COMPUTER SYSTEMS LABORATORY

STANFORD ELECTRONICS LABORATORIES
DEPARTMENT Of ELECTRICAL ENGINEERING
STANFORD UNIVERSITY · STANFORD, CA 94305

DESIGN OF TWO-LEVEL FAULT-TOLERANT NETWORKS FROM THRESHOLD ELEMENTS

Evguenij A. Butakov*

and

Marat S. Posherstnick

Technical Report No. 134

March 1977

*This work was initiated in the Department of Computer Science of the Sebastopol Institute of Instrument Engineering, Sebastopol, U.S.S.R. It was completed by Professor Butakov while he was a Visiting Scholar in the Center for Reliable Computing, Digital Systems Laboratory, Stanford University, Stanford, California in 1977.

Center for Reliable Computing

DESIGN OF TWO-LEVEL FAULT-TOLERANT NETWORKS FROM THRESHOLD ELEMENTS

Evguenij A. Butakov*

and

Marat S. Posherstnick

Technical Report No. 134

March 1977

CENTER FOR RELIABLE COMPUTING
COMPUTER SYSTEMS LABORATORY
Departments of Electrical Engineering and Computer Science
Stanford University
Stanford, California

*This work was initiated in the Department of Computer Science of the Sebastopol Institute of Instrument Engineering, Sebastopol, U.S.S.R. It was completed by Professor Butakov while he was a Visiting Scholar in the Center for Reliable Computing, Digital Systems Laboratory, Stanford University, Stanford, California in 1977.

DESIGN OF TWO-LEVEL FAULT-TOLERANT NETWORKS FROM THRESHOLD ELEMENTS

Evguenij A. Butakov* and Marat S. Posherstnick

Technical Report No. 134

March 1977

Digital Systems Laboratory
Stanford University
Stanford California 94305

ABSTRACT

Only a small part of all Boolean functions of n-variables can be realized by one threshold element (T.E.). For all other functions the net must be built with at least two T.E.'s. The problem of constructing a fault-tolerant two-level network from T.E. is investigated. The notion of limiting function is introduced. It is shown that the use of these limiting functions induces a reduction in the number of possible candidates during the process of finding a realization of an arbitrary function by threshold functions. The method is based on the two-asummability property of threshold functions and therefore is applicable to completely specified Boolean functions with less than nine variables.

INDEX TERMS: complete monotonicity, failure-tolerant, limiting function, network synthesis, threshold element, threshold function, two-assummability, two-level net

*This work was initiated in the Department of Computer Science of the Sebastopol Institute of Instrument Engineering, Sebastopol, U.S.S.R. It was completed by Professor Butakov while he was a Visiting Scholar in the Digital Systems Laboratory, Stanford University, Stanford, CA 94305.

TABLE OF CONTENTS

FIGURES

I. INTRODUCTION

The fraction of switching functions of n variables which are realizable by one threshold element (T.E.) - threshold functions [1] - decreases rapidly with n: 14 out of a total of 2^4 Boolean functions of two variables are threshold functions, but only 3% of the four variable functions and only 2.4×10^{-27}% of the total number of seven variable switching functions are linearly separable (i.e. realizable by one T.E.). For all other functions, the net must be built with at least two T.E. Two-level nets of T.E. by virtue of their regularity and high switching speed are preferable, for technological reasons, to circuits of arbitrary configuration. Two kinds of two-level networks are shown in Figs. 1 and 2. If such a circuit is nonredundant, then the failure of the first-level element in general automatically implies errors at the output of the net. We shall suppose that the output element of the circuit is failure-free and all input signals are correct. Therefore, the net can be made fault-tolerant by adding some additional elements in the first level. The number of additional elements depends on the number of possible failures in the first stage. We shall say that the circuit is one-, two-, etc. fault-tolerant if its output signal is correct when one, two, etc. elements of the first level fail.

In this paper we shall discuss the methods of synthesis of fault-tolerant two-level circuits with the minimum number of threshold elements. We use the well known property of threshold functions discovered by Paul1 and McCluskey [2] and extensively studied and named *complete monotonicity* by Winder [3]. Our intention is to develop a

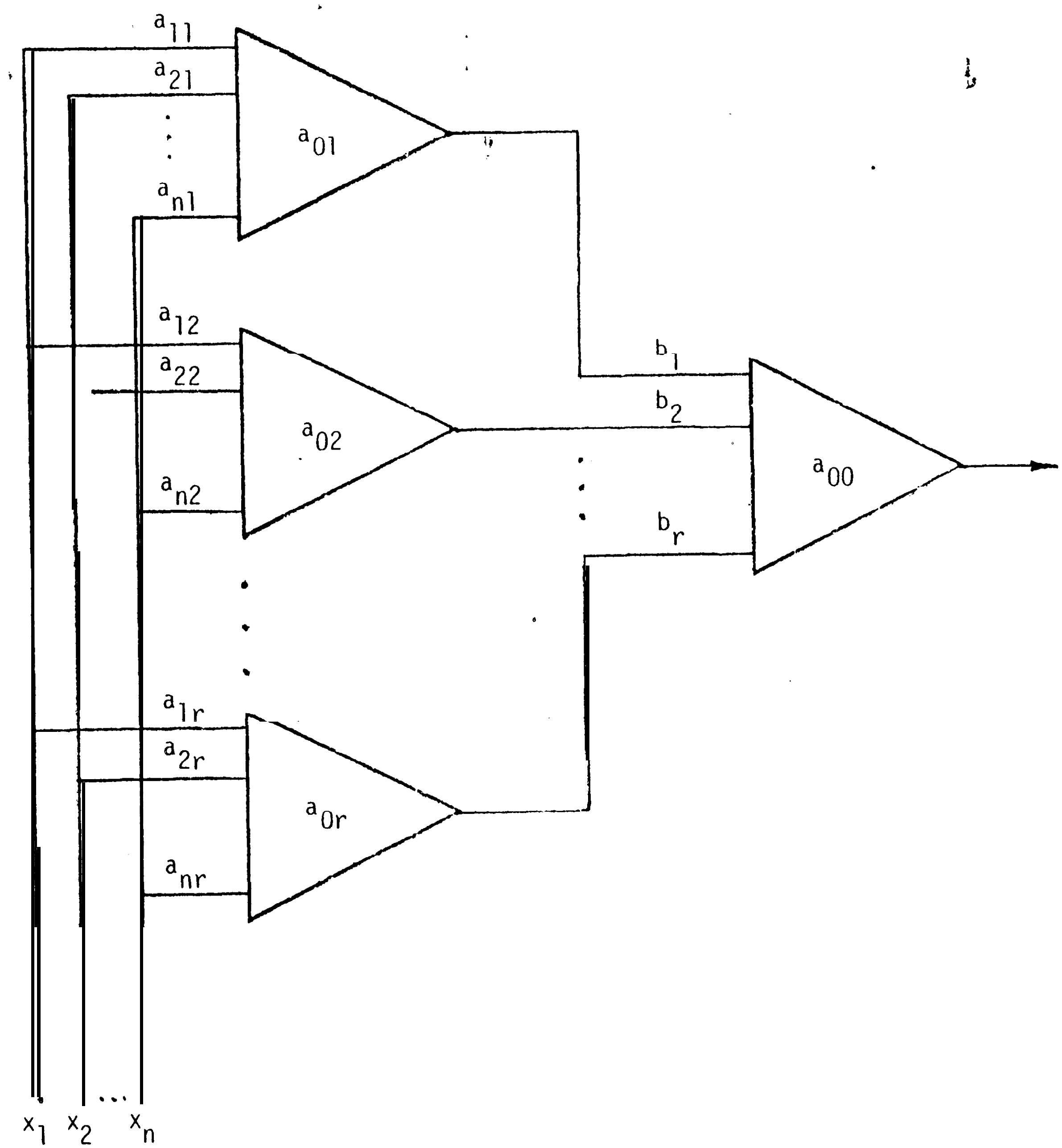


Fig. 1. Two-level circuit with input variables restricted to the first level.

