 $\{f_n\}$ を $\{f_n \mid f_n$ は n 変数関数, $n=1, 2, \cdots\}$ と定義する. 一般には, ある m に対して m 変数関数を含まないような関数列も考えられるが, ここではこのような m に対しては,

$$f_m(x_1, x_2, \cdots, x_m) = 0$$

なる定数関数が列中に入っていると考える.

関数列 $\{f_n\}$ に対して, 形式言語 $L_{\{f_n\}}$ を次のように定義する.

$$L_{\{f_n\}} = \{ f_n^{-1}(1) \mid n = 1, 2, \cdots \}$$

ここに, $f_n^{-1}(1)$ は, $f_n(x_1, x_2, \cdots, x_n) = 1$ なる入力ベクトル $(x_1, x_2, \cdots, x_n)$ を $\{0,1\}^*$ 上の系列 $x_1 x_2 \cdots x_n$ と見なしたものであるとする.

〔例〕 $\{f_n\} = \{x_1 \oplus x_2 \oplus \cdots \oplus x_n\}$ とすると, $L_{\{f_n\}}$ は, 正規表現 $0^*1(0^*10^*1)^*0^*$ で表わされる正規集合である.

逆に, $\{0,1\}^*$ 上の言語 L に対して,

$$f_n(x_1, x_2, \cdots, x_n) = \begin{cases} 0 & x_1 x_2 \cdots x_n \notin L \text{ のとき} \\ 1 & x_1 x_2 \cdots x_n \in L \text{ のとき} \end{cases}$$

によって関数列 $\{f_n\}$ を定義できる. よって, 関数列 $\{f_n\}$ と $\{0,1\}^*$ 上の形式言語は1対1に対応する.

5.2 $O(\log n)$ 段実現可能な関数列

$\{f_n\}$ が, 線形関数族, 対称関数族, 閾値関数族のいずれかの部分集合である時は, $\{f_n\}$ は $O(\log n)$ 段実現可能であることは明らかである. さらに, 正規集合に対応する関数列が $O(\log n)$ 段実現可能であることを示すことができる.

Ungerは, 順序回路を一次元展開して得られる組合せ回路を木構造にすることによって, 回路の段数を入力数の対数に比例するようにでき

ることを示した〔UNGE7704〕。この手法を使うと次の定理が導かれる。

〔定理8〕 Lを$\{0,1\}^*$上の正規集合とする。

$$f_n(x_1,x_2,\cdots x_n)=\begin{cases}1 & x_1x_2\cdots x_n\in L\text{のとき}\\ 0 & x_1x_2\cdots x_n\notin L\text{のとき}\end{cases}$$

によって定義される関数列$\{f_n\}$は$O(\log n)$段実現可能である。

（証明） Lを認識する有限オートマトン

$M=(\{0,1\},K,\delta,q_0,F)$

を考える。ここに$\{0,1\}$は入力アルファベット、$K=\{q_0,q_1,\cdots,q_{m-1}\}$は状態集合、$\delta$は$K\times\{0,1\}$から$K$の中への写像、$q_0$は初期状態、$F\subseteq K$は最終状態の集合とする。今、$K^m$から$K^m$の中への自己写像$\mu$を考える。$i=0,1$に対し、

$\mu_i(q_0,q_1,\cdots,q_{m-1})=(\delta(i,q_0),\delta(i,q_1),\cdots,\delta(i,q_{m-1}))$

によりμ_0,μ_1を定義する。自己写像の全体が作るモノイドの中で、μ_0,μ_1から生成される有限半群をGとする。Gの大きさは、Mを与えれば一意に決まる。すなわち、$\mu_j\in G$を適当にコーディングすればGの乗法を計算する回路Iは、入力系列長nに依存しない定数段で構成できる。また、入力x_iからμ_0,μ_1を計算する回路I_pも定数段で実現できる。さらに、初期状態q_0が$\mu_j\in G$を施した後に写る状態q_iを計算し、$q_i\in F$ならば1を、$q_i\notin F$ならば0を出力する回路Zも定数段で作れる。半群Gにおいては、乗法に関する結合則が成立するので、図3のように、$\lceil\log_2 n\rceil-1$段のI回路、およびI_p回路、Z回路それぞれ1段ずつでf_nが計算できる。 □

