



HAL
open science

Détail sur la résolution des équations de Boole en vue du codage

M. Carvalho

► **To cite this version:**

M. Carvalho. Détail sur la résolution des équations de Boole en vue du codage. Annales de l'ISUP, 1964, XIII (1), pp.21-44. hal-04085123

HAL Id: hal-04085123

<https://hal.science/hal-04085123>

Submitted on 28 Apr 2023

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons Attribution 4.0 International License

DÉTAIL SUR LA RÉOLUTION DES ÉQUATIONS DE BOOLE EN VUE DU CODAGE

M. CARVALLO

1 - INTRODUCTION -

Il est essentiel d'être mis en garde contre une généralisation trop hâtive du processus d'élimination exposé par G. Boole [1]. Celui-ci consiste essentiellement à éliminer une classe symbolique x entre une équation $f(x) = 0$ et l'expression fondamentale du principe de non-contradiction $x x' = 0$. L'on sort de ce point de vue, si l'on cherche à exprimer, a priori, l'inconnue x en fonction des coefficients de l'équation logique qui s'y rapporte.

Le but de cette étude est de préciser les notions de paramètre, de variable effective, de substitution, de "forme générale telle que...", notions d'usage courant en algèbre, mais dont l'emploi systématique, sans considération profonde des hypothèses faites, conduit à des erreurs. Tout particulièrement l'Algèbre de Boole, traitée sans ces précautions, devient source de fautes logiques. La difficulté une fois levée, oriente des recherches, principalement dans le problème de la représentation paramétrique : on donne ici en particulier un algorithme de résolution des équations et inéquations de Boole à l'aide du nombre minimal de paramètres, utile au calcul opérationnel.

Cette étude comprend quatre parties :

- 1/ L'équation à une inconnue de Boole où intervient la notion de *paramètre*
- 2/ Le problème du codage résolu par *substitution* ou à partir de solutions *particulères*
- 3/ Les systèmes d'équations qui rendent utile l'introduction d'un algorithme *matriciel* permettant d'explicitier les solutions.
- 4/ Les équations fonctionnelles, où intervient la notion de variable *effective*

Les méthodes et exemples exposés ici, précisent et généralisent les méthodes données antérieurement [2]. D'une part, les résultats englobent une variété plus grande de solutions car ils tiennent compte d'une liberté plus grande des données (celle de dépendre de paramètres), d'autre part, ils permettent un dénombrement ou une classification des solutions, d'où un choix de celles-ci mieux adapté aux problèmes pratiques éventuels.

Des applications seront données dans un prochain ouvrage [3].

2 - EQUATIONS A UNE INCONNUE -

$$ax' + bx = 0 \quad (1)$$

Toute équation de Boole à une inconnue peut être mise sous cette forme puisque $\begin{bmatrix} a & b \end{bmatrix} \begin{bmatrix} x' \\ x \end{bmatrix}$ est la forme disjonctive normale d'une fonction d'une variable x .

Nous avons donné plusieurs méthodes de résolution. Elles consistent à supposer que x est une fonction booléenne de a et b dont la forme la plus générale est :

$$x = [\alpha \beta \gamma \delta] \begin{bmatrix} ab \\ ab' \\ a'b \\ a'b' \end{bmatrix} \quad (2)$$

La première matrice est celle des coefficients indéterminés, la seconde est la "base" des données.

La substitution de (2) dans (1) donne la condition :

$$ab = 0 \quad (3)$$

et la solution :

$$x = a + \delta b' \quad (4)$$

(δ paramètre de Boole arbitraire).

Il faut remarquer que si a et b ne sont pas des constantes données, mais peuvent dépendre de paramètres de Boole, alors la condition (3) reste la même, mais la formule (4) n'est pas nécessairement la forme la plus générale de x satisfaisant (1).

Ainsi, supposons que a et b dépendent du paramètre de Boole t, on a :

$$\begin{aligned} a(t) &= \begin{bmatrix} a(1) & a(0) \end{bmatrix} \begin{bmatrix} t \\ t' \end{bmatrix} \\ b(t) &= \begin{bmatrix} b(1) & b(0) \end{bmatrix} \begin{bmatrix} t \\ t' \end{bmatrix} \\ x &= \begin{bmatrix} \alpha & \beta \end{bmatrix} \begin{bmatrix} t \\ t' \end{bmatrix} \\ x' &= \begin{bmatrix} \alpha' & \beta' \end{bmatrix} \begin{bmatrix} t \\ t' \end{bmatrix} \end{aligned}$$

En portant ces expressions dans (1), on obtient la condition de possibilité $ab = 0$, quel que soit t, ou

$$\begin{aligned} a(1) b(1) &= 0 \\ a(0) b(0) &= 0 \end{aligned} \quad (5)$$

et la solution,

$$x = (a(1) + \lambda \overline{b(1)})t + (a(0) + \mu \overline{b(0)})t' \quad (6)$$

λ et μ sont des paramètres de Boole indépendants provenant de la résolution des équations

$$\begin{aligned} a(1)\alpha' + b(1)\alpha &= 0 \\ a(0)\beta' + b(0)\beta &= 0 \end{aligned}$$

L'application de (4) à ces deux équations en α et β donne en effet

$$\begin{aligned} \alpha &= a(1) + \lambda \overline{b(1)} \\ \beta &= a(0) + \mu \overline{b(0)} \end{aligned}$$

avec les conditions (5).

