$\|y_{k+1}\|^2 \leq \|y_k\|^2$ and, consequently, the convergence of the norm sequence.

Lemma 2: Let Assumptions 1 and 2 hold, and let $y_1 \in \mathscr{L}(a_1, \cdots, a_m)$ in (4). Then the vector sequence $\{y_k\}_{k=1}^{\infty}$ converges to the zero vector.

Proof: Since $y_1 \in \mathscr{L}(a_1, \cdots, a_m)$, it is evident by induction that each $y_k \in \mathscr{L}(a_1, \cdots, a_m)$. Since the sequence $\{y_k\}$ by Lemma 1 is defined in a compact set, it has a cluster point $u \in \mathscr{L}(a_1, \cdots, a_m)$; furthermore, $\|y_k\| \geq \|u\|$ for each k. Choose $\mathscr{E} > 0$, and consider the set $\mathscr{U}_{\mathscr{E}} = \{x \in \mathscr{R}^n \mid \|x - u\| < \mathscr{E}, \|x\| \geq \|u\|\}$. Choose an index p such that $y_p \in \mathscr{U}_{\mathscr{E}}$. Then $y_{p+1} - u = y_p - u - \alpha_p[u^T a_{i_p} + (y_p - u)^T a_{i_p}]a_{i_p}$; if $u^T a_{i_p} = 0$, then it is easy to show that $\|y_{p+1} - u\| \leq \|y_p - u\|$, and thus $y_{p+1} \in \mathscr{U}_{\mathscr{E}}$, too. By induction, it is also evident that $y_{p+2} \in \mathscr{U}_{\mathscr{E}}$ if $u^T a_{i_{p+1}} = 0$, etc. However, since $u \in \mathscr{L}(a_1, \cdots, a_m)$, it cannot be orthogonal to every a_j unless it is the zero vector. Suppose, then, that $\|u\| = v > 0$. Because of Assumption 2, there must now be a first index $r > p$ such that $u^T a_{i_{r-1}} = 0$ and, consequently, $y_r \in \mathscr{U}_{\mathscr{E}}$, but $u^T a_{i_r} \neq 0$. It is straightforward to show that $\|y_{r+1}\| < v$ if $\mathscr{E}$ is chosen small enough. The following inequality is not difficult to establish:

$$\begin{aligned}\|y_{r+1}\|^2 &= \|y_r\|^2 - (2\alpha_r \|a_{i_r}\|^{-2} - \alpha_r^2)(y_r^T a_{i_r})^2 \|a_{i_r}\|^2 \\ &< v^2 + \mathscr{E}^2 + 2\mathscr{E}v - \gamma[(u^T a_{i_r})^2 - 2\mathscr{E}\|a_{i_r}\| \, |u^T a_{i_r}|] \\ &= v^2 - \gamma(u^T a_{i_r})^2 + 0(\mathscr{E})\end{aligned}$$

where some intervening steps using the Cauchy–Schwartz inequality and the triangle inequality have been left out. The positive scalar γ is the lower bound of $(2\alpha_r \|a_{i_r}\|^{-2} - \alpha_r^2)\|a_{i_r}\|^2$. Since $u^T a_{i_r} \neq 0$ with u a fixed vector and a_{i_r} a member of a finite vector set, it is evident that the choice of an $\mathscr{E}$ small enough leads to the contradiction $\|y_{r+1}\|^2 < v^2$. Thus, u must be the zero vector and by Lemma 1, the sequence $\{y_k\}$ converges to zero.

The theorem is now an obvious consequence of Lemma 2.

Proof of the Theorem: Let $x \in \mathscr{R}^n$ be arbitrary with the decomposition $x = x_1 + x_2$ where $x_1 \in \mathscr{L}(a_1, \cdots, a_m)$ and $x_2 \in \mathscr{L}^{\perp}(a_1, \cdots, a_m)$. Consider the sequence of vectors $T_k x = T_k x_1 + T_k x_2$. By Lemma 2, $T_k x_1$ converges to zero; on the other hand, $T_k x_2 = x_2$ for all k since x_2 is orthogonal to every a_j. Thus, $\lim_{k \to \infty} T_k x = x_2$, or the component of x on $\mathscr{L}^{\perp}(a_1, \cdots, a_m)$. Since x was arbitrary, the theorem has been proven.

If the set $\{a_1, \cdots, a_m\}$ spans $\mathscr{R}^n$, then the projection matrix on $\mathscr{L}^{\perp}(a_1, \cdots, a_m)$ is the zero matrix and the following corollary holds true.

Corollary: Let the assumptions of the theorem hold and let $\mathscr{L}(a_1, \cdots, a_m) = \mathscr{R}^n$. Then $\{T_k\}$ diverges to the zero matrix.

III. An Application: Adaptive Filling of an Associative Memory

In [3] and [4], the infinite product of (1) is used in a model of adaptive formation of an associative memory, associating a set of q-dimensional vectors $b_1, \cdots, b_m$ with the previously introduced n-dimensional vectors $a_1, \cdots, a_m$, according to the following recursion:

$$\begin{aligned}M_k &= M_{k-1} + \alpha_k(b_{i_k} - M_{k-1}a_{i_k})a_{i_k}^T \\ &= M_{k-1}(I - \alpha_k a_{i_k} a_{i_k})^T + \alpha_k b_{i_k} a_{i_k}^T \end{aligned} \tag{6}$$

with M_0 an arbitrary $(q \times n)$ matrix. Reid and Frame [7] have the same problem with all vectors a_j of unit norm and each α_j equal to one. It is now possible to show that under Assumptions 1 and 2, a necessary and sufficient condition for the convergence of $\{M_k\}$ is that the set of equations

$$\bar{M}a_j = b_j, \qquad j = 1, \cdots, m \tag{7}$$

has a solution, i.e., there exists an associative mapping $\bar{M}$ between the vector sets $\{a_1, \cdots, a_m\}$ and $\{b_1, \cdots, b_m\}$. If $\{M_k\}$ converges, then the limit matrix is a solution of (7), and has the explicit form [4]

$$\bar{M} = BA^+ + M_0(I - AA^+) \tag{8}$$

with B and A matrices having the vectors b_j and a_j as columns. An important special case assumed by Reid and Frame [7], in which (7) always has a solution, is the linear independency of the vector set $\{a_1, \cdots, a_m\}$.

