UNIUNEA EUROPEANĂ

GUVERNUL ROMÂNIEI

Instrumente Structurale
2007-2013

Platformă de e-learning și curriculă e-content pentru învățământul superior tehnic

Testarea Sistemelor

12. GAT independentă de defecte

GENERAREA AUTOMATĂ A TESTELOR INDEPENDENTE DE DEFECTE

Elemente de calcul diferenţial discret

Utilizarea diferenţei Boole-ene, iniţial în teoria codurilor şi apoi în studiul funcţiilor binare de variabile binare a condus la dezvoltarea unui aparat matematic similar celui din calculul diferenţial clasic. Acest aparat matematic a fost numit datorită acestei similitudini calcul diferenţial discret.

Aplicaţiile calculului diferenţial discret, între altele, sunt deosebit de interesante în teoria testării circuitelor de comutaţie, în optimizarea booleană a reţelelor combinaţionale multi-nivel (calculul termenilor interni nespecificaţi de observabilitate), dar şi în proiectarea şi optimizarea circuitelor VLSI construite cu porţi SAU-EXCLUSIV, în sinteza circuitelor cu bune proprietăţi de testabilitate etc.

1 Definiţii şi notaţii

Fie f o funcţie binară scalară de n variabile binare: $f\colon B^n \rightarrow B$. Domeniul de definiţie al functiei f conţine 2^n vectori, fiecare cu n componente binare, de forma $\mathbf{x} = (x_{n-1}, x_{n-2},...,x_0)$. Vom considera vectorii **x** ordonaţi lexicografic, astfel fiecărui vector **x** îi asociem un număr natural j de forma:

$$j = 2^{n-1}x_{n-1} + 2^{n-2}x_{n-2} + ... +2^0x_0. \tag{1}$$

In relaţia (1) expresia din dreapta semnului egal este calculată ajutorul operatorilor aritmetici obişnuiţi. Este făcută această precizare deoarece în continuare semnul plus şi juxtapunerea vor fi folosite, din motive de simplitate a scrierii, cu sensul de reuniune (disjuncţie sau sumă logică) respectiv intersecţie (conjuncţie sau produs logic), exceptând cazul în care se precizează, aşa cum s-a procedat anterior, explicit un alt sens. Relaţia (1) vine să asocieze unui minterm **x** un număr natural j unic. În continuare, se va utiliza şi corespondeţa inversă: cu ajutorul exponenţierii binare scalare se va asocia unui număr natural un minterm unic.

Definiţia 1. Fie a şi x două variabile binare, atunci *exponenţierea binară* cu baza x şi argument a este dată de expresia:

$$x^{(a)} = x \oplus a' \tag{2}$$

unde semnul $\oplus$ reprezintă suma modulo-2 (disjuncţia exclusivă) iar a' este notaţia pentru a complementat .

Folosind proprietăţile sumei modulo-2, se observă că se poate deduce cu uşurinţă:

$$x^{(1)}=x \text{ şi } x^{(0)}=x'.$$

Utilizând definiţia exponenţierii binare se introduc, în cele ce urmează două notaţii. Astfel, dacă $\boldsymbol{x}$ este o variabilă vectorială cu n ranguri iar j este un număr natural, $0 \leq j \leq 2^n-1$, în baza 2 numărul j se scrie poziţional astfel $\boldsymbol{j} = (j_{n-1}\ j_{n-2}\ ...\ j_0)_2$, atunci *exponenţierea scalară* pentru **x** la puterea **j** este conjuncţia:

$$\mathbf{x}^{(\mathbf{j})} = \bigcap_{k=1}^{n} x_{n-k}^{(j_{n-k})}.$$

Prin această operaţie de exponenţiere scalară, se poate remarca faptul că se pot genera mintermi corespunzători numerelor j.

Spre deosebire de exponenţierea binară scalară care generează un termen produs, un minterm, *exponenţierea binară vectorială* a variabilei **x** la exponentul **j**, produce vectorul notat astfel:

$$(\mathbf{x}^{(\mathbf{j})}) = (x_{n-1}^{(j_{n-1})}, x_{n-2}^{(j_{n-2})}, \ldots, x_0^{(j_0)})$$

cu alte cuvinte vectorul acesta reprezintă punctul din B^n corespunzător numărului natural *j*.

