The logic of Boolean matrices

C. R. Edwards

School of Electrical Engineering, University of Bath, Claverton Down, Bath BA2 7AY

A Boolean matrix algebra is described which enables many logical functions to be manipulated simultaneously. The algebra is similar to conventional matrix algebra and its structure includes a topological description of logic circuits. The matrices are readily manipulated by the digital computer.

(Received December 1971)

The type of Boolean matrices considered here were first developed by J. O. Campeau in the late 1950s. To the author's knowledge little has been published in this field recently but we may look forward to a reappraisal of this topic as it enables logic analysis and synthesis to be carried out with relative ease, especially with the aid of a digital computer.

The manipulation of Boolean matrices is similar to that of conventional matrices so that logic design engineers will have little difficulty in appreciating its advantages over more usual methods and employing it as a useful tool.

It is not the purpose of this article to submit a series of rigorous mathematical proofs, for these the list of references should be consulted, but to give the reader a good working knowledge of this rather fascinating branch of applied mathematics.

1. Basic concepts

We require that Boolean functions may be expressed in a matrix algebra similar to that of conventional matrix algebra, viz.

$$\begin{bmatrix}\text{COEFFICIENTS}\\ \text{(Defining the}\\ \text{function(s))}\end{bmatrix}\begin{bmatrix}\text{INPUT}\\ \text{VARIABLES}\end{bmatrix}=\begin{bmatrix}\text{REQUIRED}\\ \text{FUNCTION(S)}\end{bmatrix}$$

It will be recalled that in 'conventional' matrix algebra the coefficients are arranged in a particular order so that under multiplication the correct coefficient is associated with a particular variable, e.g.

$$\begin{bmatrix}3 & 2\\ -4 & 1\end{bmatrix}\begin{bmatrix}x\\ y\end{bmatrix}=\begin{bmatrix}P\\ Q\end{bmatrix}$$

gives

$$3x + 2y = P$$
$$-4x + y = Q$$

This is also true of Boolean matrix algebra, the coefficients being those, in order, of the chosen 'canonical' form. Since there are many different canonical forms, each of which may be used to describe a particular function, it is necessary to 'standardise' this form and thereafter not deviate from it. The choice of the canonical form is made, not as it may appear for aesthetic reasons, but to enable a straightforward 'unit' matrix to be developed later.

An equation of canonic form as used here consists of 2^n terms, where n is the number of input variables. Each term contains all the variables, complemented or uncomplemented, and a constant (0 or 1); no two terms are the same. By choosing particular values for the constants, in (1) below for example, the reader may quickly verify that any logical function of x_1 and x_2 may be defined.

The two-variable canonical form is:

$$f(x_1, x_2) = (c_1.\bar{x}_1.\bar{x}_2 + c_2.\bar{x}_1.x_2 + c_3.x_1.\bar{x}_2 + c_4.x_1.x_2)\,, \quad (1)$$

i.e. 'Sum of products' form.

Note that if the variables x_1, x_2 are given the logical value '1', the digits following each coefficient 'increase' by '1' as the coefficient number increases, i.e.

$$f(1, 1) = (c_1.0.0 + c_2.0.1 + c_3.1.0 + c_4.1.1)\,.$$

A single function in Boolean matrix form is written as:

$$[c_1\; c_2\; c_3\; c_4]\begin{bmatrix}x_1\\ x_2\end{bmatrix} = [T]$$

For example, if $T = \bar{x}_2 = \bar{x}_1.\bar{x}_2 + x_1.\bar{x}_2$, then in canonical form:

$$T = (1.\bar{x}_1.\bar{x}_2 + 0.\bar{x}_1.x_2 + 1.x_1.\bar{x}_2 + 0.x_1.x_2)$$

giving

$$[1\;\;0\;\;1\;\;0]\begin{bmatrix}x_1\\ x_2\end{bmatrix} = [T]$$

But there is no reason why we should not write any number of functions in this way:

Given

$$T = x \oplus y = x.\bar{y} + \bar{x}.y\,,$$
$$U = x.y\,,$$
$$V = \bar{x} + \bar{y}\,,$$

then from

$$\begin{bmatrix}c_{1T} & c_{2T} & c_{3T} & c_{4T}\\ c_{1U} & c_{2U} & c_{3U} & c_{4U}\\ c_{1V} & c_{2V} & c_{3V} & c_{4V}\end{bmatrix}\begin{bmatrix}x\\ y\end{bmatrix} = \begin{bmatrix}T\\ U\\ V\end{bmatrix},$$

we have

$$\begin{bmatrix}0 & 1 & 1 & 0\\ 0 & 0 & 0 & 1\\ 1 & 1 & 1 & 0\end{bmatrix}\begin{bmatrix}x\\ y\end{bmatrix} = \begin{bmatrix}T\\ U\\ V\end{bmatrix}.$$

as the reader may verify.