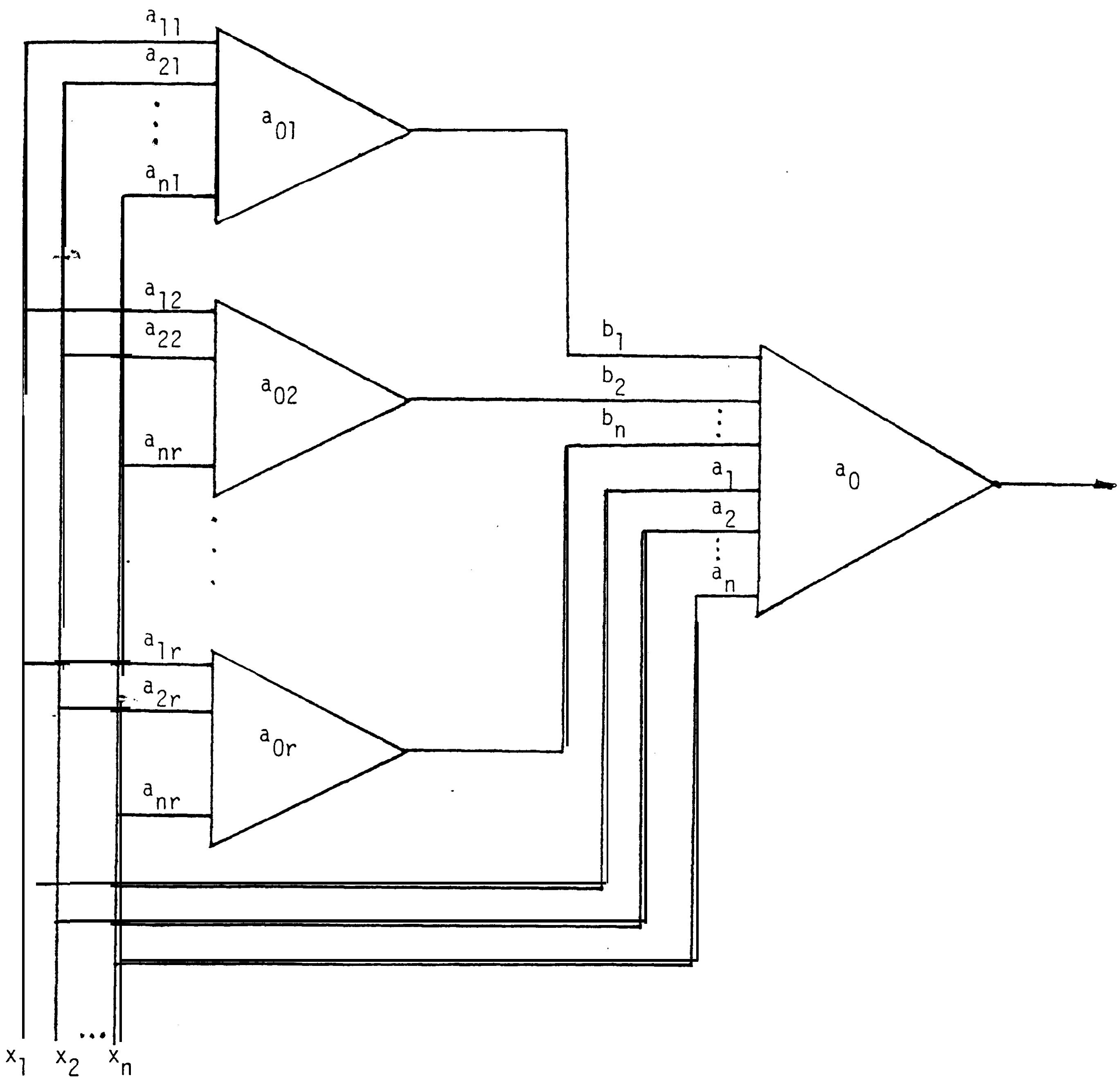


Fig. 2. Two level circuit with the input variables available at each level.

systematic approach to computer-aided synthesis of two-level networks from threshold elements, which can also be extended for synthesis of fault-tolerant circuits.

In section II we shall introduce the basic notations, formulate the problem and discuss some properties of fault-tolerant circuits. The notion of limiting functions and their properties are studied in section III. The algorithm for constructing limiting functions is also presented and illustrated. The next two sections, section IV and section V, give the methods of synthesis for two types of realization, Fig. 1 and Fig. 2.

II. BASIC NOTATIONS AND PROPERTIES

Let Q be the set {0,1} and let Q^n be the set of n bit wide vectors over Q. Boolean function of n-variables $F(x_1,x_2,\dots,x_n) = F(X)$ can be described as a mapping $F:Q^n \to Q$.

We shall assume that a Boolean function F(X) is given by the list of the sets:

$$M_1 = F^{-1}(1) = \{\vec{\alpha}_1,\vec{\alpha}_2,\dots,\vec{\alpha}_m\},\ \vec{\alpha}_i = (\alpha_{i1},\alpha_{i2},\dots,\alpha_{in}),\ (i = 1,\dots,m)$$

$$M_0 = F^{-1}(0) = \{\vec{\beta}_1,\vec{\beta}_2,\dots,\vec{\beta}_k\},\vec{\beta}_j = (\beta_{j1},\beta_{j2},\dots,\beta_{jn}),(j=1,\dots,k;\ m+k=2^n).$$

By definition, F(X) is a threshold function if and only if there exists a solution with respect to the unknowns $a_0,a_1,\dots,a_n$ of the system of linear inequalities:

$$\left.\begin{array}{l} a_1\alpha_{11}+a_2\alpha_{12}+\dots+a_n\alpha_{1n}\geq a_0 \\ \dots\dots\dots\dots\dots\dots \\ a_1\alpha_{m1}+a_2\alpha_{m2}+\dots+a_n\alpha_{mn}\geq a_0 \end{array}\right\}(1a)$$

$$\left.\begin{array}{l} a_1\beta_{11}+a_2\beta_{12}+\dots+a_n\beta_{1n}\leq a_0-1 \\ \dots\dots\dots\dots\dots\dots \\ a_1\beta_{k1}+a_2\beta_{k2} \text{[illegible]} \end{array}\right\}(1b)$$

(1a) and (1b) together: (1c)

Rearanging, we have,

$$\left.\begin{array}{l} -a_0+a_1\alpha_{11}+a_2\alpha_{12}+\dots+a_n\alpha_{1n}\geq 0 \\ \dots\dots\dots\dots\dots\dots \\ -a_0+a_1\alpha_{m1}+a_2\alpha_{m2}+\dots+a_n\alpha_{mn}\geq 0 \\ a_0-a_1\beta_{11}-a_2\beta_{12}-\dots-a_n\beta_{1n}\geq 1 \\ \dots\dots\dots\dots\dots\dots \\ a_0-a_1\beta_{k1}-a_2\beta_{k2}-\dots-a_n\beta_{kn}\geq 1 \end{array}\right\}(1d)$$

From the coefficient of system (1d) we form the 2" by (n+1)-matrix A, by defining the elements as follows:

$$a_{i1} = \begin{cases} -1 & \text{for } i=1,2,\dots,m \\ 1 & \text{for } i=m+1,\dots,m+k, \end{cases}$$

$$a_{ij} = \begin{cases} \alpha_{i(j-1)} & \text{for } i=1,\dots,m \\ -\beta_{i(j-1)} & \text{for } i=m+1,\dots,m+k=2^n, j=2,\dots,(n+1). \end{cases}$$

Furthermore, let $\vec{a} = (a_0, a_1, \dots, a_n)$ and $\vec{I} = (\underbrace{0,\dots,0}_{m}, \underbrace{1,\dots,1}_{k})^T$ (T as upper index means the transformation operation). In this notation system (1) may be written compactly in vector form [4,5]:

$$A \cdot \vec{a}^T \geq \vec{I} \qquad (2)$$

The solution $(a_0, a_1, \dots, a_n)$ of system (2) constitutes the realization of the given function and we shall write $F(X):[a_1, \dots, a_n, a_0]$. If system (2) is incompatible, then the original Boolean function is not realizable by a single threshold element. In this case we shall seek a realization of this function by a two-stage circuit as shown in Fig. 1.