If algorithm (6) were actually used for numerical computations to solve (7) for $\bar{M}$, then it would naturally be desirable to use an optimal choice of the free parameters α_r and i_r at each step r. It is easy to show that the speed of convergence of (6) is directly related to that of the norm sequence $\{\|y_k\|\}$ in (5), which immediately shows [because of (5)] that the best value for α_k would be $\|a_{i_k}\|^{-2}$, making the elementary matrix $I - \alpha_k a_{i_k} a_{i_k}^T$ idempotent. A good way to choose i_k would then be a cyclic sequence.

However, an iteration like (6) where all but one of the eigenvalues of the iteration matrices are unity has usually bad numerical behavior, and it may be questioned whether it would yield satisfactory results in large dimensional cases with many continuous-valued vectors, if small iteration errors were desired. On the other hand, if (6) is used as a model of a dynamical adaptive physical system, then the conditions imposed by Assumptions 1 and 2 cover a wide range of possibilities for the gain and input sequences.

References

[1] M. Altman, "On the approximate solution of linear algebraic equations," *Bull. Acad. Pol. Sci.*, vol. 5, pp. 365–370, Apr. 1957.
[2] ——, "An approximation process for the Gaussian least squares principle in the error theory," *Bull. Acad. Pol. Sci.*, vol. 5, pp. 371–374, Apr. 1957.
[3] T. Kohonen, "New analog associative memories," presented at the 3rd Int. Joint Conf. Artificial Intelligence, Stanford University, Stanford, CA, Aug. 20–23, 1973.
[4] T. Kohonen, E. Oja, and M. Ruohonen, "Adaptation of a linear system to a finite set of patterns occurring in an arbitrarily varying order," *Acta Polytech. Scand.*, vol. 25, Dec. 1974.
[5] T. Kohonen, *Associative Memory—A System-Theoretical Approach*, vol. 17 (Series Commun. Cybern.). New York: Springer-Verlag, 1977.
[6] L. D. Pyle, "A generalized inverse $\mathscr{E}$-algorithm for constructing intersection projection matrices, with applications," *Num. Math.*, vol. 10, pp. 86–102, May 1967.
[7] R. J. Reid and J. S. Frame, "Convergence in iteratively formed correlation matrix memories," *IEEE Trans. Comput.* (Corresp.), vol. C-24, pp. 827–830, Aug. 1975.
[8] G. J. Simmons, "Application of an associatively addressed, distributive memory," in *1964 Spring Joint Comput. Conf., AFIPS Conf. Proc.*, vol. 25. Washington, DC: Spartan, 1964, pp. 493–513.
[9] S. Watanabe and N. Pakvasa, "Subspace method in pattern recognition," in *Proc. 1st Int. J. Conf. Pattern Recognition*, 1973, pp. 25–32.

On the Number of Fanout-Free Functions and Unate Cascade Functions

TSUTOMU SASAO AND KOZO KINOSHITA

***Abstract*—In this correspondence, the number of functions and the number of equivalence classes of functions realized by fanout-free networks and cascades of AND's, OR's, and inverters are presented.**

Manuscript received April 15, 1977; revised October 1977 and January 16, 1978.
The authors are with the Department of Electronic Engineering, Osaka University, Osaka, Japan.

For fanout-free functions, recursive formulas for $T_{ND}(n)$ and $\Phi_{ND}(n)$, the number of n-p-n-equivalence classes of n-variable functions and p-equivalence classes of n-variable functions, respectively, are derived. For unate cascade functions, a recursive formula for $\psi_{ND}(n)$, the number of n-variable functions, and formulas for $U_{ND}(n)$ and $\Psi_{ND}(n)$, the number of n-p-n-equivalence classes and p-equivalence classes, respectively, are derived. Some asymptotic properties of $\psi_{ND}(n)$, $U_{ND}(n)$, and $\Psi_{ND}(n)$ are also examined and it is shown that $\psi_{ND}(n)/\psi(n) \to 1/\sqrt{2}$, $U_{ND}(n)/U(n) \to 1/2$, and $\Psi_{ND}(n)/\Psi(n) \to 1/\sqrt{2}$ as $n \to \infty$, where $\psi(n)$ is the number of distinct unate cascade functions of up to n variables, and $U(n)$ and $\Psi(n)$ are the number of distinct n-p-n- and p-equivalence classes of unate cascade functions of up to n variables, respectively.

Index Terms—**Cascade, disjunctive networks, enumeration of equivalence classes, enumeration of switching functions, fanout-free function, threshold function, unate function.**

I. Introduction

In this correspondence, several previously unsolved enumeration problems are considered. The first considered here concerns *fanout-free functions*, which can be realized by circuits satisfying the following restrictions.

1) They are constructed from the two-input unate gates (AND, OR, NAND, and NOR, etc.) and NOT gates (inverters).

2) The fanout of each gate is one.

3) Each primary input line connects to the input of exactly one gate.

The class of fanout-free functions is a special class of functions which are realized by *disjunctive networks*. Disjunctive networks satisfy the restriction 1′) instead of 1) in addition to 2) and 3) stated above.

1′) They are constructed from arbitrary two-input gates and NOT gates.

Disjunctive networks have been studied by Levy, Winder, and Mott [1], Maruoka and Honda [2], and Butler and Breeding [3].

Butler [4] has recently derived expressions for $N_{\text{dis}}(n)$, the number of n-variable functions realized by disjunctive circuits constructed from EXCLUSIVE-OR gates as well as unate gates.

Hayes [5] has recently derived expressions for $\phi_{ND}(n)$, the number of n-variable fanout-free functions.

In Section III, we derive formulas for $T_{ND}(n)$ and $\Phi_{ND}(n)$, the number of n-p-n-equivalence classes and p-equivalence classes, respectively, of n-variable fanout-free functions. These formulas have been computed for values of n up to 8.

The second enumeration problem considered here concerns *unate cascade functions* which can be realized by unate cascade circuits satisfying restrictions 1), 2), and 3) stated above and the following.

4) Each gate connects to at most one output line of another gate.

The class of unate cascade function is a special class of *cascade realizable functions*. A cascade satisfies restrictions 1′), 2), 3), and 4). Cascades have a number of interesting properties and have been the subject of many papers [6]–[12].