Three points to note here. Firstly, the number of columns appearing in the 'coefficient' part of the matrix equation is 2^n where 'n' is the number of variables. Secondly, the number of rows appearing in the 'coefficient' matrix is equal to the number of functions. Thirdly, the chosen canonical form, and order of terms must be rigidly adhered to, otherwise incorrect results occur.

The next problem to be overcome is the solution of a matrix equation of the form following. Suppose that we are given:

If

$$\begin{bmatrix}1 & 0 & 0 & 1\\ 0 & 1 & 0 & 0\end{bmatrix}\begin{bmatrix}1\\ 1\end{bmatrix}=\begin{bmatrix}S\\ T\end{bmatrix},$$

then what are the values of S and T? This is, in effect, the 'multiplication' of a vector by a two-dimensional matrix. In practical terms we require the solution to the equations

$$S = \overline{x \oplus y}\,,$$

and

$$T = \bar{x}.y$$

when

$$x = 1$$
$$y = 1$$

One method of doing this is to put $x = 1, y = 1$ in the canonical equation:

$$f(x, y) = (c_1.\bar{x}.\bar{y} + c_2.\bar{x}.y + c_3.x.\bar{y} + c_4.x.y)$$
$$f_S(1, 1) = (c_1.0.0 + c_2.0.1 + c_3.1.0 + c_4.1.1)$$

and entering the coefficients for function 'S':

$$S = f_S(1, 1) = (1.0.0 + 0.0.1 + 0.1.0 + 1.1.1) = 1$$

similarly

$$T = (0.0.0 + 1.0.1 + 0.1.0 + 0.1.1) = 0$$

Then combining the two we write:

$$\begin{bmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \end{bmatrix} = \begin{bmatrix} S \\ T \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$$

This is clearly rather tedious and we require a faster method and one which is similar to 'conventional' matrix multiplication.

Since the equations are in canonical form, the values of the variables (x and y say) can only 'select' one coefficient column from the coefficient matrix as the answer. In the example above, c_4 was 'selected' by the values $x = 1$, $y = 1$ and the answer was given by column 4 of the coefficient matrix. If in the above example we put $x = 0$, $y = 0$, the required solution lies in column 1 of the coefficient matrix; if $x = 1$, $y = 0$ column 3 gives the solution, etc. Therefore, if we write below the coefficient matrix the values of the input variables so that the answer is directly given by the coefficients above them, the 'multiplication' can be performed by inspection, e.g.

$$\begin{bmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \end{bmatrix} = \begin{bmatrix} S \\ T \end{bmatrix}$$

$$\begin{bmatrix} 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 \end{bmatrix} \leftarrow \text{(additional identifying matrix)}$$

Whence:

$$\begin{bmatrix} 1 \\ 0 \end{bmatrix} = \begin{bmatrix} S \\ T \end{bmatrix}$$

Moreover, and perhaps surprisingly, we have 'solved' for two Boolean equations simultaneously.

The matrix which we write below the coefficient matrix is termed the 'unit' or 'identity' matrix. It has the same properties as the unit matrix in conventional matrix algebra, i.e. the multiplication of any other vector (or matrix) by the unit matrix leaves that vector (or matrix) unchanged. It is also a re-definition of the canonical form discussed above.

For convenience we will define the unit matrix $[A]_{\text{UNIT}}$ as being a binary progession from left to right and then we can always derive the respective canonical form from it, e.g. for two input variables,

$$[A]_{\text{UNIT}} = \begin{bmatrix} 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 \end{bmatrix}$$

In 'shorthand' we may 'decode' each column of the unit matrix to give a 'decimal' form, namely:

$$[A']_{\text{UNIT}} = [0 \quad 1 \quad 2 \quad 3]$$

Then for three variables we can write the unit matrix directly as

$$[A']_{\text{UNIT}} = [0 \quad 1 \quad 2 \quad 3 \quad 4 \quad 5 \quad 6 \quad 7],$$

or

$$[A]_{\text{UNIT}} = \begin{bmatrix} 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \end{bmatrix}$$

If the variables are say

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix},$$

then the canonical form is

$$f(x, y, z) = (c_1.\bar{x}.\bar{y}.\bar{z} + c_2.\bar{x}.\bar{y}.z + c_3.\bar{x}.y.\bar{z} + c_4.\bar{x}.y.z + c_5.x.\bar{y}.\bar{z} + c_6.x.\bar{y}.z + c_7.x.y.\bar{z} + c_8.x.y.z)$$

Example:
Evaluate

$$S = x + y$$
$$T = x.y.z + \bar{x}.y.\bar{z}$$
$$U = \bar{x}$$

simultaneously for $x = 0$, $y = 1$, $z = 1$

Writing the truth table in canonical form:

x	y	z	S	T	U
0	0	0	0	0	1
0	0	1	0	0	1
0	1	0	1	1	1
0	1	1	1	0	1
1	0	0	1	0	0
1	0	1	1	0	0
1	1	0	1	0	0
1	1	1	1	1	0

we have immediately:

$$\begin{bmatrix} 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \\ 1 \end{bmatrix} = \begin{bmatrix} S \\ T \\ U \end{bmatrix}$$

$$\begin{bmatrix} 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \end{bmatrix} \leftarrow [A]_{\text{UNIT}}$$

which gives:

$$\begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} S \\ T \\ U \end{bmatrix}$$

Mathematical treatment of the above

Let $\bigcup$ represent union over a field, $\bigcap$ represent intersection over a field and $\cap$ represent intersection.