<u>Definition</u>: The faults in two-level networks are defined as appearances of an incorrect signal at the output of one element of the first stage. We shall distinguish two kinds of faults:

a) a "1" appears at the output of an element instead of "0" (0/1 fault);

b) a "0" appears at the output of an element instead of "1" (1/0 fault).

If the network in Fig. 1 is to operate correctly even in the presence of faults, one needs to introduce some redundant elements into the circuit at the first level. The problem may be formulated in the following way:

It is neccessary to find a minimal number $r > 0$ and a mapping ψ of the vector of space $X = \{x_1, \dots, x_n\}$ onto vectors of space $Y = \{y_1, \dots, y_r\}$ (not necessarily one-to-one)

$$\begin{aligned} \psi(\vec{\alpha}_i) &= \vec{c}_i = (c_{i1}, c_{i2}, \dots, c_{ir}) \quad i = 1, 2, \dots, m \\ \psi(\vec{\beta}_j) &= \vec{c}_j = (c_{j1}, c_{j2}, \dots, c_{jr}) \quad j = m+1, \dots, k \end{aligned} \tag{3}$$

satisfying the following conditions:

1) the sequence of values $c_{1q}, c_{2q}, \dots, c_{(m+k)q} = \vec{c}^{\,q}$ (i.e. the column number q in the matrix C formed by $\vec{c}_1, \vec{c}_2, \dots, \vec{c}_{m+k}$) forms the threshold function y_q.
2) among the Boolean functions of r variables a threshold function

$$\phi(y_1, y_2, \dots, y_r) : [b_1, b_2, \dots, b_r ; a_{00}] \tag{4}$$

can be found such that $\phi(\vec{c}_i) = 1$, $\phi(\vec{c}_j) = 0$.
3) the net must be fault-tolerant with reference to 1/0 faults, or 0/1 faults, or both.

From the formal viewpoint the fault of the type (a) or (b) in one element means replacement of one of the rows of the matrix C by another row which differs from it by the value of one element. This, in turn, is equivalent to replacing one input array v of function ϕ by another - θ. If $\phi(\theta) = \phi(\nu)$, then the malfunction does not lead to an error at the output, since we assume that the output element is absolutely reliable.

Let μ be a set of different arrays in the sequence $\vec{c}_1, \dots, \vec{c}_m$, i.e. $\mu = \{\vec{\mu}_1, \dots, \vec{\mu}_t\}$ where $\vec{\mu}_1 = \vec{c}_{i_k}, \vec{\mu}_2 = \vec{c}_{i_1}, \dots, \vec{\mu}_t = \vec{c}_{i_v}$, let λ be a set of different arrays in the sequence $\vec{c}_{m+1}, \dots, \vec{c}_{m+k}$ in (3), $\lambda = \{\vec{\lambda}_1, \dots, \vec{\lambda}_s\}$. We use μ_i^d to denote the set of all arrays for which the Hamming distance from array μ_i $(i=1\ 2\ ,\ldots,t)$ is less or equal to d, let $\overset{d}{\mu} = \bigcup_{i=1}^{t} \{\mu_i^d\}$. Similarly one can introduce λ_j^d and λ^d $(j=1,2,\ldots,s)$.

Theorem 1: In order to construct a two-level network which is h-fault-tolerant with respect to only 0/1 faults or only 1/0 faults, it is necessary for the Hamming distance between any μ_i $(i = 1,2,\ldots,t)$ and any λ_j $(j = 1,2,\ldots,s)$ to be no less than $h+1$.

This theorem is analogous to Theorem 2 in [6] and we omit its proof.

The proofs of the next two theorems are simple and are obtained from the same reasoning as above.

Theorem 2: For the construction of the two-level h-fault-tolerant network from threshold elements with 1/0 faults it is necessary and sufficient for the Boolean function ϕ defined by the partitioning

$$\phi^{-1}(0) = M_0^\phi \supseteq \lambda \cup \lambda^h,\ \phi^{-1}(1) = M_1^\phi \supseteq \mu,\ M_0^\phi \quad M_1^\phi = \emptyset \qquad 5)$$

to be a threshold function.

Theorem 3: For the construction of a two-level h-fault-tolerant network from threshold elements with 0/1 faults it is necessary and sufficient for the Boolean function ϕ defined by the partitioning

$$\phi^{-1}(1) = M_1^\phi \supseteq \mu \cup \mu^h,\ M_0^\phi \supseteq \lambda,\ M_1^\phi \cap M_0^\phi = \emptyset \qquad (6)$$

to be a threshold function.

In general, we can have simultaneously both types of faults: 0-1 and 1-0. To tolerate this double fault, it is necessary to have a double Hamming distance between input arrays. From Theorems 1-3 we have the next statement.

Corollary: To construct a two-level net from threshold elements tolerant of both 0-1 and 1-0 faults in h elements it is necessary and sufficient if the Hamming distance between any $\vec{\mu}_i (i=1,2,\ldots,t)$ and any $\lambda_j (j=1,2,\ldots,s)$ is at least $2h+1$ and the function ϕ defined by partitioning

$$M_1^\phi \supseteq \mu \cup \mu^h, \; M_0^\phi \supseteq \lambda \cup \lambda^h, \; M_1^\phi \cap M_0^\phi = \emptyset \qquad (7)$$

is realizable by one threshold element.

III. LIMITING FUNCTIONS

The basic idea of the synthesis procedure is to construct matrix C with the minimal number of columns. There are several restrictions on this matrix, and the first of them is that every column of this matrix $\vec{c}^1, \vec{c}^2, \ldots, \vec{c}^r$ must be a threshold function $y_1, y_2, \ldots, y_r$ of $x_1, \ldots, x_n$.

Let F(X) be a given arbitrary Boolean function and let $F^{-1}(1) = \{\vec{\alpha}_1, \vec{\alpha}_2, \ldots, \vec{\alpha}_m\}$, $F^{-1}(0) = \{\vec{\beta}_1, \vec{\beta}_2, \ldots, \vec{\beta}_k\}$, $k+m = 2^n$.

Definition 1: We shall say that the function f(X) exceeds the function g(X) with respect to the function F(X)

$$f(X) \overset{F}{>} g(X),$$

if the following conditions are satisfied:

$$\begin{aligned} f(\vec{\alpha}_i) &\geq g(\vec{\alpha}_i), \ i \in \{1,2,\ldots,m\} \\ f(\vec{\beta}_j) &\leq g(\vec{\beta}_j), \ j \in \{1,2,\ldots,k\} \end{aligned} \tag{8}$$

and either there is a value i = s such that $f(\vec{\alpha}_s) > g(\vec{\alpha}_s)$ or there is a value $j = p$ for which we have $f(\vec{\beta}_p) < g(\vec{\beta}_p)$.

Definition 2: Call the threshold function f(X) limiting with respect to F(X) if there does not exist a threshold function h(X) such that

$$h(X) \overset{F}{>} f(X).$$

Theorem 4: Let mapping matrix C satisfy the conditions (1) and (2) (see Formula 4). Then all threshold functions $y_1, y_2, \ldots, y_r$ in C can be replaced by threshold functions $y_1', y_2', \ldots, y_r'$ exceeding them and the new matrix C^1 will also satisfy conditions (1) and (2).