Frécon [24] has considered many enumeration problems for cascades. Chakrabarti and Kolp [20], and Butler [4] also have derived expressions for the number of n-variable functions realized by cascades constructed from EXCLUSIVE-OR gate as well as unate gates. Sklansky, Korenjak, and Stone [13] have derived expressions for the number of n-p-equivalence classes of cascade realizable functions. Enumeration problems dealing with other types of functions have been considered in the literature [14]–[16], [21].

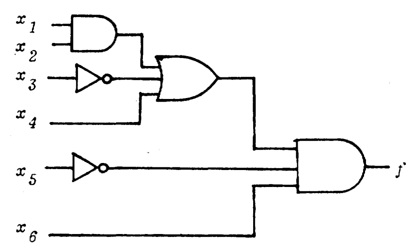

Fig. 1. Fanout-free circuit.

In Section IV, we derive formulas for $\psi_{ND}(n)$, the number of n-variable unate cascade functions, $U_{ND}(n)$ and $\Psi_{ND}(n)$, the numbers of n-p-n-equivalence classes and p-equivalence classes, respectively, of n-variable unate cascade functions. These formulas have been computed for values of n up to 8. Some asymptotic properties of ψ_{ND}, U_{ND}, and Ψ_{ND} are also examined.

II. Background

In this correspondence, it is assumed that circuits are constructed using unate elements (AND, OR, NAND, NOR, and NOT, etc.) only. First we define fanout-free functions and fanout-free circuits.

Definition 2.1 [5]: The functions 0, 1, x, $\bar{x}$, are fanout-free. If $f_1(X_1)$ and $f_2(X_2)$ are fanout-free functions and $\{X_1\} \cap \{X_2\} = \varnothing$, then $\bar{f}_1(X_1)$, $f_1(X_1) \cdot f_2(X_2)$, and $f_1(X_1) \vee f_2(X_2)$ are fanout-free, where $\{X_1\}$ and $\{X_2\}$ denote the sets of variables in X_1 and X_2, respectively. The only fanout-free functions are given above.

Definition 2.2 [5]: A combinational circuit N is fanout-free if it has a single primary output line, and every other line in N is connected to an input of exactly one gate.

The foregoing definitions imply that a function is fanout-free if and only if it can be realized by a fanout-free circuit. For example, the function realized by the circuit of Fig. 1 is fanout-free.

It can easily be shown that every fanout-free function is unate, but the converse is false.

Definition 2.3 [5]: Let $f(X)$ be any function. The AND rank of f is the largest number r such that there exist r functions $f_1, f_2, \cdots, f_r$ and a partition $\{X_1, X_2, \cdots, X_r\}$ of X with the property

$$f(X) = f_1(X_1) f_2(X_2) \cdots f_r(X_r)$$

and

$$\{X_i\} \cap \{X_j\} = \varnothing \; (i \neq j).$$

A function $f(X)$ is termed degenerate if it is independent of one or more variables in $\{X\}$; otherwise it is nondegenerate.

Definition 2.4 [5]: $\phi(n)$ is the number of distinct fanout-free functions of up to n variables. $\phi_D(n)$ and $\phi_{ND}(n)$ denote the number of degenerate and nondegenerate fanout-free functions of n variables, respectively.[1]

Clearly

$$\phi(n) = \phi_D(n) + \phi_{ND}(n),$$

and

$$\phi_D(n) = \sum_{i=0}^{n-1} C(n, i) \cdot \phi_{ND}(i),$$

where $C(n, i)$ denotes the number of combinations choosing i objects out of n objects.

Definition 2.5 [5]: $A(n, r)$ is the number of nondegenerate fanout-free functions of n variables with AND rank r.

[1] The subscripts D and ND denote degenerate and nondegenerate functions, respectively. Similar notation will be used throughout this correspondence.

TABLE I
THE NUMBER OF FANOUT-FREE FUNCTIONS AND UNATE CASCADE FUNCTIONS

	Fanout-free function			Unate cascade function		
	number of functions	number of P-eq. classes	number of NPN-eq. classes	number of functions	number of P-eq. classes	number of NPN-eq. classes
n	$\phi_{ND}(n)$	$\Phi_{ND}(n)$	$T_{ND}(n)$	$\psi_{ND}(n)$	$\Psi_{ND}(n)$	$U_{ND}(n)$
1	2	2	1	2	2	1
2	8	6	1	8	6	1
3	64	20	2	64	20	2
4	832	80	5	736	68	4
5	15104	340	12	10624	232	8
6	352256	1570	33	183936	792	16
7	10037248	7540	90	3715072	2704	32
8	337936384	37610	261	85755392	9232	64

Lemma 2.1 [5]:

$$A(n, 1) = \sum_{r=2}^{n} A(n, r), \; \phi_{ND}(n) = 2A(n, 1), \qquad (n \geq 2).$$

Table I shows the values of $\phi_{ND}(n)$ for $n \leq 8$.

III. THE NUMBER OF EQUIVALENCE CLASSES OF FANOUT-FREE FUNCTIONS

In this section, we will derive recursive formulas for the number of n-p-n-equivalence classes and p-equivalence classes of fanout-free functions.

Definition 3.1: f is n-p-n-equivalent to g, denoted by $f \overset{\text{n-p-n}}{\sim} g$, if g can be obtained from f by any combination of the following three operations:

1) Negation of one or more variables of f.
2) Permutation of variables of f.
3) Negation of f.

Definition 3.2: f is p-equivalent to g, denoted by $f \overset{\text{p}}{\sim} g$, if g can be obtained by permutation of variables of f.

Obviously, the binary relations $\overset{\text{n-p-n}}{\sim}$ and $\overset{\text{p}}{\sim}$ are both equivalence relations.

Definition 3.3 [5]: The circuits which satisfy the following structural constraints are said to be type A circuits.

1) They are fanout-free and contain AND, OR, and NOT gates (inverters) only.
2) Inverters can only appear in the primary input lines of the circuits, with at most one inverter per primary input.
3) Every AND and OR gate has at least two input lines, and AND(OR) gate cannot be directly connected to the input of another AND(OR) gate, i.e., AND and OR gates must alternate along every path in the circuit.

Every fanout-free function has a unique type A realization.

Example 3.1: Fig. 1 shows the type A realization of $f(X) = (x_1 x_2 \vee \bar{x}_3 \vee x_4)\bar{x}_5 x_6$.