Given the coefficient matrix $[C]$, the identity matrix $[A]_{\text{UNIT}}$ and the variable vector $X]$, then:

$$f_i(x_1, x_2, \ldots x_n) = \bigcup_{j=1}^{2^n} c_{ij} \cap \left\{ \bigcap_{k=1}^{n} (a_{kj}.x_k + \bar{a}_{kj}.\bar{x}_k) \right\} \quad (2)$$

Gives

$$[C]\, X] = f_x]$$

For two input variables x_1, x_2 we have, from (2):

$$f_1(x_1, x_2) = \{c_{11}.(a_{11}.x_1 + \bar{a}_{11}.\bar{x}_1).(a_{21}.x_2 + \bar{a}_{21}.\bar{x}_2) + c_{12}.(a_{12}.x_1 + \bar{a}_{12}.\bar{x}_1).(a_{22}.x_2 + \bar{a}_{22}.\bar{x}_2) + c_{13}.(a_{13}.x_1 + \bar{a}_{13}.\bar{x}_1).(a_{23}.x_2 + \bar{a}_{23}.\bar{x}_2) + c_{14}.(a_{14}.x_1 + \bar{a}_{14}.\bar{x}_1).(a_{24}.x_2 + \bar{a}_{24}.\bar{x}_2)\}$$

Conversion of a matrix in binary form to 'decimal' form:

$$q'_j = \sum_{k=1}^{n} 2^{n-k} q_{kj}, \qquad 1 \leqslant j \leqslant 2^n \quad (3)$$

(q' is decimal form).

The identity matrix is chosen so that in its 'decimal' form $[A']_{\text{UNIT}}$ is given by:

$$a'_j = j - 1, \qquad 1 \leqslant j \leqslant 2^n \quad (4)$$

2. Boolean matrix multiplication

The method of multiplying two Boolean matrices together is an extension of the method employed above for the multiplication of a vector by a matrix.

We wish to multiply two Boolean matrices, say:

$$[B]\,[C] = [D]$$

we write the unit matrix directly under the pre-multiplying matrix:

$$[B]\,[C] = [D]$$
$$[A]_{\text{UNIT}}$$

Suppose the first column of [C] matches the fourth column of $[A]_{\mathrm{UNIT}}$ then the *first* column of [D] is equal to the *fourth* column of [B]. Again, suppose that the second column of [C] matches the second column of [A] then the second column of [D] is equal to the second column of [B]. We proceed in this fashion until all the columns of [C] (and thus [D]) have been operated upon, e.g.

$$\begin{bmatrix}1&0&1&1\\0&0&1&1\end{bmatrix}\begin{bmatrix}1&0&0&0\\1&1&0&1\end{bmatrix}=\begin{bmatrix}1&0&1&0\\1&0&0&0\end{bmatrix}$$

$$\begin{bmatrix}0&0&1&1\\0&1&0&1\end{bmatrix}\longleftarrow [A]_{\mathrm{UNIT}}$$

We may carry out the same operation in 'decimal' form:

$$[2\ \ 0\ \ 3\ \ 3]\,[3\ \ 1\ \ 0\ \ 1] = [3\ \ 0\ \ 2\ \ 0]$$
$$[0\ \ 1\ \ 2\ \ 3] \longleftarrow [A]_{\mathrm{UNIT}}$$

In general it can be shown that

$$[B]\,[C] \neq [C]\,[B]\,,$$

one exception is

$$[A]_{\mathrm{UNIT}}\,[B] = [B] = [B]\,[A]_{\mathrm{UNIT}}$$

Note also that the number of columns appearing in each matrix must be equal to 2^n where 'n' is the number of variables upon which the matrix is defined.