L'application directe de (4) à (1) n'eût permis de faire apparaître qu'un paramètre donnant une solution moins générale

$$x = (a(1) + \lambda \overline{b(1)})t + (a(0) + \lambda \overline{b(0)})t'$$

D'une façon générale, la solution de (1) où a et b dépendent de paramètres t_1, t_2, \dots, t_m s'obtiendra en portant dans (1), la forme générale de x

$$x = [\alpha_{2^m-1}, \dots, \alpha_0] T \tag{7}$$

où T est la base des paramètres t_i

$$T = \begin{bmatrix} t_1 & t_2 & \dots & t_m \\ t'_1 & t'_2 & \dots & t'_m \\ \dots & \dots & \dots & \dots \\ t'_1 & t'_2 & \dots & t'_m \end{bmatrix}$$

on a

$$x' = [\alpha'_{2^m-1} \dots \alpha'_0] T$$

La substitution amène donc à résoudre un système de 2^m équations à 2^m inconnues α_i

$$a_i \alpha'_i + b_i \alpha_i = 0$$

d'où

$$\alpha_i = a_i + \lambda'_i b'_i$$

où

$$a(t_1, t_2, \dots, t_m) = [a_{2^m-1} \dots a_i \dots a_0] T$$

$$b(t_1, t_2, \dots, t_m) = [b_{2^m-1} \dots b_i \dots b_0] T$$

λ_i paramètres de Boole arbitraires

$$\begin{aligned} a_{2^m-1} &= a(1, 1, \dots, 1) \\ \dots & \dots \dots \dots \dots \dots \dots \dots \\ b_0 &= b(0, 0, \dots, 0) \end{aligned}$$

La solution est donc :

$$x = [(a_{2^m-1} + \lambda_{2^m-1} b'_{2^m-1}) \dots (a_0 + \lambda_0 b'_0)] T \tag{8}$$

avec la condition

$$(\forall i) a_i b_i = 0 \quad \text{soit} \quad ab = 0$$

Application - $f(a,b)$ et $g(a,b)$ étant des fonctions de Boole données, déterminer la forme la plus générale de la fonction de Boole $x(a,b)$ telle que

$$\begin{aligned} f(a, b) = 1 &\longrightarrow x = 1 \\ g(a, b) = 1 &\longrightarrow x = 0 \end{aligned} \tag{9}$$

Le système donné est équivalent à l'équation

$$f x' + g x = 0$$

Ce qui exige la condition $fg = 0$.

La solution la plus générale n'est pas $x = f + \lambda g$, mais l'expression à 4 paramètres de Boole indépendants suivante

$$x = \left[\left(f(1,1) + \lambda_3 g'(1,1) \right) \left(f(1,0) + \lambda_2 g'(1,0) \right) \right. \\ \left. \left(f(0,1) + \lambda_1 g'(0,1) \right) \left(f(0,0) + \lambda_0 g'(0,0) \right) \right] \begin{bmatrix} ab \\ ab' \\ a'b \\ a'b' \end{bmatrix}$$

3 - EQUATIONS DE BOOLE A n INCONNUES

$$f(x_1, x_2, \dots, x_n) = 0 \tag{11}$$

La méthode d'élimination donne toujours une solution, c'est-à-dire une représentation paramétrique

$$x_i(\lambda_1, \lambda_2, \dots, \lambda_m)$$

à paramètres de Boole λ_j indépendants.

Elle consiste à écrire (11) sous la forme (1) par rapport à une inconnue x_1 , par exemple

$$f(0, x_2, \dots, x_n) x_1' + f(1, x_2, \dots, x_n) x_1 = 0 \tag{12}$$

d'où x_1 par la formule (4), est une nouvelle équation portant sur $n - 1$ inconnues par la formule (3).

Cette méthode ne donne pas nécessairement la solution la plus économique en paramètres, car d'après notre remarque du paragraphe I, la solution la plus générale de (12) n'est pas celle donnée par (4).

Exemple - L'équation

$$xy + yz + xz = 0$$

traitée par élimination, donne

$$yzx' + (y + z)x = 0$$

d'où

$$x = yz + \lambda (y + z)'$$

avec la condition

$$yz(y + z) = 0 \quad \text{ou} \quad yz = 0$$

d'où

$$y = \alpha$$

$$z = \alpha' \beta$$

et

$$x = \lambda \alpha' \beta'$$

Cette solution fait apparaître 3 paramètres α , β , λ , indépendants, elle suppose en effet que x est nécessairement de la forme (4), ce qui n'est pas le cas dans la solution à 2 paramètres

$$x = \alpha\beta$$

$$y = \alpha\beta'$$

$$z = \alpha' \beta$$

Remarque - On peut déterminer a priori le nombre minimum p de paramètres nécessaires au codage (N étant le nombre de solutions de (11), on a $2^p \geq N$) et chercher si les x_i peuvent être choisis parmi les éléments de la base de l'espace des paramètres.

Cette solution, si elle existe, peut échapper à la méthode d'élimination.