Definition 3.4: The rooted tree T which satisfies the following conditions is called *the tree of a fanout-free function f*. In the type A realization of f:

1) The output gate corresponds to the root of T.
2) Primary input terminals, AND gates, and OR gates correspond to nodes of T. (Inverters are neglected.)
3) Input lines of gates correspond to edges of t.

Example 3.2: Fig. 2 shows the tree of the fanout-free function of Example 3.1.

Lemma 3.1: If f and g are fanout-free functions and $f \overset{\text{n-p-n}}{\sim} g$, then the trees of f and g are isomorphic.

The tree of an n-variable fanout-free function has exactly n leaves (nodes whose out-degrees are zeros), and the out-degree of every node except the leaves of the tree is at least two. Conversely, a tree which satisfies these conditions corresponds to an n-p-n-equivalence class of a fanout-free function.

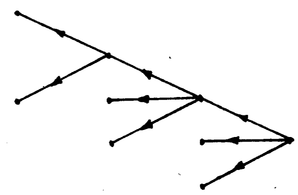

Fig. 2. Tree of $f(X) = (x_1 x_2 \vee \bar{x}_3 \vee x_4)\bar{x}_5 x_6$.

Next, we will derive a recursive formula for $T_{ND}(n)$, the number of distinct n-p-n-equivalence classes of n-variable fanout-free functions. Since the functions which belong to an n-p-n-equivalence class of a fanout-free function have a unique tree, $T_{ND}(n)$ is equal to the number of distinct trees of n-variable fanout-free functions.

Example 3.3: Fig. 3 shows all the trees of fanout-free functions of up to four variables.

Definition 3.5: $N(n)$ is the number of distinct trees of n-variable nondegenerate fanout-free functions. $N(n, r)$ is the number of trees of n-variable nondegenerate fanout-free functions whose roots have outdegree r.

Theorem 3.1:[2]

$$T_{ND}(n) = N(n); \; N(1) = 1;$$

$$N(n) = \sum_{r=2}^{n} N(n, r), \qquad \text{for } n \geq 2;$$

$$N(n, r) = \sum\nolimits^{1} v(i_1, i_2, \cdots, i_r);$$

where

$$v(i_1, i_2, \cdots, i_r) = \prod_{k=1}^{s}{}^{2} \, C(N(j_k) + a_k - 1, a_k).$$

$\sum^1$ and $\prod^2$ denote the computation over the combinations such that

$$(i_1, i_2, \cdots, i_r) = (\underbrace{j_1, j_1, \cdots, j_1}_{a_1}, \underbrace{j_2, j_2, \cdots, j_2}_{a_2}, \cdots, \underbrace{j_s, j_s, \cdots, j_s}_{a_s})$$

$$= ((j_1)^{a_1}, (j_2)^{a_2}, \cdots, (j_s)^{a_s});$$

[2] Cayley [23] has calculated $N(n)$, but [23] does not contain the recursive formula. We will include the proof because similar technique will be used throughout this correspondence.

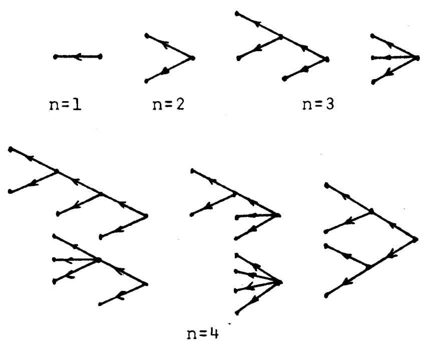

Fig. 3. Trees of fanout-free functions of up to four variables.

$$i_1 \le i_2 \le \cdots \le i_r;\ i_1 + i_2 + \cdots + i_r = n;$$

$$j_1 < j_2 < \cdots < j_s \quad \text{and} \quad a_1 + a_2 + \cdots + a_s = r.$$

Proof: Obviously,

$$N(n) = \sum_{r=2}^{n} N(n, r), \qquad \text{for } n \ge 2.$$

To obtain $N(n, 2)$, consider the tree structure with the root of out-degree two, as shown in Fig. 4. Suppose this tree structure has two subtrees of i_1 variables and i_2 variables $(i_1 + i_2 = n)$, and let $v(i_1, i_2)$ be the number of distinct trees of this type. When $i_1 < i_2$, we obtain

$$v(i_1, i_2) = N(i_1)N(i_2) = C(N(i_1), 1) \cdot C(N(i_2), 1).$$

When $i_1 = i_2$, by considering the symmetry, we obtain

$$v(i_1, i_2) = C(N(i_1) + 1, 2).$$

To obtain $N(n, r)$, consider the tree structure with the root of out-degree r, as shown in Fig. 5. Suppose this tree structure has r subtrees of i_1 variables, i_2 variables, $\cdots$, i_r variables $(i_1 + i_2 + \cdots + i_r = n)$, and let $v(i_1, i_2, \cdots, i_r)$ be the number of distinct trees of this type. When $i_1 < i_2 < \cdots < i_r$, we have

$$\begin{aligned} v(i_1, i_2, \cdots, i_r) &= N(i_1)N(i_2) \cdots N(i_r) \\ &= C(N(i_1), 1) \cdot C(N(i_2), 1) \\ &\quad \cdot \cdots \cdot C(N(i_r), 1). \end{aligned}$$

When

$$i_1 \le i_2 \le \cdots \le i_r$$

and

$$(i_1, i_2, \cdots, i_r) = ((j_1)^{a_1}, (j_2)^{a_2}, \cdots, (j_s)^{a_s})$$

and

$$j_1 < j_2 < \cdots < j_s,$$

we have

$$\begin{aligned} v(i_1, i_2, \cdots, i_r) &= C(N(j_1) + a_1 - 1, a_1) \\ &\quad \cdot C(N(j_2) + a_2 - 1, a_2) \\ &\quad \cdot \cdots \cdot C(N(j_s) + a_s - 1, a_s). \end{aligned}$$

Hence, we obtain the theorem. Q.E.D.

Table I shows the values of $T_{ND}(n) = N(n)$ for $n \le 8$.

Next, we will derive the recursive formula for $\Phi_{ND}(n)$, the number of distinct p-equivalence classes of n-variable fanout-free functions of n variables.

Definition 3.6: $B(n, r)$ is the number of distinct p-equivalence classes of n-variable nondegenerate fanout-free functions with AND rank r.