This principle may be extended to cover more than two matrices and the following can be shown to apply:

$$[B]\,[C]\,[D] = \{[B]\,[C]\}\,[D] = [B]\,\{[C]\,[D]\}$$

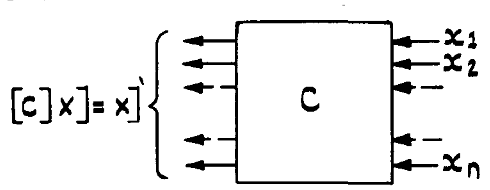


Fig. 1(*a*)

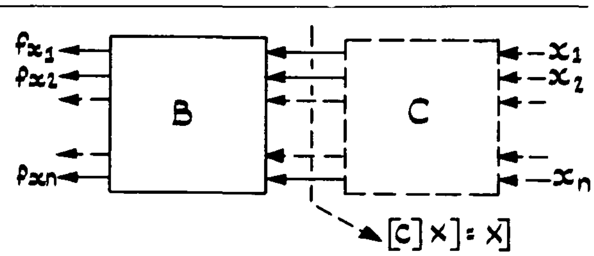


Fig. 1(*b*)

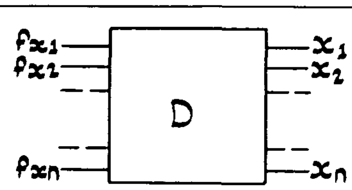


Fig. 1(*c*)

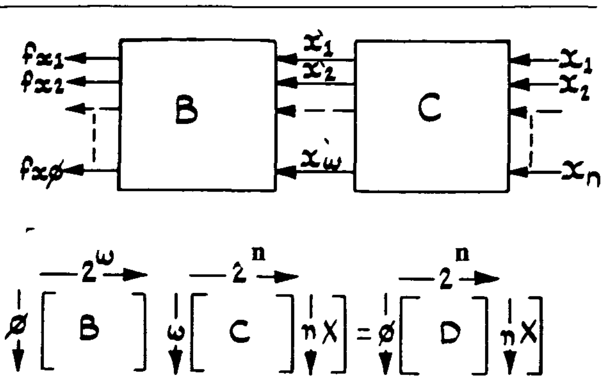


Fig. 1(*d*)

3. An interpretation of matrix multiplication

Consider

$$[B]\,[C]\,X] = f_x]$$

where [B] and [C] are two 'coefficient' matrices, X] is the variable vector and f_x] is the resulting vector. If we first evaluate [C] X] we in fact mathematically describe a system of the form shown in **Fig. 1**(*a*) where 'C' represents a logic circuit.

The output [C] X] we will call X]′, and is in fact a new variable vector. We have reduced the initial equation to

$$[B]\,X]' = f_x]$$

which may be interpreted as **Fig. 1**(*b*). Alternatively, if we had first evaluated [B] [C] = [D] we would have reduced Fig. 1(*b*) to a single logic block **Fig. 1**(*c*)).

Boolean matrix multiplication therefore, can be regarded as a representation of cascaded logic blocks.

It should be noted that matrix multiplication is not limited to matrices of the same dimensions. **Fig. 1**(*d*) shows the multiplication of matrices of differing dimensions, together with a cascaded block representation.

Example:

It is required that the two logic units shown in **Fig. 2**(*a*) should be cascaded as shown in **Fig. 2**(*b*). What are the functions appearing at the output?

Given; truth tables for 'B' and 'C':

x_b, x_c	y_b, y_c	fx_b	fy_b	fx_c	fy_c
0	0	1	0	0	1
0	1	1	0	1	1
1	0	1	1	1	0
1	1	0	1	0	0

We have

$$\begin{bmatrix}1&1&1&0\\0&0&1&1\end{bmatrix}\begin{bmatrix}0&1&1&0\\1&1&0&0\end{bmatrix}\begin{bmatrix}x\\y\end{bmatrix}=\begin{bmatrix}f_x\\f_y\end{bmatrix}$$

$$\begin{bmatrix}0&0&1&1\\0&1&0&1\end{bmatrix}\longleftarrow [A]_{\mathrm{UNIT}}$$

giving the result:

$$\begin{bmatrix}1&0&1&1\\0&1&1&0\end{bmatrix}\begin{bmatrix}x\\y\end{bmatrix}=\begin{bmatrix}f_x\\f_y\end{bmatrix}$$

As a truth table:

x	y	f_x	f_y
0	0	1	0
0	1	0	1
1	0	1	1
1	1	1	0

whence

$$f_x = (\bar{x}.\bar{y} + x.\bar{y} + x.y) = x + \bar{y}$$
$$f_y = (\bar{x}.y + x.\bar{y}) = x \oplus y$$

The answer may be checked by applying Boolean algebra to each unit in turn, a rather tedious process which illustrates one of the advantages of Boolean matrix multiplication.