Definţia 2 Fie *a* şi *x* două variabile binare, atunci *exponenţierea normală* în baza *x* şi argument *a* se calculează astfel:
$x^1 = x$, dacă *a* are valoarea 1 şi (3)
$x^0 = 1$, dacă *a* are valoarea 0. (3')

Exponenţierea normală se mai numeşte şi *exponenţiere inelară* deoarece poate fi privită ca o exponenţiere în inelul construit peste mulţimea B cu două elemente şi înzestrată cu o legea aditivă (suma modulo-2) şi cu o lege multiplicativă (intersecţia): B(⊕,•).
Similar exponenţierii binare, se pot introduce alte două notaţii folosind aceleaşi precizări ca şi mai înainte .*Exponenţierea normală scalară* este conjuncţia:

$$\mathbf{x}^{\mathbf{j}} = \bigcap_{k=1}^{n} x_{n-k}^{j_{n-k}}$$

iar *exponenţierea normală vectorială* a variabilei x la exponent j este vectorul:

$$(\mathbf{x}^{\mathbf{j}}) = (x_{n-1}^{j_{n-1}}, x_{n-2}^{j_{n-2}}, \ldots, x_0^{j_0})$$

Exponenţierea normală scalară acţionează ca un filtru taie-toate acele variabile pentru care exponentul este nul, în timp ce exponenţierea normală vectorială calculează un vector ce are o parte din componente nominalizate cu valoarea 1. Exponeţierea normală (inelară) este adesea, din motive de exprimare uşoara, referită (simplu) şi ca exponenţiere.

Dacă **x** este un vector binar din B^5, spre exmplu, atunci $\mathbf{x}^{(10)}=x_4'x_3x_2'x_1x_0'$, adică un minterm, (deoarece $10=(01010)_2$ pe 5 ranguri), iar $(\mathbf{x})^{(10)}=(x_4',x_3,x_2',x_1,x_0')$ şi pentru exponenţierea normală obţinem: $\mathbf{x}^{10}=x_3x_1$ iar $(\mathbf{x})^{10}=(1,x_3,1,x_1,1)$.

2 Derivate discrete

In continuare definiţia 1.8 oferă cea mai simplă metodă de calcul direct a derivatei discrete pentru o funcţie binară de variabile binare.

Definiţia 3 Derivata discretă a funcţiei *f* în raport cu variabila x_i, $0 \leq i \leq n$, este dată de expresia:

$$\frac{df}{dx_i} = f(x_{n-1},\ldots,x_{i+1},1,x_{i-1},\ldots,x_0) \oplus f(x_{n-1},\ldots,x_{i+1},0,x_{i-1},\ldots,x_0)$$

Aplicarea iterativă a derivatei discrete, în raport cu anumite variabile conduce la necesitatea definirii derivatei discrete în raport cu o submulţime a mulţimii variabilelor funcţiei *f*.
Fie $\mathbf{x}=(\mathbf{x}_2,\mathbf{x}_1)$ o partiţie a vectorului **x**, astfel încât $\mathbf{x}_1$ sa conţină primele *t* variabile din **x** începând cu x_0 (eventual renumerotând variabilele).

Definiţia 4. Derivata discretă a funcţiei f în raport cu $\mathbf{x}_1$ este funcţia:

$$\frac{df}{d\mathbf{x}_1} = \frac{d}{dx_0}\left(\frac{d}{dx_1}\left(\ldots\left(\frac{df}{dx_{t-1}}\right)\ldots\right)\right)$$

Derivata discretă are un număr mare de proprietăţi matematice. În continuare sunt enumerate unele dintre cele mai importante proprietăţi ale derivatei discrete şi anume cele legate de domeniul proiectării şi testării subsistemelor digitale .