Dans l'exemple précédent

$$xy + yz + xz = 0$$

on a $N = 4$, d'où $p = 2$, on peut prendre pour x , y , z , 3 éléments différents quelconques de la base

$$\begin{bmatrix} \alpha\beta \\ \alpha\beta' \\ \alpha'\beta \\ \alpha'\beta' \end{bmatrix}$$

D'une façon générale, on peut, pour chercher un codage de $f(x_1, x_2, \dots, x_n) = 0$ à p paramètres $\alpha_1, \alpha_2, \dots, \alpha_p$, chercher une

on a donc à résoudre le système

$$(\forall i) \begin{cases} f(\lambda_1^i, \lambda_2^i, \dots, \lambda_n^i) = 0 \\ \prod_k \lambda_i^k + \prod_k \bar{\lambda}_i^k = 0 \end{cases}$$

Mais, précisément, le codage obtenu par élimination donne, en faisant varier les paramètres du codage, toutes les solutions de (11) ; on pourra donc considérer un système de 2^p solutions

$$X = \begin{bmatrix} x_1^1 & \dots & x_1^{2^p} \\ x_2^1 & \dots & x_2^{2^p} \\ \dots & \dots & \dots \\ x_n^1 & \dots & x_n^{2^p} \end{bmatrix}$$

X est une matrice (n, 2^p) dont chaque colonne représente un système de solutions, obtenu en donnant aux paramètres du codage des valeurs particulières.

Aucune ligne de X ne doit être formée exclusivement de 0, ni exclusivement de 1 et toutes les solutions de (11) devront apparaître dans les colonnes de X.

Le codage à p paramètres cherché est donc

$$\begin{bmatrix} x_1 \\ x_2 \\ \dots \\ x_n \end{bmatrix} = X \begin{bmatrix} \alpha_1 \times \alpha_2 \times \dots \times \alpha_p \\ \alpha_1^1 \times \alpha_2^1 \times \dots \times \alpha_p^1 \end{bmatrix}$$

ce codage convient si tous les α_p sont effectivement explicites dans le développement.

Remarque : La résolution par élimination n'est pas nécessaire si l'on connaît 2^p solutions de l'équation ($2^p \geq N$). Ainsi considérons l'équation $x_1 + x_2 + x_3 = 2$ où les x sont des nombres de Boole et le signe + le signe d'addition de l'arithmétique ordinaire.

Les solutions sont au nombre de 4 :

$$\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 1 & 1 \\ 1 & 1 & 1 & 0 \\ 0 & 1 & 0 & 1 \end{bmatrix}$$

d'où la représentation paramétrique à 2 paramètres ($2^2 = 4$)

$$\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 1 & 1 \\ 1 & 1 & 1 & 0 \\ 0 & 1 & 0 & 1 \end{bmatrix} \begin{bmatrix} \alpha_1 & \alpha_2 \\ \alpha_1 & \alpha_2' \\ \alpha_1' & \alpha_2 \\ \alpha_1' & \alpha_2' \end{bmatrix} = \begin{bmatrix} \alpha_1 & \alpha_2 + \alpha_1' \\ \alpha_1 + \alpha_2 \\ \alpha_2' \end{bmatrix} = \begin{bmatrix} \alpha_1' + \alpha_2 \\ \alpha_1 + \alpha_2 \\ \alpha_2' \end{bmatrix} \quad (+ \text{ de Boole})$$

on arrive au même codage en traitant par élimination l'équation de Boole équivalente :

$$x_1 x_2 x_3 + x_1' x_2' + x_1' x_3' + x_2' x_3' = 0 \quad (+ \text{ de Boole})$$

(correspondant à $2^p > N$)

Soit à résoudre l'inéquation arithmétique portant sur des variables de Boole

$$\sum_{i=1}^n x_i \leq 1$$

on a $(n+1)$ solutions

$$\begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}$$

qui sont les colonnes de la matrice

unité $E(n, n)$, et la solution

$$\begin{bmatrix} 0 \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}$$

On détermine donc le plus petit entier p tel que $2^p \geq n+1$.
 $[\alpha]$ étant alors la matrice de base $(2^p, 1)$ des paramètres

$$[\alpha] = \begin{bmatrix} \alpha_1 \times \alpha_2 \times \dots \times \alpha_n \\ \alpha_1' \times \alpha_2' \times \dots \times \alpha_n' \end{bmatrix}$$

On peut choisir pour $X(n, 2^p)$, la matrice $X = [E|0]$ formée de la matrice unité $E(n, n)$ et d'une matrice $0(n, 2^p - n)$ dont tous les éléments sont nuls. On obtient alors le codage à p paramètres

$$\begin{bmatrix} x_1 \\ x_2 \\ \cdot \\ \cdot \\ x_n \end{bmatrix} = [E|0] [\alpha]$$

Ainsi, pour $n = 4$, on a $p = 3$, on trouve

$$\begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} \alpha_1 & \alpha_2 & \alpha_3 \\ \alpha_1 & \alpha_2 & \alpha_3' \\ \alpha_1' & \alpha_2' & \alpha_3 \\ \alpha_1 & \alpha_2' & \alpha_3' \end{bmatrix}$$

Nota : Le résultat donné par R. Fortet [4] dans ce cas particulier ($n = 4$)

$$\begin{bmatrix} x_1 = \alpha_2 \alpha_3 \\ x_2 = \alpha_2 \alpha_3' \\ x_3 = \alpha_2' \alpha_3 \\ x_4 = \alpha_1 \alpha_2' \alpha_3 \end{bmatrix}$$

correspondant au choix de X suivant

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

on voit qu'un bon choix de X peut donner des expressions plus simples à p paramètres mais qui ne respectent plus la symétrie.