Mathematical treatment of multiplication

$$[B]\,[C] = [D]$$

may be evaluated as (see Fig. 1(*d*))

$$d_{rj} = \bigcup_{m=1}^{2\omega} b_{rm} \cap \left\{ \bigcap_{i=1}^{\omega} (a_{im}.c_{ij} + \bar{a}_{im}.\bar{c}_{ij}) \right\},\ \begin{matrix}1 \leqslant r \leqslant \phi,\\ 1 \leqslant j \leqslant 2^n.\end{matrix} \tag{5}$$

In 'decimal' form we write

$$B\{C\{J\} + 1\} = D\{J\} \qquad 1 \leqslant J \leqslant 2^n \tag{6}$$

where $D\{J\}$ represents the number lying in the Jth column of [D], e.g.

for

$$[2\ \ 3\ \ 1\ \ 1]\,[0\ \ 2\ \ 1\ \ 3] = [2\ \ 1\ \ 3\ \ 1]$$

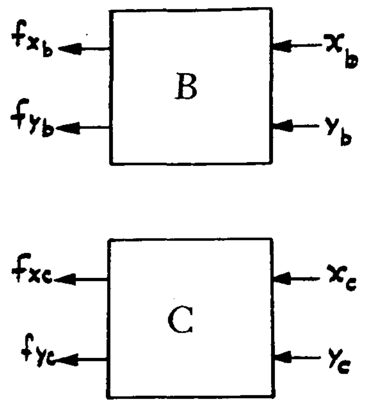


Fig. 2(*a*)

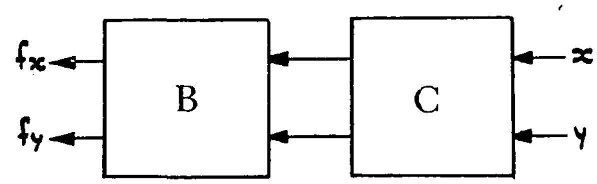


Fig. 2(*b*)

For the second column of D: $D\{2\} = B\{C\{2\} + 1\}$

$$= B\{2 + 1\}$$
$$= B\{3\}$$
$$= 1$$

4. The inverse matrix

We may have a system given by $[B]\,[C] = [D]$ where $[B]$ and $[D]$ are known and we wish to evaluate $[C]$. In 'conventional' matrix algebra we would write $[C] = [B]^{-1}\,[D]$. The question now arises, does $[B]^{-1}$ exist and how do we find it?

It can be shown that the inverse of a boolean matrix exists if the matrix is 'non-singular', that is if all columns of the matrix are different, e.g.

$$\begin{bmatrix} 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 1 \end{bmatrix} \text{ and } [2 \;\; 3 \;\; 0 \;\; 1] \text{ are non-singular,}$$

whereas

$$\begin{bmatrix} 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 \end{bmatrix} \text{ and } [0 \;\; 0 \;\; 3 \;\; 2] \text{ are singular}$$

The inverse matrix is found from:

$$[B]^{-1}[B] \triangleq [A]_{\text{UNIT}} \triangleq [B][B]^{-1}$$

Since $[A]_{\text{UNIT}}$ is the unit matrix the expression is analogous to

$$B^{-1} = \frac{1}{B}.$$

Note also that since $[A]_{\text{UNIT}}$ is non-singular by definition, both $[B]^{-1}$ and $[B]$ must be non-singular.

Pre- and post-multiplication is preserved, e.g.

Given $\quad [B]\,[C] = [D]$

then $\quad [C] = [B]^{-1}\,[D]$

$\quad\quad [B] = [D]\,[C]^{-1}$

Example:

In Fig. 2(*b*) suppose that the overall response of the system is given by $[B]\,[C]\,X] = [D]\,X]$ where $X]$ represents the vector $\begin{bmatrix} x \\ y \end{bmatrix}$.

Given

$$[D] = [2 \;\; 1 \;\; 3 \;\; 2]$$

and

$$[B] = [3 \;\; 0 \;\; 2 \;\; 1]$$

find the truth table for $[C]$.

Now $[B]$ is non-singular so its inverse exists. From

$$[B]^{-1}\,[B] = [A]_{\text{UNIT}}$$

or

$$[B]^{-1}\,[3 \;\; 0 \;\; 2 \;\; 1] = [0 \;\; 1 \;\; 2 \;\; 3]$$

we construct $[B]^{-1}$ by inspection:

$$[1 \;\; 3 \;\; 2 \;\; 0]\,[3 \;\; 0 \;\; 2 \;\; 1] = [0 \;\; 1 \;\; 2 \;\; 3]$$
$$[0 \;\; 1 \;\; 2 \;\; 3]$$

Thence the matrix $[C]$, given by $[C] = [B]^{-1}\,[D]$, is:

$$[C] = [1 \;\; 3 \;\; 2 \;\; 0]\,[2 \;\; 1 \;\; 3 \;\; 2] = [2 \;\; 3 \;\; 0 \;\; 2]$$
$$= \begin{bmatrix} 1 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 \end{bmatrix}$$

The truth table for 'C' is therefore:

x	y	fx_c	fy_c
0	0	1	0
0	1	1	1
1	0	0	0
1	1	1	0

The result should be checked by the reader.

Mathematical treatment of the inverse matrix

The inverse matrix is most conveniently expressed by the decimal notation (see (6)).