$$\frac{df}{dx_i} = 0 \Rightarrow f = f(x_{n-1},\ldots,x_{i+1},x_{i-1},\ldots,x_0) \tag{4}$$

$$\frac{d(f')}{dx_i} = \frac{df}{dx_i} \tag{5}$$

$$\frac{df}{dx_i'} = \frac{df}{dx_i} \tag{.6}$$

$$\frac{d(f \oplus g)}{dx_i} = \frac{df}{dx_i} \oplus \frac{dg}{dx_i} \tag{7}$$

$$\frac{d(f \cdot g)}{dx_i} = \frac{df}{dx_i}\cdot g \oplus \frac{dg}{dx_i}\cdot f \oplus \frac{df}{dx_i}\cdot\frac{dg}{dx_i} \tag{8}$$

$$\frac{d(f + g)}{dx_i} = \frac{df}{dx_i} \oplus \frac{dg}{dx_i} \oplus \frac{d(f.g)}{dx_i} \tag{9}$$

$$\frac{d}{dx_i}\left(\frac{df}{dx_i}\right) = 0 \tag{10}$$

$$\frac{d}{dx_i}\left(\frac{df}{dx_j}\right) = \frac{d}{dx_j}\left(\frac{df}{dx_i}\right) \tag{11}$$

3 Dezvoltări canonice disjunctiv-exclusive ale funcţiilor binare de *n* variabile binare

Se consideră dezvoltarea canonică disjunctivă (Shanon) a funcţiei f:

$$f(\mathbf{x}) = f(\mathbf{x}=0)\cdot\mathbf{x}^{(0)} + f(\mathbf{x}=1)\cdot\mathbf{x}^{(1)} + \ldots + f(\mathbf{x}=2^{n-1})\cdot\mathbf{x}^{(2^n-1)} \tag{12}$$

Ţinând cont că termenii în x sunt disjuncţi, relaţia (12) se poate rescrie astfel:

$$f(\mathbf{x}) = f(\mathbf{x}=0)\cdot\mathbf{x}^{(0)} \oplus f(\mathbf{x}=1)\cdot\mathbf{x}^{(1)} \oplus\ldots\oplus f(\mathbf{x}=2^{n-1})\cdot\mathbf{x}^{(2^n-1)} \tag{13}$$

In continuare forma (13) se va numi dezvoltarea canonică disjunctiv-exclusivă a funcţiei *f.*

Pentru o mai bună introducere a unei propoziţii importante ce urmează a fi enunţată, se consideră următorul exemplu:

Exemplul 1 Fie o funcţie binară g de o singură variabilă binară. Aplicând dezvoltarea canonică disjunctiv-exclusivă, pentru acest caz particular şi substituind în dezvoltarea obţinută literalul x' prin $x\oplus 1$, se observă că se poate ajunge la expresia:

$g(x) = g(0) \oplus (g(0)\oplus g(1))x$

Utilizând definiţia derivatei discrete, egalitatea anterioară se poate rescrie (14):

$$g(x) = g(0) \oplus \frac{dg}{dx}x \tag{14}$$

(14).

Printr-un procedeu similar, dar grupând altfel, se poate obţine şi relaţia (15):

$$g(x) = g(1) \oplus \frac{dg}{dx} x' \qquad (15)$$

In relaţiile (14) şi (15) se poate remarca faptul că derivata discretă este o constantă (funcţia g este dependentă doar de o singură variabilă).

Se poate observa că relaţiile (14) şi (15) se pot aplica şi unor funcţii de două sau mai multe variable. In acest caz procedeul se aplică după una (oarecare) dintre variabilele funcţiei, dar derivata discretă calculată nu mai este o constantă.

◊

Relaţiile (14) si (15) se pot denumi, din cauza asemănării evidente, dezvoltări Taylor în 0 şi respectiv în 1 ale funcţiei g. Propoziţia următoare, atribuită lui S.B. Akers, cuprinde generalizarea relaţiilor (14) şi (15). In toate referinţele această teoremă este enunţată fără demonstraţie. In continuare este prezentată o demonstraţie, prin inducţie completă, realizată de autor. Realizarea demonstraţiei a fost necesară în mod deosebit pentru evitarea confuziilor legate de convenţiile de notaţie, adeseori presupuse cunoscute implicit dar, totuşi, variabile de la o şcoală la alta.