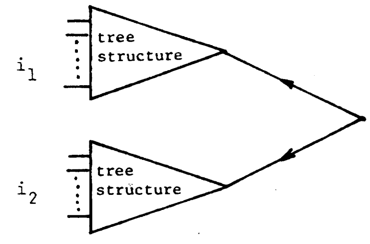

Fig. 4. Enumeration of $v(i_1, i_2)$.

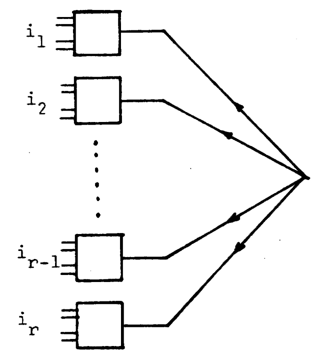

Fig. 5. Enumeration of $v(i_1, i_2, \cdots, i_r)$.

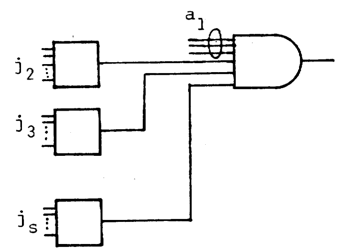

Fig. 6. Illustration of Theorem 3.2.

Theorem 3.2:

$$\Phi_{ND}(n) = 2 \cdot B(n, 1);\ B(1, 1) = 1;$$

$$B(n, 1) = \sum_{r=2}^{n} B(n, r), \qquad (n \ge 2);$$

$$B(n, r) = \sum{}^{3} \mu(i_1, i_2, \cdots, i_r);$$

where

$$\mu(i_1, i_2, \cdots, i_r) = (a_1 + 1) \cdot \prod_{k=2}^{s}{}^{4} C(B(j_k) + a_k - 1, a_k);$$

and

$$B(j_k) = B(j_k, 1).$$

$\sum^3$ and $\prod^4$ denote the computation over all combinations such that

$$(i_1, i_2, \cdots, i_r) = ((1)^{a_1}, (j_2)^{a_2}, \cdots, (j_s)^{a_s}),$$

$$i_1 \le i_2 \le \cdots \le i_r,\ i_1 + i_2 + \cdots + i_r = n,$$

$$1 < j_2 < \cdots < j_s, \quad \text{and} \quad a_1 + a_2 + \cdots + a_s = r.$$

Proof: When the circuit is drawn as in Fig. 6, we can write

$$(i_1, i_2, \cdots, i_r) = ((1)^{a_1}, j_2, j_3, \cdots, j_s).$$

Let $\mu(i_1, i_2, \cdots, i_r)$ be the number of distinct p-equivalence classes. In Fig. 6, there are $(a_1 + 1)$ ways of complementing the variables of the AND gate, since there may be no inverter, one inverter, two inverters, $\cdots$, and a_1 inverters. When $1 < j_2 < \cdots < j_s$, we have

$$\mu(i_1, i_2, \cdots, i_r) = (a_1 + 1)B(j_2)B(j_3) \cdots B(j_s) = (a_1 + 1) \cdot C(B(j_2), 1) \cdot C(B(j_3), 1) \cdot \cdots \cdot C(B(j_s), 1).$$

When

$$i_1 \le i_2 \le \cdots \le i_r$$

and

$$(i_1, i_2, \cdots, i_r) = ((1)^{a_1}, (j_2)^{a_2}, \cdots, (j_s)^{a_s}),$$

and

$$1 < j_2 < j_3 < \cdots < j_s.$$

Similarly, we have

$$\mu(i_1, i_2, \cdots, i_r) = (a_1 + 1)C(B(j_2) + a_2 - 1, a_2)C(B(j_3) + a_3 - 1, a_3) \cdot \cdots \cdot C(B(j_s) + a_s - 1, a_s).$$

The first and the second equations can be proven in a similar way to Lemma 2.1. Hence, we obtain the theorem. Q.E.D.

Table I shows the values of $\Phi_{ND}(n)$ for $n \le 8$.

IV. The Number of Unate Cascade Functions

In this section, we will derive formulas for $\psi_{ND}(n)$, the number of n-variable unate cascade functions, $U_{ND}(n)$, the number of n-p-n-equivalence classes of n-variable unate cascade function, and $\Psi_{ND}(n)$, the number of p-equivalence classes of n-variable unate cascade functions.

Definition 4.1: The functions 0, 1, x, $\bar{x}$ are unate cascade functions. If $f(X)$ is unate cascade function and $\{X\} \cap \{x_{n+1}\} = \varnothing$, then $\bar{f}(X)$, $f(X) \cdot x_{n+1}^*$, and $f(X) \vee x_{n+1}^*$ are unate cascade functions, where x_{n+1}^* denotes x_{n+1} or $\bar{x}_{n+1}$. The only unate cascade functions are those given above.

Example 4.1: The function of Example 3.1 is a unate cascade function.

It is known that a unate cascade function can be realized by a cascade of two-input unate elements.

Lemma 4.1 [18]: $f(X)$ is a unate cascade function if and only if $f(X)$ is a fanout-free threshold function.

The class of unate cascade function is a special class of threshold functions.

Definition 4.2: $E(n, r)$ is the number of distinct n-variable nondegenerate unate cascade functions with AND rank r.

Lemma 4.2: For

$$n > r \ge 2, \qquad E(n, r) = C(n, r-1) \cdot 2^{r-1} \cdot E(n - r + 1, 1).$$

For

$$n \ge 1, \qquad E(n, n) = 2^n.$$

Proof: An n-variable unate cascade function with AND rank r can be realized as shown in Fig. 7, where the output line of an $(n - r + 1)$-variable unate cascade circuit is connected to the r-input AND gate. In Fig. 7, there are $C(n, r-1)$ ways of choosing $(r - 1)$ variables from n, and 2^{r-1} ways of complementing the variables. So we have

$$E(n, r) = C(n, r-1) \cdot 2^{n-1} \cdot E(n - r + 1, 1)$$

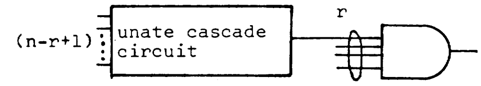

Fig. 7. Illustration of Lemma 4.2.

for $n > r$. There are 2^n ways of complementing the variables of an n-input AND gate. So we have $E(n, n) = 2^n$. Hence, we obtain the lemma. Q.E.D.