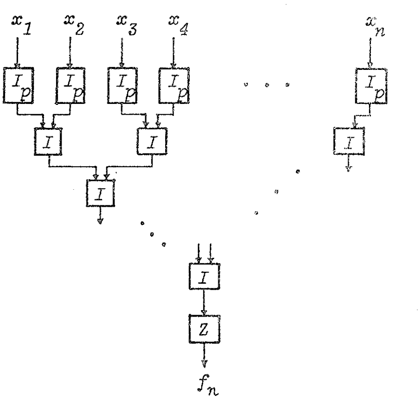

図3. 正規集合を *on set* とする関数の実現

ここで、$O(\log n)$段実現可能な関数列に対応する言語のクラス$\mathcal{L}_{log}$を次のように定義する。

$\mathcal{L}_{log}=\{L\{f_n\}\mid\{f_n\}$は$O(\log n)$段実現可能$\}$

言語$L=\{ww^R\mid w\in\{0,1\}^*$, w^Rはwの反転$\}$は、正規集合ではないが$\mathcal{L}_{log}$に含まれることが容易にわかる。これより、

〔性質5〕 正規集合のクラスは、$\mathcal{L}_{log}$に真に含まれる。

次の定理は、言語Lが$\mathcal{L}_{log}$に入るための十分条件を与える。

〔定理9〕 Lが次の3条件の1つでも満たすとき、Lは$\mathcal{L}_{log}$に含まれる。

(1) Lは正規集合である。

(2) LはCommutative language(1)である。すなわち、$a_1a_2\cdots a_n\in L$ならば、$a_{\pi(1)}a_{\pi(2)}\cdots a_{\pi(n)}\in L$である。ここに$\pi$は$(1,2,\cdots,n)$の任意の置換とする。

(3) $|L\cap\{0,1\}^n|<P(n)$ または

$|\bar{L}\cap\{0,1\}|<P(n)$ がすべての自然数で成立するような多項式$P(n)$が存在する。ここに、$\bar{L}$はLの補集合を表わす。

（証明） (1)は定理8に示した。

(2)は、Commutative languageに対応する関数列は対称関数族に含まれるから明らか。

(3)は、関数列中の任意の関数について、その *on set*, もしくは *off set* がある多項式でおさえられることを主張しているから、性質2より、この関数列は$O(\log n)$段実現可能である。 □

最後に、$\mathcal{L}_{log}$の演算の下での閉包性について述べる。

---

(1) Commutative language とは、$C_c(L)=L$となる言語である。ここに、C_cとはCommutative Closureと呼ばれ、L中の各語の記号の置換によって得られるすべての語からなる集合を表わす〔UEDAK6901〕

〔定理10〕 $\mathcal{L}_{log}$は，和集合，共通集合，補集合，連接，反転，Commutative Closure をとる各演算について閉じている．

（証明） $L_{\{f_n\}}$，$L_{\{g_n\}} \in \mathcal{L}_{log}$ と仮定する．

(1) 和集合．

$$L_{\{f_n\}} \cup L_{\{g_n\}} = L_{\{f_n+g_n\}}$$

$\{f_n+g_n\}$は$O(\log n)$段実現可能である．

(2) 共通集合

$$L_{\{f_n\}} \cap L_{\{g_n\}} = L_{\{f_n \cdot g_n\}}$$

$\{f_n \cdot g_n\}$は$O(\log n)$段実現可能である．

(3) 補集合

$$\overline{L}_{\{f_n\}} = L_{\{\overline{f_n}\}}$$

$\{\overline{f_n}\}$は$O(\log n)$段実現可能である．

(4) 反転

$$L^R_{\{f_n\}} = L_{\{h_n \mid h_n(x_1, x_2, \cdots, x_n) = f_n(x_n, x_{n-1}, \cdots, x_1)\}}$$

$\{h_n\}$は明らかに$O(\log n)$段実現可能である．

(5) 連接

$$L_{\{f_n\}} L_{\{g_n\}} = L_{\{h_n\}}$$

ここに，$\{h_n\} = \{\sum_{i=0}^{n} f_i(x_1, \cdots, x_i)\, g_{n-i}(x_{i+1}, \cdots, x_n)\}$と表わせる．ただしΣは論理和を表わすとする．$\{h_n\}$は，図4のように構成すれば，$O(\log n)$段実現可能であることがわかる．

(6) Commutative Closure

$C_c(L_{\{f_n\}})$に対応する関数列は対称関数族の部分集合である．□

6．あとがき

論理関数の計算時間を，その関数を実現する組合せ回路の段数で評価し，計算時間と入力変数の数の関係を議論した．すなわち，論理関数の集合において，各nに対し，集合中で最大の計算時間を持つn変数関数の計算時間を考え，これによってその集合の特徴付けを試みた．特に，$O(\log n)$段実現可能な集合について，ブール表現中のリテラル数との関係を与え，さらに，on setやoff set の大きさとの関係，集合中のn変数関数の数との関係を明らかにした．