Propozitia 1 (S.B. Akers)

Dezvoltarea Taylor a unei funcţii f, într-un punct $\mathbf{k}=(k_{n-1},k_{n-2},...,k_0)$ din domeniul B^n, este dată de relaţia:

$$f(\mathbf{x}) = f(\mathbf{k}) \oplus \Big(\bigoplus_{i=0}^{n} \frac{df}{dx_i}\Big|_{\mathbf{x}=\mathbf{k}} \cdot x_i^{(k_i')}\Big) \oplus$$

$$\Big(\bigoplus_{0<i<j}^{n} \frac{df}{dx_i\ dx_j}\Big|_{\mathbf{x}=\mathbf{k}} \cdot x_i^{(k_i')} x_j^{(k_j')}\Big) \oplus ... \oplus \frac{df}{dx_0\ dx_1 ... dx_{n-1}} \cdot x_0^{(k_0')} x_1^{(k_1')} ... x_{n-1}^{(k_{n-1}')}. \qquad (16)$$

Demonstraţie

Demonstraţia se va face prin inducţie completă după n, numărul de variabile al funcţiei f. Considerând relaţiile (14) şi (15) acestea se pot scrie unitar utilizând exponenţierea binară astfel:

$$f(x) = f(k) \oplus \frac{df}{dx} \cdot x^{(k')}. \qquad \text{(i)}$$

Cu aceasta se poate considera că verificarea propoziţiei pentru n=1 este realizată.

Presupunând, în continuare, că relaţia (16) este verificată pentru orice funcţie f care depinde de cel mult n-1 variabile şi în baza acestei ipoteze se va demonstra că relaţia (16) se verifică şi pentru funcţii cu n variabile. Fie f o funcţie de n variabile şi fie $\mathbf{x} = (x_{n-1}, x_{n-2}, ... ,x_0)$ vectorul variabilelor sale. Fie $\mathbf{k}$ (un punct din B^n) punctul în care se calculează dezvoltarea funcţiei f. Asupra vectorului $\mathbf{x}$ se efectuează o partiţie de forma: $\mathbf{x} = (x_{n-1}, \mathbf{x}_0)$, unde $\mathbf{x}_0$ este un vector cu n-1 componente: $\mathbf{x}_0 = (x_{n-2}, x_{n-3}, ... , x_0)$. Cu această partţie se poate scrie $f(\mathbf{x}) = f(x_{n-1},\mathbf{x}_0)$. Funcţiei f îi se poate aplica o dezvoltare în raport numai cu variabila x_{n-1}, în punctul $x_{n-1} = k_{n-1}$, astfel:

$$f(x_{n-1}, \mathbf{x}_0) = f(k_{n-1}, \mathbf{x}_0) \oplus \frac{df(x_{n-1}, \mathbf{x}_0)}{dx_{n-1}} \cdot x_{n-1}^{(k_{n-1}')}. \qquad \text{(ii)}$$

În relaţia obţinută au apărut doi termeni, fiecare depinzând de n-1 variabile: $f(k_{n-1},\mathbf{x}_0)$ şi $\frac{df(x_{n-1}, \mathbf{x}_0)}{dx_{n-1}}$.

Acestor termeni se poate aplica ipoteza de inducţie într-un punct $\mathbf{k_0}=(k_{n-2}, k_{n-3}, \ldots ,k_0)$ din spaţiul B^{n-1}, astfel încât punctul $\mathbf{k}$ din B^n este $\mathbf{k} = (k_{n-1},\mathbf{k}_0)$:

$$f(k_{n-1},\mathbf{x}_0) = f(k_{n-1},\mathbf{k}_0) \oplus \Big(\bigoplus_{i=0}^{n-2} \frac{df(k_{n-1},\mathbf{x}_0)}{dx_i}\bigg|_{\mathbf{x}_0=\mathbf{k}_0} \cdot x_i^{(k_i^{'})}\Big) \oplus \Big(\bigoplus_{j>i=0}^{n-2} \frac{df(k_{n-1},\mathbf{x}_0)}{dx_i dx_j}\bigg|_{\mathbf{x}_0=\mathbf{k}_0} \cdot x_i^{(k_i^{'})} x_j^{(k_j^{'})}\Big) \oplus \ldots$$

$$\ldots \oplus \frac{df(k_{n-1},\mathbf{x}_0)}{dx_0 dx_1 \ldots dx_{n-2}} \cdot x_0^{(k_0^{'})} x_1^{(k_1^{'})} \ldots x_{n-2}^{(k_{n-2}^{'})}, \tag{iii}$$