Généralisation : On a souvent besoin en calcul opérationnel de résoudre l'inéquation $\sum_{i=1}^n x_i \leq m$, m nombre entier $m \leq n$, \sum arithmétique. La somme du premier membre est égale à un entier s donné pour C_n^s systèmes de x_i , donc le nombre de solutions de l'inéquation est $N = 1 + C_n^1 + C_n^2 + \dots + C_n^m$. Soit donc p le plus petit entier tel que $2^p \leq N$ (si $m = n$, $N = 2^p$ et $p = n$).

Désignons par M_{n, c_n^s} une matrice de Boole (n, C_n^s) , dont toutes les colonnes sont différentes et contiennent s éléments égaux à 1, chacune ; M_{n, c_n^1} est la matrice unité $E(n, n)$; soit $O_{n, q}$ la matrice (n, q) dont tous les éléments sont nuls, $q = 2^p - N + 1$; le codage à p paramètres $\alpha_1, \dots, \alpha_p$ de base $[\alpha] (2^p, 1)$ peut s'écrire :

$$\begin{bmatrix} x_1 \\ x_2 \\ \cdot \\ \cdot \\ \cdot \\ x_n \end{bmatrix} = [E_{n, n} \quad M_{n, c_n^2} \quad M_{n, c_n^3} \quad \dots \quad M_{n, c_n^m} \quad O_{n, q}] [\alpha]$$

Méthode générale : Cette méthode de codage, suggérée par le processus de substitution, doit être précisée. (Quelles sont les conditions imposées à la matrice X ?). Reprenons-là, du point de vue formel :

Considérons n variables de Boole x_1, x_2, \dots, x_n et le vecteur $[x] (n, 1)$ de composantes x_i

$$[x] = \begin{bmatrix} x_1 \\ x_2 \\ \cdot \\ \cdot \\ \cdot \\ x_n \end{bmatrix}$$

Supposons que $[x]$ soit susceptible de prendre N valeurs différentes X^i (soit X^1, X^2, \dots, X^N) et ces N valeurs seulement. Chaque X^i est un vecteur dont les n composantes sont données numériquement (égales à 0 ou 1).

Considérons par ailleurs p paramètres de Boole, $\alpha_1, \alpha_2, \dots, \alpha_p$, p étant le plus petit entier tel que $2^p \geq N \cdot (C_1)$. Désignons par $[\alpha] (2^p, 1)$ la base de l'espace correspondant :

$$[\alpha] = \begin{bmatrix} \alpha_1 & \alpha_2 & \dots & \alpha_p \\ \alpha_1' & \alpha_2' & \dots & \alpha_p' \\ \dots & \dots & \dots & \dots \\ \alpha_1' & \alpha_2' & \dots & \alpha_p' \end{bmatrix}$$

Désignons par $X (n, 2^p)$ une matrice de Boole de colonnes X^j , associée au système de N valeurs précédentes X_i , de façon que

$$\begin{aligned} (\forall i) (E_j) [X^j = X^i] \\ (\forall j) (E_i) [X^i = X^j] \end{aligned} \quad (C_2)$$

Ainsi, si $2^p = N$, tous les X^j sont différents (c'est-à-dire ont au moins une composante différente) et si $2^p > N$, certains X^j sont égaux entre eux. A un même système X^1, X^2, \dots, X^N peuvent être associées plusieurs matrices X (différant par l'ordre des colonnes et par le choix des colonnes égales si $2^p > N$). Le résultat suivant est valable pour une quelconque de ces matrices X .

L'équation matricielle

$$[x] = [X] [\alpha]$$

donne une représentation paramétrique des $x_k (k = 1, \dots, n)$ à l'aide des paramètres $\alpha_i (i = 1, \dots, p)$.

Cela signifie :

a) Pour tout système de valeurs numériques x_k prises par les composantes de $[x]$, ($[x]$ est nécessairement une colonne de X), il existe un système de valeurs numériques α_i , tel que :

$$[x] = [X] [\alpha]$$

En effet si $[x] = X^i$, on sait qu'il existe au moins un j tel que $X^j = X^i = [x]$. Alors le système de valeurs des α qui rend égal à 1 la ligne de rang j de α , rend identiquement nulles les autres lignes de α et l'on a bien $[X^i] = [X][e_j]$ (en désignant par e_j la matrice $(n, 1)$ dont tous les éléments sont nuls sauf celui de la ligne i , égal à 1).

b) Pour tout système de valeurs numériques α_i (il y a 2^p systèmes), l'équation $[x] = [X][\alpha]$ donne pour $[x]$ un système de composantes égal à une composante de $[X]$.

En effet $[\alpha]$ se réduit à $[e_h]$, h étant le rang de la seule ligne de la base qui n'est pas annulée par le système des α_l et $[X][e_h]$ est la colonne de rang h de $[X]$.

Conclusion - Etant donné une équation $f(x_1, x_2, \dots, x_n) = 0$, on cherchera le système des N solutions différentes $x = (x_1, \dots, x_n)$ (soit directement soit par élimination) et on trouvera tous les codages possibles avec le nombre p de paramètres minimal en associant à ce système les différentes matrices $[X]$ ($n, 2^p$), et la base $[\alpha]$ ($2^p, 1$).