また，よく知られた各種関数族について検討し，線形関数族，対称関数族，閾値関数族が$O(\log n)$段実現可能であることを示し，それぞれの構成法も与えた．一方，単調増大（減小）関数族，ユネイト関数族，自己双対関数族は$O(\log n)$段実現可能ではないことも示した．

最も基本的な関数集合として，各nに対し，n変数関数をちょうど1つずつ含むような関数列を導入し，その性質を調べた．関数列は$\{0,1\}^*$上の形式言語と1対1に対応することを利用し，関数列を形式言語の記述法で記述し，形式言語理論の導入を試みた．特に，正規集合やCommutative languageと対応する関数列が$O(\log n)$段実現可能であることを示した．

加算や乗算は$O(\log n)$段で実現可能であるが除算は，$O((\log n)^2)$段の構成法が知られてい

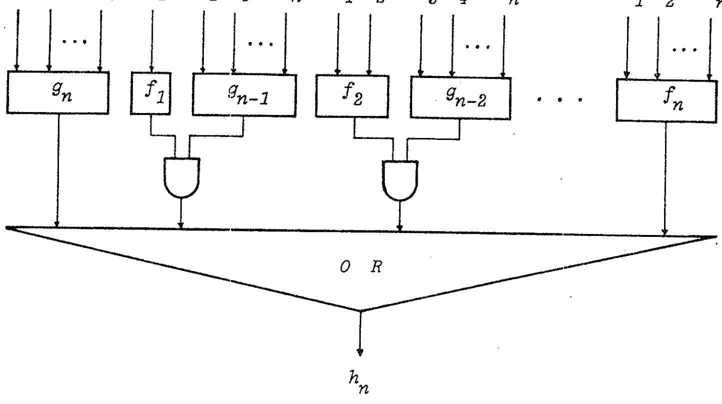

図4．言語の連接に対応する関数h_nの実現．

るだけで，$O(\log n)$段で構成できるかどうかはわかっていない．一方，回路規模の評価では，乗算と除算は同じオーダーの素子数で構成できることが知られている[SAVA76]．今後，本稿の議論を基に，このような現実的問題の解決をも考えていきたい．

謝辞

日頃，御討論頂く，本学上林弥彦助教授，稲垣耕作博士はじめ矢島研究室の諸氏に感謝します。

参考文献

KARA062 A.Karatsuba and Y.Ofman, "Multiplication of multidigit numbers with computers," Dokl. Akad. Nauk. SSSR, vol.145, no.2, p.293, 1962.

MCCO76 M.F.McColl, "The depth of Boolean functions," Edited by S.Michaelson and R.Milner, 'Automata Languages and Programming,' Edinburg University Press,pp.307-321, 1976.

MULLP7504 D.E.Muller and F.P.Preparata, "Bounds to complexity of network for sorting and for switching," J.ACM, vol.22, no.2, pp.195-201, April, 1975.

MUROI7606 室賀，茨木，北橋，しきい論理"，産業図書，1976年6月.

OFMA62 Y.Ofman, "On the algorithmic complexity of discrete functions," Dokl. Akad. Nauk. SSSR, vol.145, no.1, pp.48-51, 1962.

PREPM7705 F.P.Preparata and D.E.Muller, "Reduction of depth of Boolean networks with fan-in constraints," IEEE Trans Comput., vol.C-26, no.5, pp.474-479, May 1977.

SAVA76 J.E.Savage, "the complexity of computing," Wiley-Interscience, 1976.

SPIR6905 P.M.Spira, "On the computation time of cirtain classes of Boolean functions," Conf. Rec. ACM. Theory of Computing, may 1969.

SPIR7101 P.M.Spira, "On the time necessary to compute switching functions," IEEE Trans Comput. vol.C-20, no.1, pp.104-105, Jan. 1971.

SPIR7306 P.M.Spira, "Computation times of arithmetic and Boolean functions in (d,r) circuits," IEEE Trans Comput., vol.C-22, no.6, pp.552-555, June 1973.

UEDAK6901 上田，上林，清野，"Commutative Language について"，信学論，vol.52-C, no.10, pp.656-657, 1969年1月.

UNGE7704 S.H.Unger, "Tree realizations of iterative circuits," IEEE Trans Comput., vol.C-26, no.4, pp.365-383, April 1977.

WINO6504 S.Winograd, "On the time required to perform addition," J.ACM, vol.12, no.2, pp.277-285, April 1965.

WINO6710 S.Winograd, "On the time required to perform multiplication," J.ACM, vol.14, no.4, pp.793-802, Oct. 1967.

YASUY7807 安浦，矢島 "論理関数の計算時間について"，夏のLAシンポジウム，1978年7月.

YASUY7810 安浦，矢島 "$O(\log n)$段実現可能な論理関数" 関西連大，G6-11，1978年10月.