Proof: It is obvious that condition (1) is satisfied for C . Condition (2) must be proved. We shall assume that the function ϕ in (4) is positive in $y_1, y_2, \ldots, y_r$ since this can always be arranged, taking the inverses of those variables in which ϕ is negative [1]. By definition

$$(b_1, b_2, \ldots, b_r) \cdot c_i^{\prime T} \geq a_{00} \quad (i = 1,2,\ldots,m) \tag{9}$$

$$(b_1, b_2, \ldots, b_r) \cdot c_j^{\prime T} \leq a_{00} - 1 \quad (j = 1,2,\ldots,k) \tag{10}$$

In the transition from the functions $y_1, y_2, \ldots, y_r$ to the functions $y_1' \overset{F}{\geq} y_1, \ldots, y_r' \overset{F}{\geq} y_r$, the vectors $\vec{c}_i, \vec{c}_j$ are replaced by $\vec{c}_i', \vec{c}_j'$ where $\vec{c}_i' \geq \vec{c}_i$ (i.e. the value of each component of vector $\vec{c}_i'$ is greater than or equal to the value of the corresponding component of vector $\vec{c}_i$), while $\vec{c}_j' \leq \vec{c}_j$. By virtue of the positiveness of the quantities $b_1, b_2, \ldots, b_r$ none of inequalities (9), (10) is violated. Consequently, matrix C', formed by rows $\vec{c}_i', \vec{c}_j'$, $i = 1,2,\ldots,m$, $j = 1,2,\ldots,k$ is a mapping satisfying conditions (1),(2). Q.E.D.

Corollary: In the synthesis of a two-level network, realizing the function F(X) every column of matrix C may be a limiting function with respect to F(X).

The number of limiting functions with respect to F(X) is substantially less than the number of threshold functions of n variables. Its mean value for four-variable functions is 37. It is easier to choose the functions $y_1, y_2, \ldots, y_r$ from the limiting functions of F than from all threshold functions. The major remaining problem is to find these limiting functions. The next theorem will be very useful for that purpose. Before we formulate it we can notice that according to Theorem 4.3.2.4 in book [1] there exists at least one equality in each of the two subsystems (la) and (lb). We call extreme points the input arrays which correspond to these equalities. It can be shown that if the system (la,lb) is consistent, then consistency will also hold for the system which is derived from it by transforming the extreme points from (la) into (lb) and vice versa. It means that for any threshold function f we can transform all extreme points from $f^{-1}(1)$ into $f^{-1}(0)$ and the new function, which we will call transformed function f*, will be the threshold. The next lemma establishes a stronger result.

Lemma: If g(X) is a threshold function, then the function of $g_1(X)$ received from g(X) by changing the meaning of exactly one extreme point is also a threshold function.

Proof: If g(X) is a threshold function, $g(X):[\vec{a},T]$, then according to the 2-assumability condition [1-5], which is necessary for a function to be threshold, it is impossible to find two pairs of vertices $\vec{\alpha}_i,\vec{\alpha}_j \in g^{-1}(1)$ and $\vec{\beta}_k,\vec{\beta}_l \in g^{-1}(0)$ such that $\vec{\alpha}_i+\vec{\alpha}_j=\vec{\beta}_k+\vec{\beta}_l$. Here the + sign means ordinary componentwise addition of vectors. Assume now that the function $g_1(X)$ obtained from g(X) by changing the value of the extreme point δ belonging to $g^{-1}(1)$ is not a threshold function. The only reason for its unrealizability by one threshold element is due to the change of value for $\vec{\delta}$. Therefore two points $\vec{\alpha}_1,\vec{\alpha}_2$ belonging to $g_1^{-1}(1)$ and $\vec{\beta}$ belonging to $g_1^{-1}(0)$ can be found such that $\vec{\alpha}_1+\vec{\alpha}_2 = \vec{\beta}+\vec{\delta}$. Let us multiply both sides of this equality by $\vec{a}$: $\vec{a}\cdot\vec{\alpha}_1+\vec{a}\cdot\vec{\alpha}_2 = \vec{a}\cdot\vec{\alpha}+\vec{a}\cdot\vec{\delta}$. Then from $\vec{a}\cdot\vec{\alpha}_1 \geq T$, $\vec{a}\cdot\vec{\alpha}_2 \geq T$ and $\vec{a}\cdot\vec{\beta} \leq T-1$ we obtain $2T \leq T-1+\vec{a}\cdot\vec{\delta}$ and $\vec{a}\cdot\vec{\delta} \geq T+1$. Hence $\vec{\delta}$ is not an extreme point of g. Similarly, a contradiction will be obtained if we assume that the extreme point $\vec{\delta}$ belongs to $g^{-1}(0)$. So $g_1(X)$ is also a threshold function. Q.E.D.

Theorem 5: A threshold function g(X) is limiting with respect to F(X) if, on the set of extreme points of g(X), both of the functions g(X) and F(X) have the same values.

Proof: Assume that there is a limiting function g(X) with respect to F(X) and that there are some extreme points, $\vec{\delta}_1, \vec{\delta}_2, \ldots, \vec{\delta}_r$ such that $F(\vec{\delta}_p) \neq g(\vec{\delta}_p)$ for $p = 1,2,\ldots,r$. Take one of these points, $\vec{\delta}_v$, and transform g(X) into $g_1(X)$ in the following way: if $F(\vec{\delta}_v) = 1$, then $g_1^{-1}(1) = g^{-1}(1) \cup \vec{\delta}_v$, $g_1^{-1}(0) = g^{-1}(0) \backslash \vec{\delta}_v$, if $F(\vec{\delta}_v) = 0$, then $g_1^{-1}(0) = g^{-1}(0) \cup \vec{\delta}_v$ and $g_1^{-1}(1) = g^{-1}(1) \backslash \vec{\delta}_v$. According to the previous lemma, $g_1(X)$ is a threshold function and it can easily be seen that $g_1(X) \overset{F}{>} g(X)$. Hence g(X) is not a limiting function. This is the contradiction, and so the theorem is proved. Q.E.D.

The algorithm for finding all limiting functions with respect to given F(X) may be represented in the following way:

1) From the table of threshold functions of n-variables sequentially pick up the extreme points of every threshold function.

2) Compare the values of F(X) with those of the threshold function on their extreme points.

3) If all values are coincident, then f(X) is limiting; if not, choose the next threshold function from the table.

Example: Table I gives all the limiting functions, $f_1, \ldots, f_{12}$, with respect to the parity function of three variables:

$$S(x_1, x_2, x_3) = x_1\bar{x}_2\bar{x}_3 \vee \bar{x}_1x_2\bar{x}_3 \vee \bar{x}_1\bar{x}_2x_3 \vee x_1x_2x_3 \qquad (11)$$

Note that this function has the greatest number of limiting functions among all functions of three variables.

The main advantage of this method is its programming simplicity. However, its drawbacks are severe, especially when the number of variables is large. It works well for functions up to six variables. For functions with more than six variables, more sophisticated methods not directly related to the tables of threshold functions must be invented.

TABLE I

x_3	x_2	x_1	S	f_1	f_2	f_3	f_4	f_5	f_6	f_7	f_8	f_9	f_{10}	f_{11}	f_{12}
0	0	1	1	0	0	0	1	0	1	1	1	1	1	1	1
0	1	0	1	0	0	1	0	1	0	1	1	1	1	1	1
1	1	1	1	0	1	0	0	1	1	0	1	1	1	1	1
1	0	0	1	1	0	0	0	1	1	1	0	1	1	1	1
0	0	0	0	0	0	0	0	0	0	1	0	0	1	1	1
0	1	1	0	0	0	0	0	0	0	0	1	1	0	1	1
1	0	1	0	0	0	0	0	0	1	0	0	1	1	0	1
1	1	0	0	0	0	0	0	1	0	0	0	1	1	1	0

IV. SYNTHESIS METHOD

A fault-tolerant realization with threshold gates will be found if we construct a matrix C that satisifes conditions (1) and (2) (below Formula (3))and Theorem 1.4. If we try the limiting functions for the columns of the matrix C, then the condition (1) will be satisfied. Therefore, the next problem is to find a minimal number of limiting functions (the columns of C) with the desired distance between its rows.

For every limiting function $f_1, f_2, \ldots$ with respect to a given F(X), we build mx k matrices, $W^1, W^2, \ldots,$ which we call difference matrices. The element w^{ℓ}_{ij} equals 1, if $f_{\ell}(\vec{\alpha}_i) \neq f_{\ell}(\vec{\beta}_j)$ for $i \in \{1,2,\ldots,m\}$, $j \in \{1,2,\ldots,k\}$.