La condition (C_1) assure la représentation la plus économique en paramètres, les conditions (C_2) assurent l'existence d'une application du système des N points de l'espace à n dimensions sur les 2^p sommets du cube de Boole à p dimensions.

Inversement, considérons l'équation $f(x_1, x_2, \dots, x_n) = 0$ et ses N solutions X_1, X_2, \dots, X_n et considérons une matrice $[X]$, ($n, 2^p$), de colonnes X^j .

Si l'équation matricielle

$$[x] = [X][\alpha]$$

représente un codage, où $[\alpha]$ est la base des paramètres, les conditions C_2 sont nécessairement satisfaites.

En effet

1/ Si $[x] = X^i$ est solution de $f = 0$, il lui correspond nécessairement un système α_l de paramètres, satisfaisant l'équation matricielle, qui se réduit alors à

$$X^i = X^j$$

j étant le rang de la ligne de $[\alpha]$ non annulée par α_l , ainsi

$$(i) (E_j) [X^j = X^i]$$

2/ Si α_l est un système donné de paramètres, (celui qui n'annule pas la ligne de rang j de $[\alpha]$), il lui correspond nécessairement une (et une seule) solution $[x] = X^i$, de $f = 0$, satisfaisant l'équation matricielle, qui se réduit alors à

$$X^i = X^j$$

ainsi

$$(j) (E_i) [X^j = X^i]$$

Il est donc nécessaire de connaître toutes les solutions de $f = 0$ pour la coder, par une équation matricielle faisant intervenir la base des paramètres.

4 - SYSTEMES DE m EQUATIONS DE BOOLE A n INCONNUES -

Le système

$$f_i(x_1, x_2, \dots, x_n) = 0 \quad i = 1, 2, \dots, m$$

est équivalent à l'équation unique

$$\sum f_i = 0$$

et peut être résolu par les méthodes précédentes.

Il est avantageux pratiquement d'utiliser l'algorithme suivant, adapté à un système du type (14) :

$$V X = U A \quad (14)$$

A est la base des données a_1, a_2, \dots, a_p :

$$A_{(2^p, 1)} = \begin{bmatrix} a_1 & a_2 & \dots & a_p \\ \dots & \dots & \dots & \dots \\ a_1' & a_2' & \dots & a_p' \end{bmatrix}$$

X est la base des inconnues x_1, x_2, \dots, x_m :

$$X_{(2^m, 1)} = \begin{bmatrix} x_1 & x_2 & \dots & x_m \\ \dots & \dots & \dots & \dots \\ x_1' & x_2' & \dots & x_m' \end{bmatrix}$$

V et U sont des matrices d'éléments 0 et 1 données

$$V_{(n, 2^m)} = \begin{bmatrix} v_{11} & \dots & v_{12^m} \\ \dots & \dots & \dots \\ v_{n1} & \dots & v_{n2^m} \end{bmatrix} \quad U_{(n, 2^p)} = \begin{bmatrix} u_{11} & \dots & u_{12^p} \\ \dots & \dots & \dots \\ u_{n1} & \dots & u_{n, 2^p} \end{bmatrix}$$

Il n'existe pas en général de matrice de Boole $H_{(2^m, n)}$ telle que $HV = E$ (matrice unité) de sorte qu'on ne peut tirer X de (14) par multiplication à gauche par H.

Mais il peut exister des matrices de Boole $P_{(2^m, 2^p)}$ telles que (15) $U = VP$.

Si P est une telle matrice, on aura

$$VX = V P A \quad (16)$$

alors

$$X = P A \quad (17)$$

satisfait à (16) donc a (14).

Inversement une solution X de (14) est nécessairement de la forme

$$X = Q A$$

Q est la matrice de Boole (2^n , 2^p) qui permet d'exprimer chaque produit fondamental de X sous forme disjonctive par rapport à la base A .

On a donc

$$V X = V Q A \text{ et d'après (14),}$$

$$U A = Q A$$

ou

$$(U - V Q) A = 0$$

$(U - VQ)$ est une matrice dont tous les éléments sont nuls car A étant une base, ses éléments sont linéairement indépendants. Ainsi $U = VQ$ (forme 15). La recherche des solutions X de (14) est donc équivalente à celles des matrices P telles que :

$$U = V P.$$

Pour que l'équation (17) soit vérifiée pour tout système de a_i , il faut que P ait un et un seul 1 par colonne, en effet, si la colonne de rang k de P était identiquement nulle, le système de a_i annulant toutes les lignes de A sauf celle de rang k , rendrait identiquement nul le produit

$$P A,$$

et

$$X = P A = 0$$

ne pourrait représenter une base.

Si la colonne de rang k de P avait deux éléments égaux à 1, le système de a_i précédent rendrait égales à 1 deux lignes du produit PA ,

X aurait deux composantes égales à 1 et ne pourrait représenter une base.

De (17) on déduit la valeur des inconnues

$$\begin{bmatrix} x_1 \\ x_2 \\ \cdot \\ \cdot \\ x_n \end{bmatrix} = \mathfrak{X} X$$

$\mathfrak{X}_{(n \times 2^m)}$, est une matrice de Boole dont les éléments sont déterminés par la donnée de m et n .

Recherche de P

Construisons la matrice $R_{(2^m, 2^p)}$, d'éléments r_{ij} de Boole tels que

$$r_{ij} = 1 \iff (k) (v_{ki} = u_{kj})$$

ou symboliquement

$$r_{ij} = 1 \iff V^i = U^j$$

V^i , i -ème colonne de V , U^j , j -ième colonne de U .