respectiv:

$$\frac{df(x_{n-1},\mathbf{x}_0)}{dx_{n-1}} \cdot x_{n-1}^{(k_{n-1}^{'})} = \Bigg(\frac{df(x_{n-1},\mathbf{x}_0)}{dx_{n-1}}\bigg|_{\mathbf{x}_0=\mathbf{k}_0} \oplus \Big(\bigoplus_{i=0}^{n-2} \frac{df(x_{n-1},\mathbf{x}_0)}{dx_{n-1} dx_i}\bigg|_{\mathbf{x}_0=\mathbf{k}_0} \cdot x_i^{(k_i^{'})}\Big) \oplus$$

$$\oplus \Big(\bigoplus_{j>i=0}^{n-2} \frac{df(x_{n-1},\mathbf{x}_0)}{dx_{n-1} dx_i dx_j}\bigg|_{\mathbf{x}_0=\mathbf{k}_0} \cdot x_i^{(k_i^{'})} x_j^{(k_j^{'})}\Big) \oplus \ldots \oplus \frac{df(x_{n-1},\mathbf{x}_0)}{dx_{n-1} dx_0 dx_1 \ldots dx_{n-2}} \cdot x_0^{(k_0^{'})} x_1^{(k_1^{'})} \ldots x_{n-2}^{(k_{n-2}^{'})}\Bigg) x_{n-1}^{(k_{n-1}^{'})}. \tag{iv}$$

Sumând modulo-2 relaţiile (iii) şi (iv), şi ţinând cont că $f(k_{n-1},\mathbf{k}_0) = f(\mathbf{k})$ se obţine, după gruparea convenabilă a termenilor, egalitatea:

$$f(\mathbf{k}) \oplus \frac{df(x_{n-1},\mathbf{x}_0)}{dx_{n-1}} x_{n-1}^{(k_{n-1}^{'})} = f(\mathbf{k}) \oplus \Bigg(\Big(\bigoplus_{i=0}^{n-2} \frac{df(k_{n-1},\mathbf{x}_0)}{dx_i}\bigg|_{\mathbf{x}_0=\mathbf{k}_0} \cdot x_i^{(k_i^{'})}\Big) \oplus \frac{df(x_{n-1},\mathbf{x}_0)}{dx_{n-1}}\bigg|_{\mathbf{x}_0=\mathbf{k}_0} \cdot x_{n-1}^{(k_{n-1}^{'})}\Bigg) \oplus$$

$$\oplus \Bigg(\Big(\bigoplus_{j>i=0}^{n-2} \frac{df(k_{n-1},\mathbf{x}_0)}{dx_i dx_j}\bigg|_{\mathbf{x}_0=\mathbf{k}_0} \cdot x_i^{(k_i^{'})} x_j^{(k_j^{'})}\Big) \oplus \Big(\bigoplus_{i=0}^{n-2} \frac{df(x_{n-1},\mathbf{x}_0)}{dx_{n-1} dx_i}\bigg|_{\mathbf{x}_0=\mathbf{k}_0} \cdot x_{n-1}^{(k_{n-1}^{'})} x_i^{(k_i^{'})}\Big)\Bigg) \oplus \ldots$$

$$\ldots \oplus \frac{df(x_{n-1},\mathbf{x}_0)}{dx_0 dx_1 \ldots dx_{n-1}} \cdot x_0^{(k_0^{'})} x_1^{(k_1^{'})} \ldots x_{n-1}^{(k_{n-1}^{'})}. \tag{v}$$

Acum se observă cu uşurinţă ca relaţia (v) este egală cu relaţia (16) şi demonstraţia este încheiată.

◊

Pentru o mai mare uşurinţă în manevrarea formulei (16) aceasta se poate rescrie, mai compact, folosind ca argument al funcţiei *f* punctul în care se calculează derivata discretă, astfel:

$$f(\mathbf{x}) \;=\; \bigoplus_{\mathbf{j}=0}^{2^n-1} \frac{df(\mathbf{x}=\mathbf{k})}{d\mathbf{x}^{\mathbf{j}}} . (\mathbf{x} \oplus \mathbf{k})^{\mathbf{j}} \tag{17}$$

Folosind forma (17), se pot deduce câteva proprietăţi interesante, în continuare.