Example: Table II represents all 12 difference matrices for the limiting functions in Table I.

Let Z_s be the mx k matrix for which each element is exactly s. The process of forming the matrix C with the desired distance d between its rows (from the limiting functions) is reduced to choosing a collection of difference matrices, $W^{\ell_1}, W^{\ell_2}, \ldots, W^{\ell_r}$ such that

$$W^{\ell_1} + W^{\ell_2} + \ldots + W^{\ell_r} \geq Z_s \qquad (12)$$

Here the + sign denotes component-wise addition of matrices, and two matrices $Z' = ||Z'_{ij}||$ and $Z'' = ||Z''_{ij}||$ are coupled by the relationship $Z' \geq Z''$, if $Z'_{ij} \geq Z''_{ij}$ for any $i \in \{1,2,\ldots,m\}$ and $j \in \{1,2,\ldots,k\}$. It can easily be shown that this process can be reduced to a problem of integer linear programming. However, it is more convenient to

TABLE II

$W^1 =$

	000	011	101	110
001	0	0	0	0
010	0	0	0	0
111	0	0	0	0
100	1	1	1	1

$W^2 =$

	000	011	101	110
001	0	0	0	0
010	0	0	0	0
111	1	1	1	1
100	0	0	0	0

$W^3 =$

	000	011	101	110
001	0	0	0	0
010	1	1	1	1
111	0	0	0	0
100	0	0	0	0

$W^4 =$

	000	011	101	110
001	1	1	1	1
010	0	0	0	0
111	0	0	0	0
100	0	0	0	0

$W^5 =$

	000	011	101	110
001	0	0	0	1
010	1	1	1	0
111	1	1	1	0
100	1	1	1	0

$W^6 =$

	000	011	101	110
001	1	1	0	1
010	0	0	1	0
111	1	1	0	1
100	1	1	0	1

$W^7 =$

	000	011	101	110
001	0	1	1	1
010	0	1	1	1
111	1	0	0	0
100	0	1	1	1

$W^8 =$

	000	011	101	110
001	1	0	1	1
010	1	0	1	1
111	1	0	1	1
100	0	1	0	0

$W^9 =$

	000	011	101	110
001	1	0	0	0
010	1	0	0	0
111	1	0	0	0
100	1	0	0	0

$W^{10} =$

	000	011	101	110
001	0	1	0	0
010	0	1	0	0
111	0	1	0	0
100	0	1	0	0

$W^{11} =$

	000	011	101	110
001	0	0	1	0
010	0	0	1	0
111	0	0	1	0
100	0	0	1	0

$W^{12} =$

	000	011	101	110
001	0	0	0	1
010	0	0	0	1
111	0	0	0	1
100	0	0	0	1

solve it by sorting in a manner similar to what is used in searching for coverings [7].

Now we are able to give the main steps of the method.

1) For given function F(X), find all limiting functions and form the difference matrix for each of them.

2) Minimizing the number of columns and, using relation (12), form the matrix C with the desired distance between μ_i and λ_j, for $i \in \{1,2,...,m\}$ and $j \in \{1,2,...,k\}$.

3) Depending on the fault, form the partitioning (5),(6),(7).

4) If the function ϕ defined by this partitioning is a threshold function, the problem has been solved; otherwise go to step 2 and form the next matrix C.

Example: Let us find a network for the function (11), of the form shown in Fig. 1, which is tolerant to O/l fault in one element. We shall proceed according to the steps as defined above.

1) All limiting functions are displayed in Table I, all difference matrices are represented in Table II.

2) According to Theorem 1, the distance between μ_i and λ_j in C must be no less than 2. The set of limiting functions f_5, f_6, f_7, f_8 is the minimal one out of all the sets which satisfy condition (12) when s= 2. Let us form the matrix C from these functions (l-arrays are above the dotted line).

$$C = \left|\begin{array}{cccc} 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 \\ \hline 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{array}\right|$$

3),4) In Fig. 3a a Weich diagram is displayed for the function ϕ, in which $y_1 = f_5$, $y_2 = f_6$, $y_3 = f_7$, $y_4 = f_8$. Applying the methods for realization of incompletely specified functions by one threshold element [1,3,8] for the function ϕ complemented in accordance with (5), we obtain the threshold function $\phi'(y_1,y_2,y_3,y_4)$ (Fig. 3b) which defines the output element for a network tolerant to 0\1 faults of one element. An analogous complementation inaccordance with (6) yields the threshold function $\phi''(y_1,y_2,y_3,y_4)$ (Fig. 3c), which defines the output element for a network tolerant to 1\0 faults of one element (Fig. 4). Note that the fault-tolerant realization for this function has only one more element than the intolerant realization.

For a more complicated example, we shall consider the problem of constructing a one-digit combinational binary adder, which is stable relative to malfunctions of the 1\0 and 0\1 types in any element except the output elements. As it is well known, this problem can be reduced to finding an implementation for the following two functions: function (11) and function

$$\sigma(x_1,x_2,x_3) = x_1x_2\bar{x}_3 \vee x_1\bar{x}_2x_3 \vee \bar{x}_1x_2x_3 \qquad (13)$$

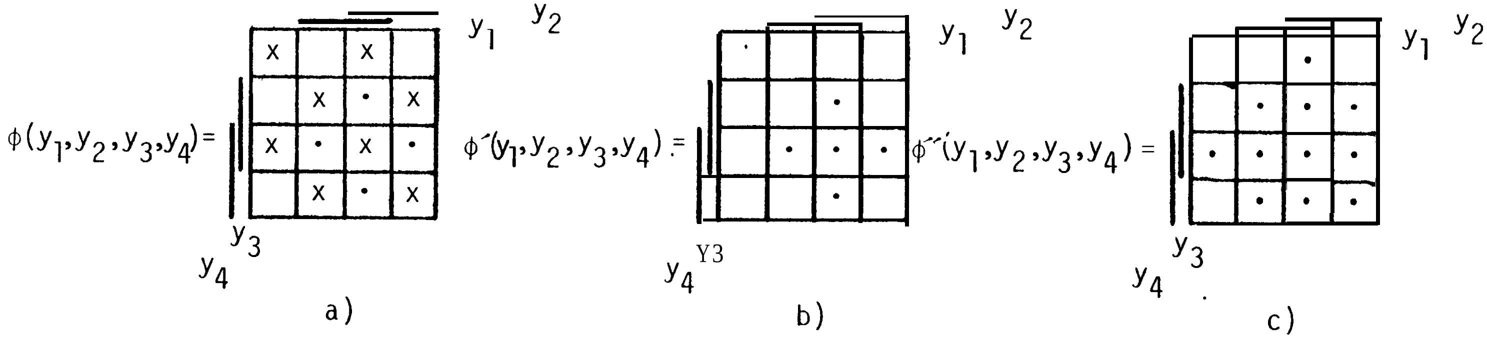
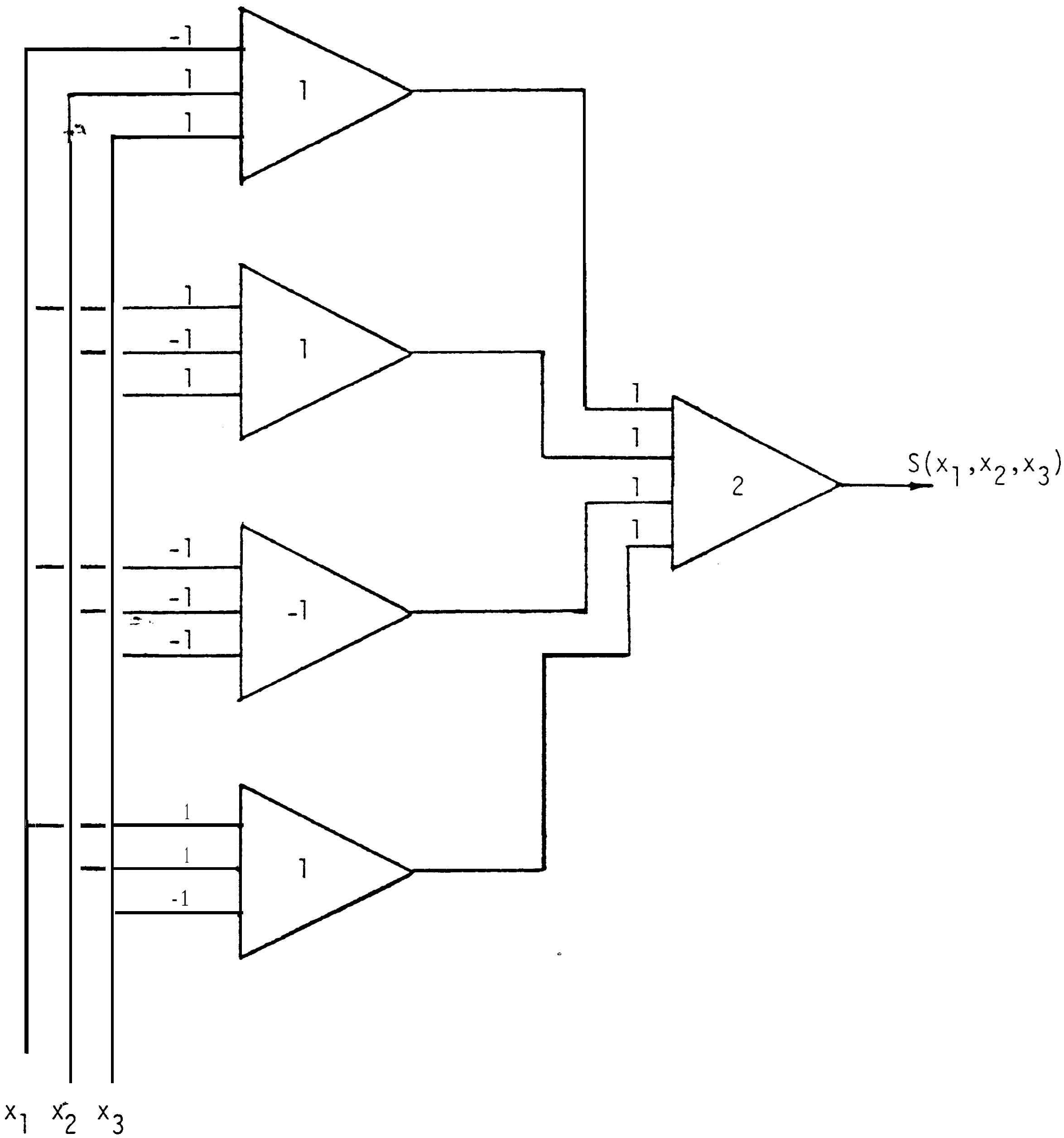