Montrons que toute matrice déduite de R et de mêmes dimensions que R , de façon à ne conserver qu'un et un seul 1 par colonne, est une matrice P (telle que $U = VP$). Considérons que $(V^i \equiv U^j)$ est un nombre de Boole égal à 1 si et seulement si $V^i = U^j$, R peut alors s'écrire :

$$R = \begin{bmatrix} (V^1 \equiv U^1) & (V^1 \equiv U^2) & \dots & (V^1 \equiv U^{2^p}) \\ \dots & \dots & \dots & \dots \\ (V^{2^m} \equiv U^1) & \dots & \dots & (V^{2^m} \equiv U^{2^p}) \end{bmatrix}$$

P , déduite de R de façon à n'avoir qu'un et un seul 1 par colonne est de la forme

$$P = \begin{bmatrix} 0 & 0 & \dots \\ \cdot & \cdot & \\ \cdot & \cdot & \\ \cdot & \cdot & \\ (V^{\lambda_1} \equiv U^1) & 0 & \dots \\ \cdot & \cdot & \\ \cdot & \cdot & \\ \cdot & \cdot & \\ 0 & 0 & \dots \end{bmatrix} \quad (V^{\lambda_j} \equiv U^j)$$

Les expressions $(V^{\lambda_j} \equiv U^j)$ figurant dans P sont supposées égales à 1 et il y a une et une seule expression de ce genre par colonne. (Il n'y a pas de conditions sur le nombre de 1 des lignes, de sorte que certains λ_j peuvent être égaux).

Le produit VP est alors une matrice Q dont l'élément q_{ij} est

$$Q_{ij} = [v_{i1} \ v_{i2} \ \dots \ v_{i\lambda_j} \ \dots \ v_{i2^m}] \begin{bmatrix} 0 \\ 0 \\ \cdot \\ \cdot \\ (V^{\lambda_j} \equiv U^j) = 1 \\ 0 \\ \cdot \\ \cdot \\ 0 \end{bmatrix} = v_{i\lambda_j} \times (V^{\lambda_j} \equiv U^j) = v_{i\lambda_j}$$

Mais puisque $(V^{\lambda_j} \equiv U^j) = 1$, les colonnes V^{λ_j} et U^j sont identiques et les termes de la ligne i sont égaux : $v_{i\lambda_j} = u_{ij}$

Ainsi $V.P = [u_{ij}] = U$

Remarques - On peut donner une expression booléenne de r_{ij}

$$r_{ij} = \prod_k (v_{ki} \equiv u_{kj}) = \prod_k (u_{kj} v_{ki} + u'_{kj} v'_{ki})$$

Les matrices P sont définies par $P = [p_{ij}]$

avec $p_{ij} = 1 \longrightarrow r_{ij} = 1$ soit $p_{ij} r'_{ij} = 0 \ (\forall i), (\forall j)$

et $(\forall j) \sum_i p_{ij} = 1 \quad (\sum \text{arithmétique})$

P n'est pas nécessairement une matrice de permutation car elle peut avoir plusieurs 1 par ligne.

Si l'on s'intéresse plus spécialement aux solutions provenant d'une matrice de permutation, on cherchera P par adjonction de la condition

$$(\forall i) \sum_j p_{ij} = 1$$

On est alors amené à un problème rencontré et résolu par codage par R. Fortet à l'occasion de la recherche des circuits hamiltoniens dans un graphe.

Signalons une méthode de calcul rapide dans le cas d'une matrice carrée (qui pourrait être appliquée aux graphes).

On considère la matrice $R_{(1,s)}$ en revenant à la notation r_{ij} pour $r_{ij} = 1$ et en conservant les 0.

Le déterminant de R est une somme algébrique de tous les produits en r_{ij} , construits en prenant les produits de s facteurs choisis dans des lignes et des colonnes différentes.

A chaque produit non identiquement nul (de degré s par rapport aux r) $\prod_{i,j} r_{ij}$, est associée une matrice P dont tous les éléments sont nuls sauf ceux qui occupent les positions (i j) figurant dans \prod . P a bien un et un seul élément égal à 1 par ligne et par colonne.

Cas particulier

Système

$$\prod_i x_i^{(\xi_i^j)} = \prod_i a_i^{(a_i^j)} \quad \begin{array}{l} j = 1, 2, \dots, m \\ i = 1, 2, \dots, n \end{array}$$

Rappelons la signification de la matrice $a^{(q)}$:

$$\begin{cases} a^{(1)} = a \text{ si } q = 1 \\ a^{(q)} = a' \text{ si } q = 0 \end{cases}$$

Posons

$$A^j = \prod a_i^{(a_i^j)} \quad \text{et} \quad X^j = \prod x_i^{(\xi_i^j)}$$

le système s'écrit

$$(\forall j) (A^j = X^j)$$

$[A]_{(2^n, n)}$ étant la base des a_i , on a

$$x_k = [\lambda_k^{2^n-1} \dots \lambda_k^0] A \quad \text{d'où} \quad X^j = \left[\prod_{i=1}^n \lambda_i^{2^n-1(\xi_i^j)} \dots \prod_{i=1}^n \lambda_i^0(\xi_i^j) \right] A$$

Chaque équation $A^j = X^j$ donne $n \lambda : \lambda_i^{j_0(\xi_i^{j_0})} = 1, i = 1, \dots, n$

Une solution particulière est

$$x_i = \sum_{j=1}^m \left[\xi_i^j \prod_{k=1}^n a_k^{(a_k^j)} \right] \quad (p)$$

Les autres solutions seront obtenues en déterminant les λ non identiques à 1 à l'aide d'équations du type

$$\prod_{k=0}^{2^n-1} \lambda_k^q = 0$$

que l'on pourra résoudre par codage, les λ étant exprimés à l'aide de paramètres α indépendants.