Theorem 4.1:[3]

$$\psi_{ND}(n) = 2 \cdot E(n),$$

where

$$E(1) = 2, \; E(2) = 4$$

and

$$E(n) = \sum_{r=2}^{n-1} C(n, r-1) \cdot 2^{r-1} \cdot E(n - r + 1) + 2^n, \qquad (n \ge 3).$$

Proof: By Lemma 4.2

$$E(n, 1) = \sum_{r=2}^{n-1} C(n, r-1) \cdot 2^{r-1} \cdot E(n - r + 1, 1) + 2^n; \quad (n \ge 2).$$

Let $E(n) = E(n, 1)$, similar to Lemma 2.1, we have

$$\psi_{ND}(n) = \sum_{r=1}^{n} E(n, r) = 2 \cdot E(n, 1) = 2 \cdot E(n). \qquad \text{Q.E.D.}$$

Table I shows the values of $\psi_{ND}(n)$ for $n \le 8$.

Conjecture 4.1:[3] For large n, $\psi_{ND}(n) \simeq \alpha\beta^n \cdot n!$, where

$$\alpha \simeq 0.4426950, \; \beta = (2/\log_e 2) \simeq 2.88539008.$$

Informal proof supporting the conjecture: Let $\psi_{ND}(n) = \alpha\beta^n n!$. Substituting it into the formula of Theorem 4.1, we obtain

$$(\tfrac{1}{2})\alpha\beta^n n! = \sum_{r=2}^{n-1} C(n, r-1)2^{r-1}(\tfrac{1}{2})\alpha\beta^n \cdot \beta^{-(r-1)} \cdot (n - r + 1)! + 2^n.$$

Dividing both sides by $(\frac{1}{2})\alpha\beta^n n!$, we have

$$1 = \sum_{r=2}^{n-1} \frac{2^{r-1}}{(r-1)!} \beta^{-(r-1)} + \frac{2^n}{\frac{1}{2}\alpha\beta^n n!}.$$

As $n \to \infty$, we have

$$1 = \sum_{r=2}^{\infty} \frac{(2/\beta)^{r-1}}{(r-1)!} = \sum_{k=1}^{\infty} \frac{(2/\beta)^k}{k!} = e^{(2/\beta)} - 1.$$

Thus $\beta = (2/\log_e 2) \simeq 2.88539008$.

Let $\psi_{ND}(n) = \alpha(2/\log_e 2)^n \cdot n!$. By using the value of $\psi_{ND}(n)$ for $n = 8$ shown in Table I, we have $\alpha \simeq \psi_{ND}(8)/(\beta^8 \cdot 8!) \simeq 0.4426950$. Thus, we obtain the conjecture.

This approximation is very good. For $n = 2, 3, 4, 5, 6$, and 7, the percentage errors are 7.9, 0.3, 0.06, 0.005, 0.0006, and 0.00008 percent, respectively.

Conjecture 4.2: $\psi_{ND}(n)/\psi(n) \to 1/\sqrt{2}$ $(n \to \infty)$.

Informal proof supporting the conjecture: Let $\psi_{ND}^*(n) = \alpha\beta^n n!$ and $\psi_D^*(n)$ be approximations for $\psi_{ND}(n)$ and $\psi_D(n)$, respectively. $(\beta = 2/\log_e 2.)$ Obviously

$$\psi_D^*(n) = \sum_{i=0}^{n-1} \binom{n}{i} \psi_{ND}^*(i) = \sum_{i=0}^{n-1} \frac{n!}{i!\,(n-i)!} \alpha\beta^i \cdot i!.$$

[3] After the submission of this correspondence, the authors were informed that Frécon [24] and Bender and Butler [22] have independently obtained similar results. In particular Frécon has obtained an upper bound for $\psi_{ND}(n)$, and Bender and Butler have shown that $\psi_{ND}(n) \simeq \alpha\beta^n n!$, where $\alpha = \beta/2 - 1$ and $\beta = (2/\log_e 2)$.

Let $n - i = k$. We have

$$\psi_D^*(n) = \alpha \sum_{k=1}^{n} \frac{n!}{k!} \cdot \beta^{(n-k)} = \psi_{ND}^*(n) \cdot \sum_{k=1}^{n} \frac{(1/\beta)^k}{k!}.$$

Clearly

$$\frac{\psi^*(n)}{\psi_{ND}^*(n)} = \frac{\psi_{ND}^*(n) + \psi_D^*(n)}{\psi_{ND}^*(n)} = 1 + \frac{\psi_D^*(n)}{\psi_{ND}^*(n)}.$$

Hence, as $n \to \infty$

$$\frac{\psi^*(n)}{\psi_{ND}^*(n)} \to 1 + \sum_{k=1}^{\infty} \frac{(1/\beta)^k}{k!} = e^{(1/\beta)} = \sqrt{2}.$$

Thus, we have the conjecture.

The authors have not obtained complete proof of this conjecture, but the data of Table I show that $\psi_{ND}(n)/\psi(n)$ approaches to $1/\sqrt{2}$ as n increases. For $n = 3, 4, 5, 6, 7$, and 8, the percentage errors are within 6, 0.9, 0.08, 0.004, 0.0002, and 0.0003 percent, respectively.

Theorem 4.2:

$$U_{ND}(n) = M(n),$$

where

$$M(1) = 1, \quad \text{and for } n \geq 2,\ M(n) = 2^{n-2}.$$

Proof: The class of unate cascade functions is a special class of fanout-free functions. Therefore, the proof can be done in a similar way to that of Theorem 3.1. Let $M(n, r)$ be the number of distinct trees of n-variable nondegenerate unate cascade function whose roots have out-degree r, and let $M(n) = M(n, 1)$. Since

$$(i_1, i_2, \cdots, i_r) = ((1)^{r-1}, n - r + 1),$$

we have

$$\begin{aligned} v(i_1, i_2, \cdots, i_r) &= C(M(1) + (r-1) + 1, r-1) \\ &\quad \cdot C(M(n-r+1), 1) \\ &= M(n-r+1). \end{aligned}$$