Fig. 3. Output functions (Don't care points marked with x's).


-1
1
1
1
1
-1
1
1
-1
-1
-1
-1
1
1
-1
1
1
1
1
1
2
S(x1,x2,x3)
x1
x2
x3

The last function is a threshold function and at the same time coincides with the inverse of the function f_7 in Table I. Therefore, it is expedient to seek the set of difference matrices which include W^7. It is not difficult to check that the set of limiting functions $f_2, f_5, f_6, \overline{f}_7, \overline{f}_7, f_8, f_9$ satisfies the condition (12) with $s = 3$. The realization of a 1-fault-tolerant adder is displayed in Fig. 5.

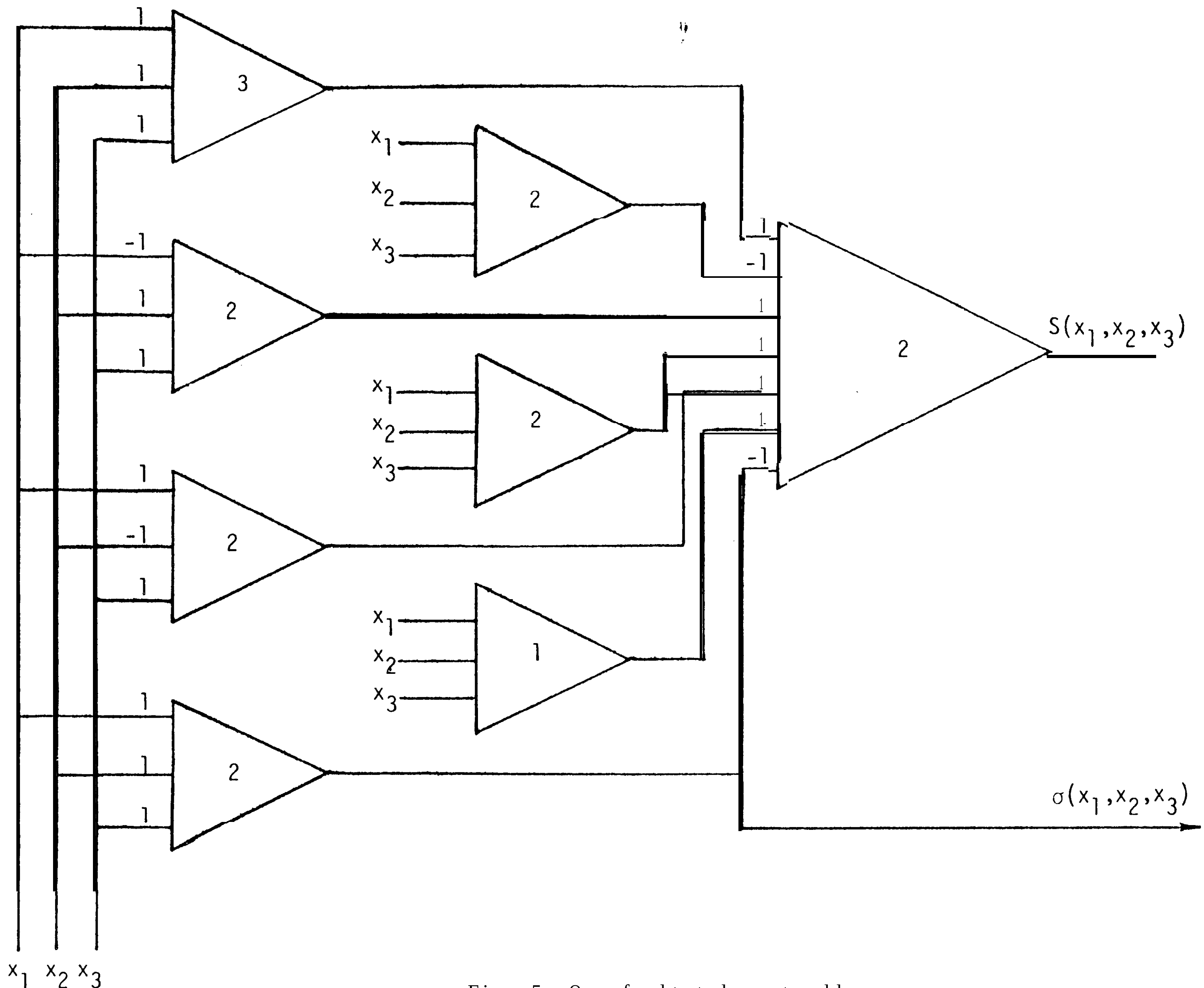


Fig. 5. One fault-tolerant adder.

V. GENERAL CASE

In Fig. 2, the general case of two-level network realization is shown (with the primary input variables $x_1, x_2, \ldots, x_n$, also available as inputs to the output element of the circuit). Let us introduce and denote by J the square matrix of order m+ k in which the diagonal elements of the first m rows are equal to 1, while the diagonal* elements of the next k rows are equal to -1. The remaining elements are equal to zero. For finding a realization of the given function F(X), by a circuit shown in Fig. 2, the problem is formulated in the following manner.