Exemple :

$$\begin{cases} x \cdot y' = a \cdot b' \\ x' \cdot y = a' \cdot b \end{cases}$$

a) Méthode générale

Le système s'écrit

$$\begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} x \cdot y \\ x \cdot y' \\ x' \cdot y \\ x' \cdot y' \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} a \cdot b \\ a \cdot b' \\ a' \cdot b \\ a' \cdot b' \end{bmatrix}$$

La matrice R est ici

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

Les matrices P déduites de R sont

$$P_1 = \begin{bmatrix} \bar{1} & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & \bar{1} \end{bmatrix}$$

$$P_2 = \begin{bmatrix} \bar{1} & 0 & 0 & \bar{1} \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

$$P_3 = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ \bar{1} & 0 & 0 & \bar{1} \end{bmatrix}$$

$$P_4 = \begin{bmatrix} 0 & 0 & 0 & \bar{1} \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ \bar{1} & 0 & 0 & 0 \end{bmatrix}$$

On a :

$$\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} \bar{1} & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} x y \\ x y' \\ x'y \\ x'y' \end{bmatrix}$$

d'où la matrice

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

Les systèmes de solutions $\begin{bmatrix} x \\ y \end{bmatrix} = X P A$ sont

$$1 \begin{cases} x = a \\ y = b \end{cases} \quad 2 \begin{cases} x = a + b' \\ y = b + a' \end{cases} \quad 3 \begin{cases} x = ab' \\ y = a'b \end{cases} \quad 4 \begin{cases} x = b' \\ y = a' \end{cases}$$

Les solutions 1 et 4 correspondent aux matrices de permutation P_1 et P_4 . La solution 3 est la solution particulière donnée par la formule (p).

b) Méthode du cas particulier

Posons

$$x = [\alpha \ \beta \ \gamma \ \delta] A$$

$$y = [\alpha_1 \ \beta_1 \ \gamma_1 \ \delta_1] A$$

A étant la matrice base des données :

$$A = \begin{bmatrix} a & b \\ a & b' \\ a' & b \\ a' & b' \end{bmatrix}$$

$$x y' = [0 \ 1 \ 0 \ 1] \quad A = a b'$$

$$x'y = [0 \ 0 \ 1 \ 0] \quad A = a'b$$

d'où 8 équations à 8 inconnues

$$\alpha \alpha'_1 = 0 \quad \beta \beta'_1 = 1 \quad \gamma \gamma'_1 = 0 \quad \delta \delta'_1 = 0$$

$$\alpha' \alpha_1 = 0 \quad \beta' \beta_1 = 0 \quad \gamma' \gamma_1 = 1 \quad \delta' \delta_1 = 0$$

$$\text{d'où } \alpha_1 = \alpha, \quad \delta_1 = \delta, \quad \beta = \gamma_1 = 1, \quad \beta_1 = \gamma = 0$$

et

$$x = a b' + \alpha a + \delta b'$$

$$y = a'b + \alpha b + \delta a'$$

α et δ paramètres indépendants.

5 - EQUATION FONCTIONNELLE -

$$\Phi[f(x_i, \dots, x_k), \dots, f(x_j, \dots, x_1)] = 1 \quad (19)$$

Φ est une fonction booléenne donnée des fonctions de Boole f portant sur un nombre déterminé n de variables x .

Il s'agit de trouver la forme la plus générale $f(x_1, x_2, \dots, x_n)$ satisfaisant à (19).

On écrira :

$$f = A'X = [\alpha_{2^n-1} \dots \alpha_0] \begin{bmatrix} \bar{x}_1 & x_2 & \dots & x_n \\ \dots & \dots & \dots & \dots \\ x'_1 & x'_2 & \dots & x'_n \end{bmatrix} \quad (20)$$

$$\Phi = B'X = [\beta_{2^n-1} \dots \beta_0] \begin{bmatrix} \bar{x}_1 & x_2 & \dots & x_n \\ \dots & \dots & \dots & \dots \\ x'_1 & x'_2 & \dots & x'_n \end{bmatrix}$$

Les β se calculent facilement en fonction des α d'après (19) et (20). En effet, en considérant Φ comme fonction des $x_i : \Phi(x_1, x_2, \dots, x_n)$, on a :

$$\beta_{2^n-1} = \Phi(1, 1, \dots, 1)$$

.....

$$\beta_0 = \Phi(0, 0, \dots, 0)$$

Pour le calcul de Φ , cette méthode est plus simple que la substitution directe de (20) dans (19) peu praticable dès que Φ est compliquée.

On résout alors par codage le système de 2^n équations à 2^n inconnues α_i :

$$\beta_i(\alpha_{2^n-1}, \dots, \alpha_0) = 1 \quad (i \text{ de } 0 \text{ à } 2^n - 1) \quad (21)$$

d'où $\alpha_j = \varphi(\lambda_1, \dots, \lambda_p)$; λ_k paramètres de Boole indépendants ; d'où la forme la plus générale de f par (20).