Corolarul 1 Fie $\mathbf{x}$ un vector cu *n* componente binare şi fie $\mathbf{x}=(\mathbf{x}_1,\mathbf{x}_0)$ o partiţie a sa unde $\mathbf{x}_0$ cuprinde primele *m* variabile ($m \leq n$) din $\mathbf{x}$ (eventual renumerotând). Dezvoltarea Taylor în raport cu $\mathbf{x}_0$, în jurul unui punct $\mathbf{i}_0$ din spaţiul B^m, pentru funcţia *f* este:

$$f(\mathbf{x}) \;=\; \bigoplus_{\mathbf{j}=0}^{2^m-1} \frac{df(\mathbf{x}_0=\mathbf{i}_0)}{d\mathbf{x}_0^{\mathbf{j}}} \cdot (\mathbf{x}_0 \oplus \mathbf{i}_0)^{\mathbf{j}} \tag{18}$$

Corolarul 2
In relaţia (17) se poate observa că substituind **x** prin **x** ⊕ **k** se obţine:

$$f(\mathbf{x} \oplus \mathbf{k}) = \bigoplus_{\mathrm{j}=0}^{2^n-1} \frac{df(\mathbf{x} = \mathbf{k})}{d\mathbf{x}^{\mathrm{j}}} \cdot \mathbf{x}^{\mathrm{j}} \tag{19}$$

Dacă se însumează modulo-2 relaţiile (17) şi (19), se obţine următorul rezultat:

$$f(\mathbf{x}) \oplus f(\mathbf{x} \oplus \mathbf{k}) = \bigoplus_{\mathrm{j}=0}^{2^n-1} \frac{df(\mathbf{x} = \mathbf{k})}{d\mathbf{x}^{\mathrm{j}}} \cdot \mathbf{k}^{\mathrm{j}} \tag{20}$$

Corolarul 3 Remarcând faptul că relaţia (17) este o identitate în **k** , după o prealabilă interschimbare a variabilei **x** prin **k** , se obţine:

$$f(\mathbf{x}) \oplus f(\mathbf{k}) = \bigoplus_{\mathrm{j}=1}^{2^n-1} \frac{df(\mathbf{x})}{d\mathbf{x}^{\mathrm{j}}} \cdot (\mathbf{k} \oplus \mathbf{x})^{\mathrm{j}} \tag{21}$$

Formele de dezvoltare Taylor mai sunt cunoscute în literatură şi sub numele de *forme canonice Reed-Muller* (RMCF), iar vectorul, punctul, **k** din B^n în jurul căruia se face dezvoltarea este numit *vector de polarizare* .
Se observă cu uşurinţă că atunci când în vectorul **k** componenta de rang *j* este nulă, corespunzător acesteia variabila de rang *j* se dezvoltă în zero şi în consecinţă dezvoltarea Taylor va cuprinde variabila de rang *j* doar în forma directă. Complementar, atunci când în vectorul **k** componenta de rang *j* este nenulă, dezvoltarea Taylor va cuprinde doar forma complementată (negată) a variabilei de rang *j*.

Exemplul 2 Se consideră funcţia:

$$f(x_2,x_1,x_0) = x_0'x_1'x_2' + x_0x_1'x_2' + x_0x_1'x_2 + x_0'x_1x_2 .$$

Se va calcula dezvoltarea disjunctiv-exclusivă (17) a funcţiei pentru punctul **k**=4 (x_2=1, x_1=0 şi x_0=0, deci **x** ⊕ **k** = (x_2', x_1, x_0)). Termenii produs sunt:

j	($j_4 j_2 j_0$)	$(\mathbf{x} \oplus \mathbf{k})^{\mathrm{j}} = (x_2', x_1, x_0)^{\mathrm{j}}$
0	(0 0 0)	1
1	(0 0 1)	x_0
2	(0 1 0)	x_1
3	(0 1 1)	x_1x_0
4	(1 0 0)	x_2'
5	(1 0 1)	$x_2'x_0$
6	(1 1 0)	$x_2'x_1$
7	(1 1 1)	$x_2'x_1x_0$