To obtain the minimal number of elements in the circuit, it is necessary to add to the matrix A a minimal number of columns $J\vec{c}^1, J\vec{c}^2, \ldots, J\vec{c}^r$, where every column $\vec{c}^1, \vec{c}^2, \ldots, \vec{c}^r$ is a threshold function $y_1, y_2, \ldots, y_r$ of the input variables, so that the system of inequalities

$$\hat{A} \cdot \hat{a}^T \geq I \qquad (14)$$

is compatible. Here $\hat{A} = [A, J\vec{c}^1, J\vec{c}^2, \ldots, J\vec{c}^2]$, $\hat{a} = (\vec{a}_0, \vec{a}_1, \ldots, \vec{a}_n \vec{b}_1, \ldots, \vec{b}_r)$

In other words, we find a representation of F(X) in the form

$$F(X) = \phi(x_1, \ldots, x_n, y_1, \ldots, y_r),$$

where ϕ and $y_1, y_2, \ldots, y_r$ are threshold functions. It can be easily shown [1] that the number of circuit elements does not change if the function ϕ is assumed to be positive in $y_1, \ldots, y_r$, i.e. in both cases $b_1 > 0$, $b_2 > 0$, $\ldots b_r > 0$.

It is well known [1-5] that the condition of 2-asummability is necessary for a function to be realizable by a threshold element. This is also sufficient when the number of variables is less that 9. If a completely determined nonthreshold Boolean function is given, then there always exist at least four vertices for which the condition

$$\vec{\alpha}_i + \vec{\alpha}_j = \vec{\beta}_\ell + \vec{\beta}_s, i,j \in \{1,2,\ldots,m\}, \ell,s \in \{1,2,\ldots,k\} \qquad (15)$$

is satisfied. Such vertices are called summable in contradiction to the asummable vertices, i.e. the 1-vertices for which condition (15) is not satisfied. It is clear that the summability of vertices $\vec{\alpha}_i, \vec{\alpha}_j$ may be eliminated by the addition of a $(n+1)^{th}$ coordinate to the vectors $\vec{\alpha}_i, \vec{\alpha}_j, \vec{\beta}_\ell, \vec{\beta}_s$ in such a way that the sum of the $(n+1)^{th}$ components of the vectors $\vec{\alpha}_i$ and $\vec{\alpha}_j$ is differentfromthe sumofthe $(n+1)^{th}$ components of the vectors $\vec{\beta}_\ell$ and $\vec{\beta}_s$. The basic idea of the synthesis consists oftheaddition of a minimal number, say r, of supplementary coordinates (columns $\vec{c}^1, \vec{c}^2, \ldots, \vec{c}^r$ to matrix A). We obtain $(n+r)$ - dimensional vectors for which no two vectors are summable. Accordingly, a summability graph is constructed for which the nodes are associated with the 1-vertices $\vec{\alpha}_1, \ldots, \vec{\alpha}_m$, and the pair of nodes $\vec{\alpha}_i, \vec{\alpha}_j$ are joined by an edge if the condition (15) is satisfied.

Example: Figure 6 contains the summability graph for the function (11). The incidence matrix corresponding to this graph has the following form (only half of the symmetric matrix is shown).

	001	010	171	100
001		1	1	1
010			1	1
111				1
100				

By adding a column to matrix A, we eliminate some summable pairs and the corresponding edges in the summability graph. It is easy to determine which edges in the summability graph are eliminated by the

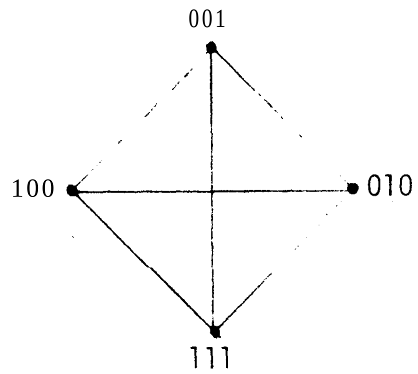


Fig. 6. The summability graph for the function (11).

given limiting function f_ℓ. For this, it is sufficient to construct as an analogy to the difference matrix, the binary matrix V^ℓ in which the element $v^\ell_{ij} = 1$ if and only if the vertices formed from $\vec{\alpha}_i, \vec{\alpha}_j$ by adding to matrix A the column $\vec{c}^\ell$ are asummable, i.e. $f_\ell(\vec{\alpha}_i) + f_\ell(\vec{\alpha}_j) \neq f_\ell(\vec{\beta}) + f_\ell(\vec{\beta}_s)$.

In Table III the matrices of V^ℓ are presented for all the limiting functions in Table I. To find the solution of the problem, it is necessary to find the minimal number of matrices V^ℓ covering all the one elements of matrix U. This problem is similar to the Quine covering problem and can be reduced to it. Not every coverage of matrix U will give a solution to the problem. It is necessary that the function $\phi(x_1,\ldots,x_n,y_1,\ldots,y_r)$ be a threshold function. Since the function ϕ is incompletely specified, the condition of 2-asummability is not sufficient even when the number of variables is very small. Therefore it is necessary to verify that the function ϕ is threshold. If it is, then the problem is solved; otherwise once needs to find another covering.

Example: Any of the matrices V^5, V^6, V^7, V^8 gives a coverage of the incidence matrix U. Thus a single additional column to the matrix A is sufficient. Figure 7 gives a realization of the function (11) with the limiting function f_7. It can be verified that functions f_5, f_6, f_8 also give a realization.

Let us assume now that all input leads are fail-free and that only the elements of the first-stage can fail. There is no need to calculate the distances between input arrays of the output elements. The difference between them is provided by input variables. Theorems similar to Theorems 2 and 3 and a similar corollary can be formulated for this case

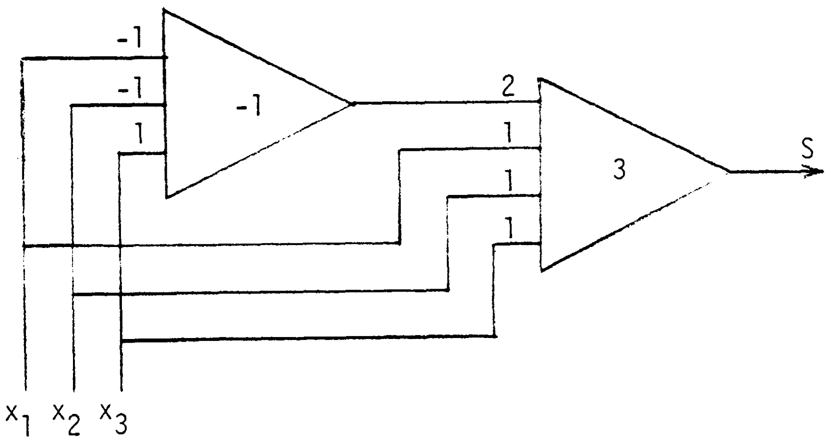


Fig. 7. Realization of the function (11) with the limiting function f_7.

TABLE III

V^1 =

	001	010	111	100
001		0	0	1
010			0	1
111				1
100				

V^2 =

	001	010	111	100
001		0	1	0
010			1	0
111				1
100				

V^3 =

	001	010	111	100
001		1	0	0
010			1	1
111				0
100				

V^4 =

	001	010	111	100
001		1	1	1
010			0	0
111				0
100				

V^5 =

	001	010	111	100
001		1	1	1
010			1	1
111				1
100				

V^6 =

	001	010	111	100
001		1	1	1
010			1	1
111				1
100				

V^7 =

	001	010	111	100
001		1	1	1
010			1	1
111				1
100				

V^8 =

	001	010	111	100
001		1	1	1
010			1	1
111				1
100				

V^9 =

	001	010	111	100
001		1	0	1
010			0	1
111				0
100				

V^{10} =

	001	010	111	100
001		1	1	0
010			1	0
111				0
100				

V^{11} =

	001	010	111	100
001		0	1	1
010			0	0
111				1
100				

V^{12} =

	001	010	111	100
001		0	0	0
010			1	1
111				1
100				

TABLE III

V^1 =

	001	010	111	100
001		0	0	1
010			0	1
111				1
100				

V^2 =

	001	010	111	100
001		0	1	0
010			1	0
111				1
100				

V^3 =

	001	010	111	100
001		1	0	0
010			1	1
111				0
100				

V^4 =

	001	010	111	100
001		1	1	1
010			0	0
111				0
100				

V^5 =

	001	010	111	100
001		1	1	1
010			1	1
111				1
100				

V^6 =

	001	010	111	100
001		1	1	1
010			1	1
111				1
100				

V^7 =

	001	010	111	100
001		1	1	1
010			1	1
111				1
100				

V^8 =

	001	010	111	100
001		1	1	1
010			1	1
111				1
100				

V^9 =

	001	010	111	100
001		1	0	1
010			0	1
111				0
100				

V^{10} =

	001	010	111	100
001		1	1	0
010			1	0
111				0
100				

V^{11} =

	001	010	111	100
001		0	1	1
010			0	0
111				1
100				

V^{12} =

	001	010	111	100
001		0	0	0
010			1	1
111				1
100				

also, but we will omit them here. When the realization of a given function is found, we add successively 1,2,... external limiting functions and we construct a partitioning similar to (5), (6), or (7). If
3 the respective Boolean function is threshold, then the problem is solved; otherwise, one needs to look for some other limiting functions.