Exemple

$$f\{f(y, z), f(x, y), f(x, z)\} = 1$$

En posant

$$\varphi(x, y, z) = f\{f(y, z), f(x, y), f(x, z)\}$$

$$f(x, y) = \alpha xy + \beta xy' + \gamma x'y + \delta x'y'$$

on a le système

- 1) $\varphi(1, 1, 1) = \alpha + \gamma\delta = 1$
- 2) $\varphi(1, 1, 0) = \alpha\delta + \alpha\beta + \beta\gamma' + \beta'\gamma\delta = 1$
- 3) $\varphi(1, 0, 1) = \alpha(\gamma + \beta'\delta) = 1$
- 4) $\varphi(1, 0, 0) = \alpha\delta + \beta\delta + \alpha\beta\gamma = 1$
- 5) $\varphi(0, 1, 1) = \alpha\delta + \alpha\gamma + \gamma\delta + \alpha\beta\gamma' = 1$
- 6) $\varphi(0, 1, 0) = \delta + \alpha\beta + \beta\gamma' = 1$
- 7) $\varphi(0, 0, 1) = \alpha\gamma + \beta\gamma\delta + \beta'\gamma'\delta = 1$
- 8) $\varphi(0, 0, 0) = \delta(\alpha + \beta) = 1$

Les équations 3 et 8 donnent $\alpha = 1$, $\delta = 1$, $\beta' + \gamma = 1$. On vérifie alors que toutes les autres équations sont satisfaites. En résolvant $\beta' + \gamma = 1$, sous la forme :

$$\begin{aligned} \beta &= \lambda \\ \gamma &= \lambda + \mu \end{aligned}$$

on a donc la forme la plus générale de f à l'aide de deux paramètres de Boole indépendants λ, μ ,

$$f = xy + \lambda xy' + (\lambda + \mu) x'y + x'y'$$

Remarque - Il se peut qu'on s'intéresse plus particulièrement aux solutions non dégénérées, c'est-à-dire dépendant de n variables effectives. On ajoutera alors au système (21) une équation exprimant que $f(x_1, x_2, \dots, x_n)$ dépend de n variables effectives.

Il suffit d'écrire pour cela

$$1 = \prod_i [f(x_1, x_2, \dots, x_{i-1}, 0, x_{i+1}, \dots, x_n) \oplus f(x_1, x_2, \dots, x_{i-1}, 1, x_{i+1}, \dots, x_n)]$$

\oplus est le signe d'addition modulo 2,

ce qui donne une condition en α de la forme

$$\sum \left[\prod (\alpha_j \alpha_k + \alpha_j' \alpha_k') \right] = 0$$

Cette condition peut simplifier le système (21) puisqu'elle élimine des solutions parasites.

Exemple - Soit à rechercher les fonctions de Boole à 2 variables effectives, associatives.

L'équation fonctionnelle s'écrit

$$f[x_1, f(x_2, x_3)] = f[f(x_1, x_2), x_3]$$

Le système (21) se réduit ici à

$$\varphi_1 = (\alpha_1 + \alpha_2) (\alpha_0 + \alpha_3') \oplus \alpha_0 \alpha_3' = 0$$

La condition pour que la fonction f soit à 2 variables explicites s'écrit

$$\begin{aligned} \varphi_2 = & \alpha_0' \alpha_1' \alpha_2' \alpha_3' + \alpha_0 \alpha_1 \alpha_2' \alpha_3' + \alpha_0 \alpha_1' \alpha_2 \alpha_3' + \alpha_0' \alpha_1 \alpha_2' \alpha_3 \\ & + \alpha_0' \alpha_1' \alpha_2 \alpha_3 + \alpha_0 \alpha_1 \alpha_2 \alpha_3 = 0 \end{aligned}$$

on a donc à résoudre

$$\varphi_3 \equiv \varphi_1 + \varphi_2 = \alpha_1' \alpha_2 + \alpha_1 \alpha_2' + \alpha_1' \alpha_3' + \alpha_0 \alpha_2 = 0$$

d'où le codage

$$\alpha_0 = \lambda' \mu, \quad \alpha_1 = \lambda, \quad \alpha_2 = \lambda, \quad \alpha_3 = \lambda' + \nu$$

En considérant les 23 valeurs de (λ, μ, ν) , on retrouve les opérations associatives $\cdot, +, \oplus, \equiv$.

REFERENCES

- [1] G. BOOLE - *An investigation of the laws of thought*. Dover Publications U.S.A. (1854).
- [2] M. CARVALLO - *Monographie des Treillis et Algèbre de Boole*, Gauthiers-Villars, Paris (1962).
- [3] M. CARVALLO - *Principes et Applications de l'Analyse Booléenne*. Collection de Mathématiques Economiques, Gauthiers-Villars, Paris (à paraître).
- [4] R. FORTET - *L'Algèbre de Boole et ses applications en Recherche Opérationnelle*, Cahiers du Centre d'Etudes de Recherche Opérationnelle, Bruxelles (1959) n°4, p.5.

Laboratoire d'Econométrie
Institut Henri Poincaré
11, rue Pierre Curie, Paris
Reçu le 15 Mars 1963