Funcţia $f(x_2,x_1,x_0)$, în punctul **x** = 4, are valoarea 0.
Determinarea derivatelor discrete conduce la expresiile:

$$\frac{df}{dx_0} = x_2; \frac{df}{dx_1} = 1; \frac{df}{dx_2} = x_0'; \frac{df}{dx_0dx_1} = 0; \frac{df}{dx_0dx_2} = 1; \frac{df}{dx_1dx_2} = 0; \frac{df}{dx_0dx_1dx_2} = 0;$$

După calcularea tuturor coeficienţilor constituiţi din valorile derivatelor discrete în punctul **x**=4, dezvoltarea disjunctiv-exclusivă care se obţine, conform **propoziţiei 1**, este:

$$f(x_2, x_1, x_0) = x_0 \oplus x_1 \oplus x_2' \oplus x_0x_2'.$$

Privitor la calculul coeficienţilor constituiţi din valorile derivatelor discrete se pot face următoarele consideraţii.
Deoarece în dezvoltarea (17) sunt necesare doar valorile derivatelor discrete în punctul **x**=**k**, calculul coeficienţilor dezvoltării se poate desfăşura mult mai eficient (mai ales în calcul automat) pornind de la vectorul de valori al funcţiei *f* (metoda este dezvoltată încontinuare). Calculul formal al derivatelor discrete este fezabil doar pentru exemple de complexitate mică dar are avantajul independenţei de punctul de dezvoltare.

ρ

4 Derivate discrete ale funcţiilor compuse

Considerăm o funcţie binară vectorială **g** de *n* variabile binare:

$$\mathbf{g} : B^n \to B^m$$

$$\mathbf{g}(\mathbf{x}) = (g_{m-1}(\mathbf{x}), g_{m-2}(\mathbf{x}), \ldots, g_0(\mathbf{x})),\ \ \mathbf{x} \in B^n;$$

şi fie *f* o funcţie binară de *m* variabile binare:

$$f : B^m \to B,$$

iar *h* este funcţia compusă

$$h : B^n \to B,$$

$$h(\mathbf{x}) = f \circ \mathbf{g}\ (\mathbf{x}).$$

Propoziţia următoare enunţă modul de derivare discret al funcţiilor compuse. Formula de calcul corespunzătoare, este adeseori numită formula de derivare în lanţ ("*chain rule*") şi este demonstrabilă folosind **Propoziţia 1**.

Propozitia 2 Derivata discretă a funcţiei *h* în raport cu variabila x_i este dată de expresia:

$$\frac{dh}{dx_i} = \Big(\bigoplus_{0\le j}^{m-1} \frac{df}{dg_j}\cdot\frac{dg_j}{dx_i}\Big) \oplus \Big(\bigoplus_{0\le j<k}^{m-1} \frac{df}{dg_j\ dg_k}\cdot\frac{dg_i}{dx_i}\cdot\frac{dg_k}{dx_i}\Big) \oplus \ldots$$

$$\oplus \frac{df}{dg_0\ dg_1\ \ldots\ dg_{m-1}} . \frac{dg_0}{dx_i} . \frac{dg_1}{dx_i} \ldots \frac{dg_{m-1}}{dx_i} \qquad (22)$$

Se poate construi o demonstraţie a formulei de calcul a derivatei compuse dezvoltând disjunctiv-exclusiv (17), într-un punct g, funcţia *h*. Derivând discret în raport cu variabila x_i, în ambii membri dezvoltării şi grupând convenabil se obţine formula de calcul (22).

5 Transformări liniare în calculul diferenţial discret

Calculul practic al dezvoltărilor Taylor într-un punct oarecare **k**, pentru o funcţie dată, se dovedeşte a fi foarte dificil de realizat atunci când se porneşte direct de la definiţie (calculul fiecărei derivate discrete).
Examinând modul în care în Exemplul 1 s-au dedus dezvoltările Taylor pentru o funcţie de o singură variabilă, se pot remarca câteva aspecte definitorii legate de acest subiect:
(i) S-a utilizat pentru început forma canonică disjuntictiv-exclusivă;

(ii) Cele două dezvoltări Taylor (în 0 şi în 1) practic diferă doar prin modul de grupare al termenilor.