Example: Figure 8 shows one-fault-tolerant realization of the function (11).

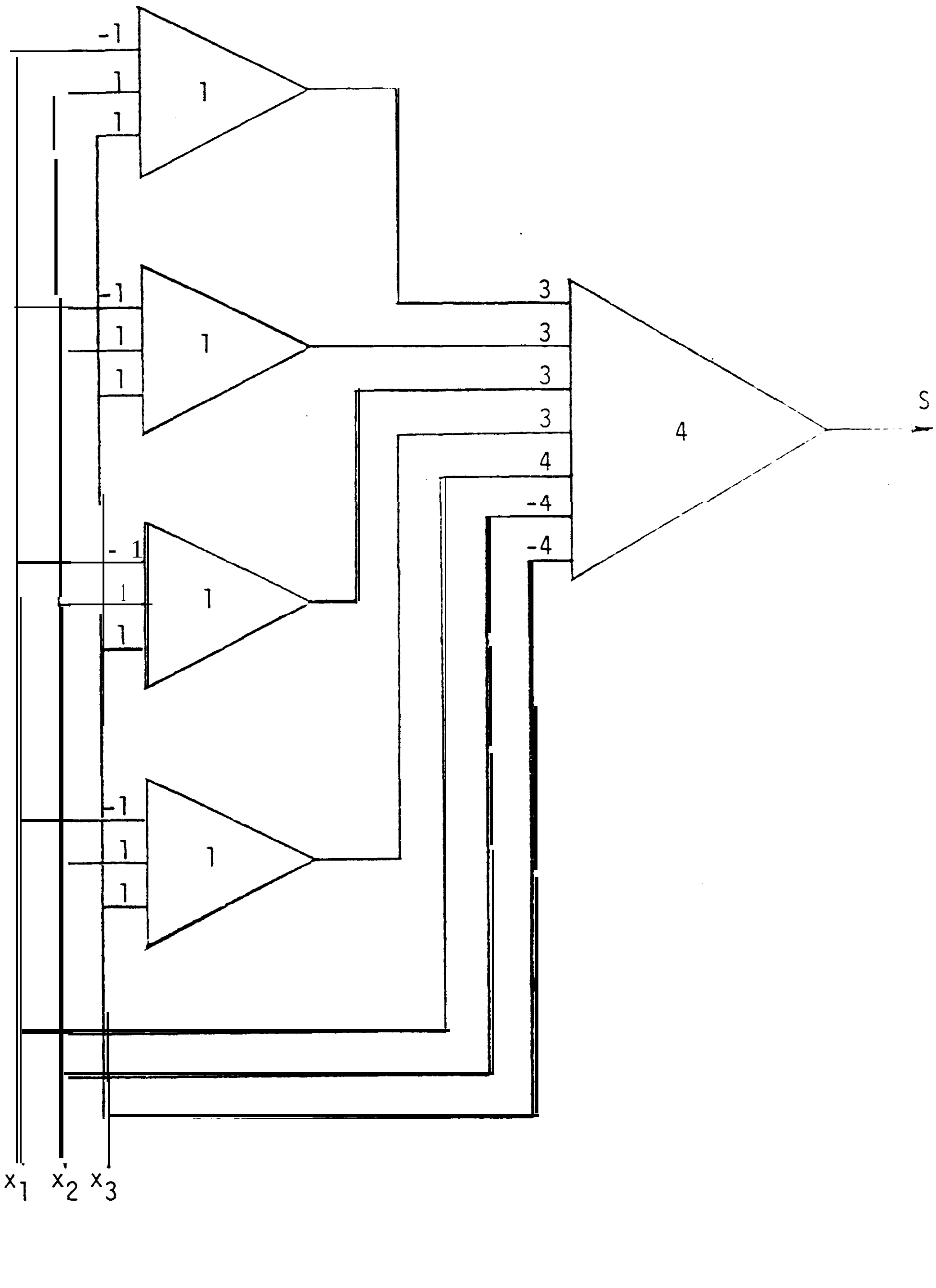


Fig. 8. A one-fault-tolerant realization of the function (11).

Acknowledgements

The authors wish to thank Professor Edward J. McCluskey, Director of the Digital Systems Laboratory of Stanford University, for his suggestions and stimulating discussions, and Dr. Jacques Losq, Research Associate in the Digital Systems Laboratory, for the careful revision and correction of the manuscript. Also many thanks to Mrs. Naomi Schulman for her valuable help in typing this paper.

Evguenij A. Butakov was born in Pzokopjevsk, USSR, on June 13, 1936. He received the M.Sc. degree in computer science from Tomsk University, USSR in 1960, candidate degree in engineering from the same university in 1965, and Doctor's degree from Lenigrad Institute of Electrical Engineering in 1974. Since 1966 he has been a professor and the chairman of the Computer Science Department at Sebastopol Institute of Instrument Engineering, Sebastopol, USSR. His current research interests are switching theory, logic design, computer systems, and pattern recognition. This work was initiated in the Swbastopol Institute of Instrument Engineering and completed while Professor Butakov was a visiting scholar at the Digital Systems Laboratory at Stanford University, Stanford, California.

Marat S. Poshertnick was born in 1947 in Simpherol, USSR. He received the engineering degree in computer science in 1968 from Sebastopol Institute of Instrument Engineering, Sebastopol, USSR. In 1969 he joined the staff of the Computer Science Department at Sebastopol Institute of Instrument Engineering. His current interests include switching theory, logical design, and programming.

REFERENCES

[1] S. Muroga, Threshold Logic and Its Applications. New York:Wiley, 1971.

[2] M. C. Paull and E. J. McCluskey, "Boolean Functions Realizable with Single Threshold Devices," Proc. IRE, vol. 48, no. 7, pp. 1335-1337, July 1960.

[3] R. O. Winder, "Threshold Logic," Ph.D. thesis, Math. Dep., Princeton Univ., Princeton, N.J., May 1962.

[4] J.E. Hopcroft and R.. L. Mattson, "Synthesis of Minimal Threshold Logic Networks," IEEE Trans. Electron. Comput., vol. EC-14, pp. 552-560, Aug. 1965.

[5] E. B. Stabler, "Threshold Gate Network Synthesis ,"Proc. IEEE Symp. Switching Circuit Theory and Logical Design, pp. 5-11, Oct. 1965.

[6] D. K. Pradhan and S. M. Reddy, "Design of Two-Level Fault-Tolerant Networks," IEEE Trans. Comput., vol. C-23, pp. 41-47, Jan. 1974.

[7] A. D. Zakrevskii, "Optimization of Set Coverages," in: A Programing Language for Logic and Coding Algorithm, M. A. Gavrilov and A. D. Zakrevskii, Eds., New York:Academic Press, 1965.

[8] E. A. Butakov, Methods for the Synthesis of Switching Devices from Treshold Elements (in Russian), Energiya:Moscow. 1970.