Therefore, we obtain $M(n, r) = M(n - r + 1)$, and

$$M(n) = \sum_{r=2}^{n} M(n, r) = \sum_{r=2}^{n} M(n - r + 1) = \sum_{i=1}^{n-1} M(i),$$

where

$$M(1) = 1.$$

Hence, we obtain

$$M(n) = 2^{n-2}, \qquad (n \geq 2). \qquad \text{Q.E.D.}$$

Theorem 4.3:

$$U_{ND}(n)/U(n) \to \tfrac{1}{2}, \qquad \text{as } n \to \infty.$$

Proof: Obviously, we have

$$U_D(n) = \sum_{i=0}^{n-1} U_{ND}(i) = 1 + 1 + \sum_{i=2}^{n-1} 2^{i-2} = 2^{n-2} + 1.$$

Therefore

$$\frac{U_{ND}(n)}{U(n)} = \frac{U_{ND}(n)}{U_{ND}(n) + U_D(n)} = \frac{2^{n-2}}{2^{n-2} + 2^{n-2} + 1} = \frac{1}{2 + 2^{2-n}}.$$

Hence

$$\frac{U_{ND}(n)}{U(n)} \to \frac{1}{2}, \qquad \text{as } n \to \infty. \qquad \text{Q.E.D.}$$

Definition 4.3: $D(n, r)$ is the number of distinct p-equivalence classes of nondegenerate n-variable unate cascade functions with AND rank r.

Lemma 4.3:

$$\begin{aligned} &D(1, 1) = 1; \\ &D(n, r) = rD(n - r + 1, 1), && (n > r); \\ &D(n, n) = n + 1, && (n \geq 2); \\ &D(n, 1) = \sum_{r=2}^{n} D(n, r), && (n \geq 2). \end{aligned}$$

Proof: An n-variable unate cascade function with AND rank r can be realized as shown in Fig. 7. There are r ways of complementing the variables of the AND gate. Therefore

$$D(n, r) = r \cdot D(n - r + 1, 1).$$

There are $(n + 1)$ ways of complementing the variable of the AND gate with AND rank n. So, $D(n, n) = n + 1$. Similar to Lemma 2.1, we have

$$D(n, 1) = \sum_{j=2}^{n} D(n, j). \qquad \text{Q.E.D.}$$

Theorem 4.4:

$$\Psi_{ND}(n) = (\tfrac{1}{2})\{(2 + \sqrt{2})^n + (2 - \sqrt{2})^n\}, \qquad (n \geq 1).$$

Proof: Let $D(n) = D(n, 1)$. By Lemma 4.3, we have

$$D(n) = \sum_{r=2}^{n-1} r \cdot D(n - r + 1) + (n + 1), \qquad (n \geq 3).$$

This recursive formula satisfies the following relation:

$$D(n) - 4 \cdot D(n-1) + 2 \cdot D(n-2) = 0. \tag{4.1}$$

By solving (4.1) under the conditions of $D(1) = 1$ and $D(2) = 3$, we have

$$D(n) = (\tfrac{1}{4})\{(2 + \sqrt{2})^n + (2 - \sqrt{2})^n\}.$$

Since $\Psi_{ND}(n) = 2 \cdot D(n, 1) = 2 \cdot D(n)$, we have the theorem.

Q.E.D.

Theorem 4.5:

$$\Psi_{ND}(n)/\Psi(n) \to 1/\sqrt{2} \qquad \text{as } n \to \infty.$$

Proof: By Theorem 4.4, $\Psi_{ND}(n)$ can be written as

$$\Psi_{ND}(n) = \alpha(\beta^n + \gamma^n),$$

where

$$\alpha = \tfrac{1}{2},\ \beta = 2 + \sqrt{2} \quad \text{and} \quad \gamma = 2 - \sqrt{2}.$$

Clearly

$$\begin{aligned} \Psi_D(n) &= \sum_{i=0}^{n-1} \Psi_{ND}(i) = \sum_{i=0}^{n-1} \alpha(\beta^i + \gamma^i) \\ &= \alpha\left\{\frac{\beta^n - 1}{\beta - 1} + \frac{1 - \gamma^n}{1 - \gamma}\right\}. \end{aligned}$$

Therefore, we have

$$\Psi_D(n) \to \frac{1}{\beta - 1} \Psi_{ND}(n) \qquad \text{as } n \to \infty.$$

Hence

$$\frac{\Psi_{ND}(n)}{\Psi(n)} = \frac{\Psi_{ND}(n)}{\Psi_{ND}(n) + \Psi_D(n)} \to \frac{1}{1 + 1/(\beta - 1)}$$

$$= \frac{1}{\sqrt{2}} \quad \text{as } n \to \infty. \qquad \text{Q.E.D.}$$

It is interesting to compare $\psi_{ND}(n)$ (the number of unate cascade functions) with $N_{cas}(n)$ (the number of cascade realizable functions). The class of cascade realizable functions includes all the unate cascade functions as well as many nonunate functions. Butler [4] has shown that for large n

$$N_{cas}(n) \simeq \xi \eta^n \cdot n!, \qquad \text{where } \xi \simeq 0.28790 \text{ and } \eta \simeq 4.04095.$$

Theorem 4.2 shows that for large n

$$\frac{\psi_{ND}(n)}{N_{cas}(n)} = \frac{\alpha\beta^n \cdot n!}{\xi\eta^n \cdot n!} = \left(\frac{\alpha}{\xi}\right)\cdot\left(\frac{\beta}{\eta}\right)^n.$$

Thus the fraction of unate cascade functions tends to zero as $n \to \infty$.

It is also interesting to compare $2 \cdot U_{ND}(n)$ (the number of n-p-equivalence classes of unate cascade functions) with B_n (the number of n-p-equivalence classes of cascade realizable functions).[4] Sklansky, Korenjak, and Stone [13] have shown that

$$B_n = \frac{1}{\sqrt{5}}\left\{\left(\frac{3+\sqrt{5}}{2}\right)^n - \left(\frac{3-\sqrt{5}}{2}\right)^n\right\} \simeq \frac{(2.62)^n}{2.24} \text{ for large } n.$$

Theorem 4.3 shows that for large n

$$2 \cdot \frac{U_{ND}(n)}{B_n} \simeq \frac{2.24 \times 2}{4}\left(\frac{2}{2.62}\right)^n.$$

Thus the fraction of the number of n-p-equivalence classes of unate cascade functions tends to zero as $n \to \infty$.

The class of unate cascade functions is a special class of fanout-free functions. Hayes [5] has shown that almost all fanout-free functions are nondegenerate, while Conjecture 4.2 shows that $1/\sqrt{2}$ of unate cascade functions are nondegenerate.