$[B]$ is non-singular if

$$B\{J\} \neq B\{K\} \qquad 1 \leqslant J < K \leqslant 2^n \tag{7}$$

Then with $[B]$ non-singular, $[B]^{-1}$ is given by

$$B^{-1}\{B\{J\} + 1\} = J - 1 \qquad 1 \leqslant J \leqslant 2^n \tag{8}$$

It can be shown that an algebra of singular inverse matrices exists which enables the generation and manipulation of functions with 'don't care' conditions. Unfortunately space will not allow the inclusion of this topic here.

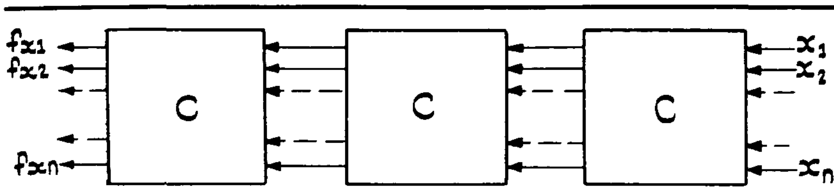


Fig. 3(*a*)

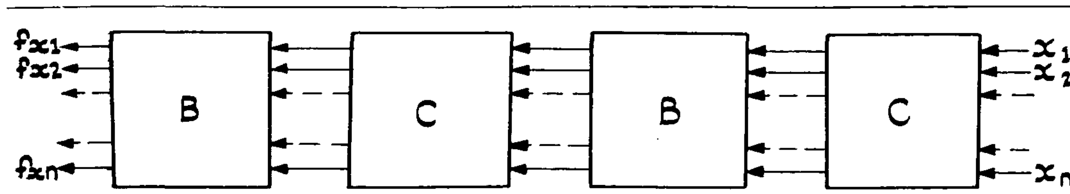


Fig.(3*b*)

5. The 'powers' of matrices

Consider **Fig.** 3(a) which represents a cascade of three identical logic blocks. We can now evaluate the response of this system as:

$$[C]\,[C]\,[C]\,X] = f]\,,$$

which is written in shorthand form as $[C]^3\,X] = f]$.

Similarly, we may evaluate **Fig.** 3(b) as $[B]\,[C]\,[B]\,[C] = f]$ or

$$[[B]\,[C]]^2 = f]$$

Note that in general $[[B]\,[C]]^2 \neq [B]^2\,[C]^2$.

The author has found that certain non-singular matrices obey an indices law similar to that of conventional algebra and such matrices may be used in synthesis of logic arrays. This, however, is beyond the scope of this article.

When a problem involves large numbers of variables or requires high 'powers' of matrices to be evaluated, the use of a digital computer is clearly essential.

6. Further operations

We may require to analyse the response of the system shown in **Fig.** 4(a), having a 'crossed' connection.

One solution is to introduce a new module 'X' containing a 'crossover'—but what is the form of the matrix to do this?

It is found that if we take the $[A]_{\text{UNIT}}$ matrix ($n = 2$) and interchange the ROWS corresponding to 'x' and 'y' the result is the matrix required, e.g. in **Fig.** 4(b)

$$[B]\begin{bmatrix}0 & 1 & 0 & 1\\ 0 & 0 & 1 & 1\end{bmatrix}[C]\begin{bmatrix}x\\ y\end{bmatrix} = \begin{bmatrix}f_x\\ f_y\end{bmatrix}$$

The same result can be obtained by deriving 'X' from a truth table but is very tedious, especially when the number of variables is high. The modification of the identity matrix in this way provides an 'operator' which can be derived quickly and easily as needs arise. A more complex example appears in **Fig.** 4(c).

The concept of an operator is not limited to simple 'crossovers' **Fig.** 4(d) shows a more interesting problem. To derive 'X', the operator, we first take the $[A]_{\text{UNIT}}$ matrix and 'AND' its two rows together term by term; this becomes the row of 'X' corresponding to 'x'. We then 'OR' the two rows of $[A]_{\text{UNIT}}$; this becomes the row of 'X' corresponding to 'y'.

That is, from

$$[A]_{\text{UNIT}} = \begin{bmatrix}0 & 0 & 1 & 1\\ 0 & 1 & 0 & 1\end{bmatrix}$$

we obtain

$$[X] = \begin{bmatrix}0 & 0 & 0 & 1\\ 0 & 1 & 1 & 1\end{bmatrix}\begin{matrix}\leftarrow \text{'}x\text{'th row}\\ \leftarrow \text{'}y\text{'th row}\end{matrix}$$

In practice this method can be very powerful in the logical manipulation of whole functions.

Mathematically it can be shown that if a matrix is multiplied by the identity matrix having its rows modified in a particular fashion then the rows of the 'multiplicand' matrix undergo the same modification.

More complex operators have been developed by the author to make possible the analysis of circuits of the type shown in **Fig. 5** in any number of dimensions. These operators, when raised to given powers, cycle the inputs and outputs of the logic cells so that the organisation of the array is duplicated.

Other operators have been developed which perturb the columns of a given matrix. It can be shown that this is equivalent to the redefinition of the functions concerned in terms of combinations of the defining variables, and as such is the first step in a synthesis procedure.

7. Computer implementation

The operations described above are readily carried out by a digital computer without resort to special programming techniques.

A matrix of ten tenth order functions may be represented by a vector of 1,024 ten-bit integers. The multiplication of two such matrices allows the manipulation of ten functions simul-

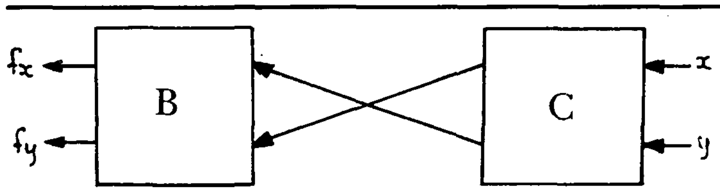


Fig. 4(a)

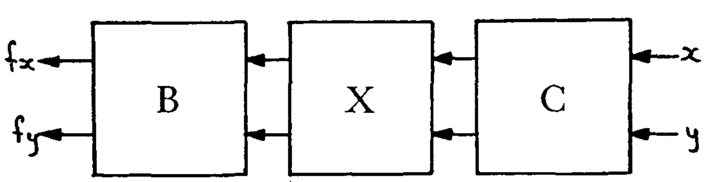
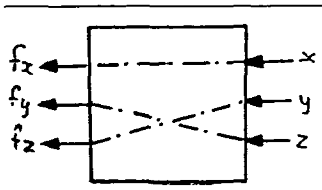


Fig. 4(b)



$$\equiv \begin{bmatrix}0 & 0 & 0 & 0 & 1 & 1 & 1 & 1\\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1\\ 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1\end{bmatrix}\begin{bmatrix}x\\ y\\ z\end{bmatrix} = \begin{bmatrix}f_x\\ f_y\\ f_z\end{bmatrix}$$

Fig. 4(c)

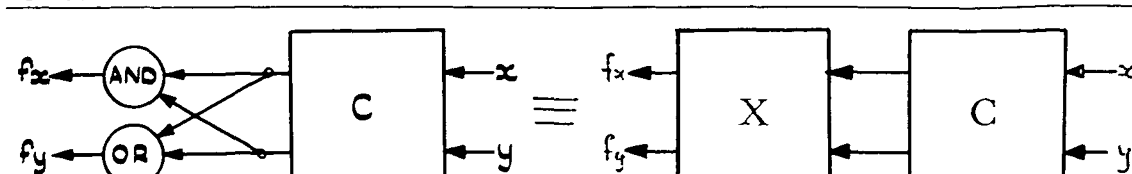


Fig. 4(d)

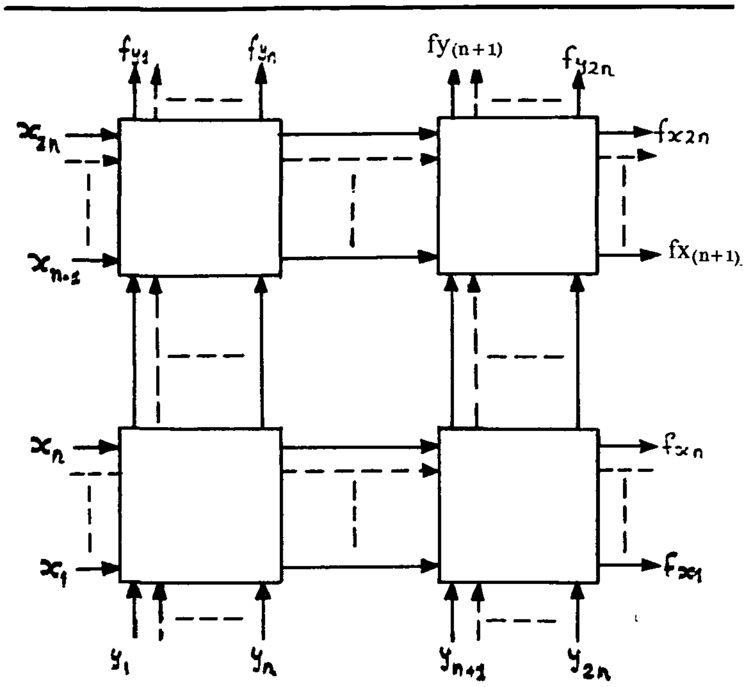


Fig. 5

2. Evaluate the operator $[Y]$
   Take

$$[A]_{\text{UNIT}} = \begin{bmatrix} 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 \end{bmatrix}$$