Acknowledgment

The authors wish to acknowledge the support and encouragement of Prof. H. Ozaki of Osaka University, Osaka, Japan.

The authors also wish to thank referees for their comments. [22], [23], and [24] were noticed by the referees.

References

[1] S. Y. Levy, R. O. Winder, and T. H. Mott, Jr., "A note on tributary switching networks," *IEEE Trans. Electron. Comput.*, vol. EC-13, pp. 148–151, Apr. 1964.
[2] A. Maruoka and N. Honda, "Logical networks of flexible cells," *IEEE Trans. Comput.*, vol. C-22, pp. 347–358, Apr. 1973.
[3] J. T. Butler and K. J. Breeding, "Some characteristics of universal cell nets," *IEEE Trans. Comput.*, vol. C-22, pp. 897–903, Oct. 1973.
[4] J. T. Butler, "On the number of functions realized by cascades and disjunctive networks," *IEEE Trans. Comput.*, vol. C-24, pp. 681–690, July 1975.
[5] J. P. Hayes, "Enumeration of fanout-free Boolean functions," *J. Ass. Comput. Mach.*, vol. 23, pp. 700–709, Oct. 1976.
[6] K. K. Maitra, "Cascaded switching networks of two-input flexible cells," *IRE Trans. Electron. Comput.*, vol. EC-11, pp. 136–143, Apr. 1962.
[7] J. Sklansky, "General synthesis of tributary switching networks," *IEEE Trans. Electron. Comput.*, vol. EC-12, pp. 464–469, Oct. 1963.
[8] R. C. Minnick, "Cutpoint cellular logic," *IEEE Trans. Electron. Comput.*, vol. EC-13, pp. 685–698, Dec. 1964.
[9] H. S. Stone and A. J. Korenjak, "Canonical form and synthesis of cellular cascade," *IEEE Trans. Electron. Comput.*, vol. EC-14, pp. 852–862, Dec. 1965.
[10] G. K. Papakonstantinou, "A synthesis method for cutpoint cellular arrays," *IEEE Trans. Comput.*, vol. C-21, pp. 1286–1292, Dec. 1972.
[11] C. D. Weiss, "The characterization and properties of cascade realizable switching functions," *IEEE Trans. Comput.*, vol. C-18, pp. 625–633, July 1969.
[12] A. Mukhopadhyay, "Unate cellular logic," *IEEE Trans. Comput.*, vol. C-18, pp. 114–121, Feb. 1969.
[13] J. Sklansky, A. J. Korenjak, and H. S. Stone, "Canonical tributary networks," *IEEE Trans. Electron. Comput.*, vol. EC-14, pp. 961–963, Dec. 1965.
[14] M. A. Harrison, *Introduction to Switching and Automata Theory*. New York: McGraw-Hill, 1965.
[15] S. Muroga, T. Tsuboi, and C. R. Baugh, "Enumeration of threshold functions of eight variables," *IEEE Trans. Comput.*, vol. C-19, pp. 818–825, Sept. 1970.
[16] S. Muroga, *Threshold Logic and Its Applications*. New York: Wiley, 1971.
[17] J. P. Hayes, "The fanout structure of switching functions," *J. Ass. Comput. Mach.*, vol. 22, pp. 551–571, Oct. 1975.
[18] T. Sasao and K. Kinoshita, "On fanout-free functions and unate cascade functions" (in Japanese), in Tech. Papers *Group Electron. Comput., Inst. Electron. Commun. Eng. Japan*, Paper EC 77-7, May 1977.
[19] C. S. Lorens, "Invertible Boolean functions," *IEEE Trans. Electron. Comput.*, vol. EC-13, pp. 529–541, May 1964.
[20] K. Chakrabarti and O. Kolp, "Fan-in constrained tree networks of flexible cells," *IEEE Trans. Comput.*, vol. C-23, pp. 1238–1249, Dec. 1974.
[21] K. Kodandapani and S. Seth, "On combinatorial networks with restricted fanout," *IEEE Trans. Comput.*, vol. C-26, pp. 309–318, Apr. 1978.
[22] E. A. Bender and J. T. Butler, "Asymptotic approximations for the number of fanout-free functions," *IEEE Trans. Comput.*, vol. C-26, pp. 000–000, Dec. 1978.
[23] A. Cayley, "On the theory of the analytical forms called trees," *Phil. Mag.*, vol. XIII, pp. 172–176, 1857; also in *Collected Mathematical Papers*, vol. 3. London and New York: Cambridge, p. 246, 1887–1897.
[24] L. Frécon, "Capacité logique des structures en ligne et en peigne d'operateurs booleens a deux variables," *Revue Francaise d'Informatique et de Recherche Operationnelle*, no. B-2, pp. 62–85, 1971.

[4] f is n-p-equivalent to g if g can be obtained by permutation and/or negation of variables of f. The number of n-p-equivalence classes of fanout-free functions and unate cascade functions are $2 \cdot T_{ND}(n)$ and $2 \cdot U_{ND}(n)$, respectively ($n \geq 2$).

On Transposing Large $2^n \times 2^n$ Matrices

MORDECHAI BEN ARI

Abstract—**This correspondence presents two variations on the algorithm of Eklundh for transposing large $2^n \times 2^n$ matrices. The first variation shows how the number of accesses to secondary storage may be reduced at the expense of an increased amount of data transferred. Formulas for I/O time are derived from which we deduce the disk characteristics under which there is an improvement. The second variation shows that a small amount of additional secondary storage can be used to greatly improve the performance of the algorithm.**

Index Terms—**Digital image processing, externally stored matrices, large matrices, transportation algorithm, two-dimensional FFT.**

I. Introduction

The implementation of a two-dimensional FFT for digital image processing requires efficient implementation of a matrix transpose for large externally stored matrices. Eklundh [3] (and independently Floyd [4]) gave a description of an algorithm for transposing large matrices stored as a sequence of rows on a random access device using very little memory. Twogood and Ekstrom [6] improved the CPU time of the algorithm when more memory is available and in [1], [2], and [5] we find generalizations to nonsquare matrices.

This investigation is concerned with improving the I/O performance of the algorithm. While the discussion is for $2^n \times 2^n$

Manuscript received August 30, 1977; revised January 30, 1978.
The author is with the Division of Computer Sciences, Department of Mathematical Sciences, Tel Aviv University, Ramat-Aviv, Tel Aviv, Israel.