   'OR' the two rows, giving:

   0 1 1 1 as the upper row of $[Y]$

   'AND' the two rows, giving:

   0 0 0 1 as the lower row of $[Y]$

$$\therefore [Y] = \begin{bmatrix} 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

3. Evaluate $[X]$, a third order system:

$$[A]_{\text{UNIT}} = \begin{bmatrix} 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \end{bmatrix}$$

   Interchange top and bottom rows to give required cross-over, whence:

$$[X] = \begin{bmatrix} 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \end{bmatrix}$$

4. Multiply out the matrix equation in decimal form to give answer:

'3'

$$\begin{matrix}[0\ 2\ 3\ 1\ 3\ 2\ 0\ 3] \\ 0\ 1\ 2\ 3\ 4\ 5\ 6\ 7\end{matrix}\ \begin{matrix}[0\ 4\ 2\ 6\ 1\ 5\ 3\ 7] \\ 0\ 1\ 2\ 3\ 4\ 5\ 6\ 7\end{matrix}\ \begin{matrix}[7\ 4\ 3\ 5] \\ 0\ 1\ 2\ 3\end{matrix}\ \begin{matrix}[0\ 2\ 2\ 3] \\ 0\ 1\ 2\ 3\end{matrix}\ [2\ 0\ 2\ 3] \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} f_x \\ f_y \end{bmatrix}$$

taneously. Calculations of this order, quite adequate for most practical purposes, may be carried out on small machines.

Matrix multiplication is implemented by store interchanging, and extensive use is made of word comparison when setting up special operators.

Because both the matrices and operations are rigorously defined, fast and elegant routines result.

An example is now given which embodies most of the techniques developed in this article.

Example:

Find the response of the system shown in **Fig. 6**. Given

$$[B] = \begin{bmatrix} 0 & 1 & 1 & 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 & 1 & 0 & 0 & 1 \end{bmatrix},$$

$$[C] = \begin{bmatrix} 1 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 1 \end{bmatrix},$$

$$[D] = \begin{bmatrix} 1 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Solution

1. Write the matrix equation of the system putting in 'dummy' matrices for the inter-block operators:

$$[B]\,[X]\,[C]\,[Y]\,[D] \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} f_x \\ f_y \end{bmatrix}$$

finally yielding: $[0 \quad 3 \quad 0 \quad 2] \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} f_x \\ f_y \end{bmatrix}$

Note that the multiplication of step 4 can be carried out column by column thus avoiding the tedious process of multiplying out completely each matrix in turn. The arrow shows the evaluation of the second column of the answer.

5. Write truth table of this final matrix:

x	y	f_x	f_y
0	0	0	0
0	1	1	1
1	0	0	0
1	1	1	0

whence: $f_x = y$
$f_y = \bar{x}.y$

8. Conclusion

The main advantages of analysis (and synthesis) of systems by Boolean matrices of the type discussed is that they define a system's interconnections and logic functions in a rigorous, formalised way and are readily manipulated by the digital computer.

Although it has not been possible to give a complete picture of all the operations possible using these matrices, indeed this is still the object of research, it is hoped that the reader will find these basic concepts useful and worthy of employment in the digital field.

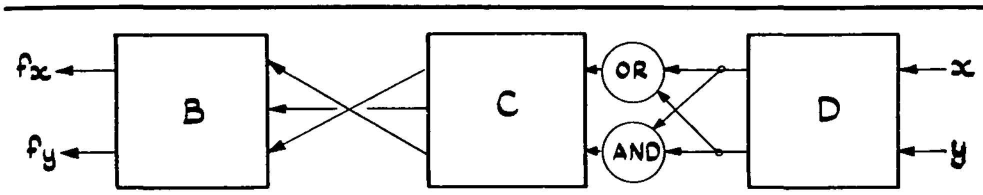


Fig. 6

Acknowledgements

The author would like to thank Dr. S. L. Hurst, University of Bath, for his invaluable advice. Prepared under Science Research Council (UK) Grant B/70/1295.

References

CAMPEAU, J. O. (1955). The Synthesis and Analysis of Counters in Digital Systems by Boolean Matrices. Masters Thesis, University of California (June).

CAMPEAU, J. O. (1957). The Synthesis and Analysis of Digital Systems by Boolean Matrices, *IRE Transactions on Electronic Computers*, Vol. EC-6, pp. 231-241.

EDWARDS, C. R. (1971). The Application of Boolean Matrices to Cellular Logic. Internal communication, University of Bath (June).

EDWARDS, C. R. (1972). Some Boolean Matrix Operators, *Electronics Letters*, Vol. 8, No. 5, pp. 113-115.

FLEGG, H. G. (1964). *Boolean Algebra and its Application*, pp. 181-206. Blackie.

LEWIN, D. (1968). *Logical Design of Switching Circuits*, pp. 31-35. Nelson.