Astfel, mai clar, dacă se reliau ecuaţiile din **Exemplul 1** acestea se pot rescrie matricial, peste inelul (B, . , ⊕), astfel :

$$g(x) \;=\; \begin{Vmatrix} g(0) & g(1) \end{Vmatrix} \cdot \begin{Vmatrix} x' \\ x \end{Vmatrix}, \tag{23}$$

$$g(x) \;=\; \begin{Vmatrix} g(0) & \dfrac{dg}{dx} \end{Vmatrix} \cdot \begin{Vmatrix} 1 \\ x \end{Vmatrix} \tag{24}$$

$$g(x) \;=\; \begin{Vmatrix} \dfrac{dg}{dx} & g(1) \end{Vmatrix} \cdot \begin{Vmatrix} x' \\ 1 \end{Vmatrix} \tag{25}$$

iar acum, se observă că în cele trei relaţii (23), (24) şi (25) au apărut trei perechi de funcţii formând tot atâtea baze ale algebrei liniare de funcţii de variabila x : (x', x), $(1, x)$ şi $(x', 1)$. Se va considera, în continuare, drept referinţă baza canonică (x', x). Între aceste trei baze se poate lesne stabili că există următoarele relaţii de transformare:

$$\begin{Vmatrix} x' \\ x \end{Vmatrix} \;=\; \begin{Vmatrix} 1 & 1 \\ 0 & 1 \end{Vmatrix} \cdot \begin{Vmatrix} 1 \\ x \end{Vmatrix}, \tag{26}$$

$$\begin{Vmatrix} x' \\ x \end{Vmatrix} \;=\; \begin{Vmatrix} 1 & 0 \\ 1 & 1 \end{Vmatrix} \cdot \begin{Vmatrix} x' \\ 1 \end{Vmatrix}. \tag{27}$$

Relaţiile (26) şi (27) pot fi interpretate ca fiind scrieri ale dezvoltării Taylor în 0, respectiv în 1, în raport cu baza canonică de mintermi a dezvoltării canonice disjunctiv-exclusive.

Pentru simplitatea scrierii, matricile pătrate din membrul drept al relaţiilor (26) respectiv (27) vor fi notate prin $\mathbf{A}_1$ respectiv $\mathbf{A}_0$.

Remarcăm că între cele două matrice are loc relaţia: $\mathbf{A}_1^{T}=\mathbf{A}_0$, adică transpusa matricii $\mathbf{A}_1$ este egală cu matricea $\mathbf{A}_0$.

Se poate demonstra, imediat prin inducţie completă, următoarea propoziţie:

> **<u>Propozitia 3</u>** Baza canonică de mintermi a dezvoltării canonice disjunctiv-exclusive este dată de relaţia:
>
> $$\mathbf{Z}_n \;=\; \begin{Vmatrix} x'_{n-1} \\ x_{n-1} \end{Vmatrix} * \begin{Vmatrix} x'_{n-2} \\ x_{n-2} \end{Vmatrix} * \ldots * \begin{Vmatrix} x'_0 \\ x_0 \end{Vmatrix}. \tag{28}$$
>
> unde prin * (asterisc) s-a notat produsul tensorial de matrice (iar prin . (punct) s-a notat produsul matricial algebric, obişnuit).

◊

Dacă **A** şi **B** sunt două matrici pătrate de dimensiune n iar **u** şi **v** sunt două matrici coloană de lungime n, se poate verifica egalitatea:

$$(\mathbf{A} \,.\, \mathbf{v}) \;*\; (\mathbf{B} \,.\, \mathbf{u}) \;=\; (\mathbf{A} * \mathbf{B}) \,.\, (\mathbf{v} \;*\; \mathbf{u}). \tag{29}$$

Prin inducţie completă şi ţinând cont de asociativitatea produsului tensorial, relaţia (29) se poate generaliza astfel:

$$(\mathbf{A}_1 . \mathbf{v}_1) * (\mathbf{A}_2 . \mathbf{v}_2) * \ldots * (\mathbf{A}_k . \mathbf{v}_k) \;=\; (\mathbf{A}_1 * \mathbf{A}_2 * \ldots * \mathbf{A}_k) . (\mathbf{v}_1 * \mathbf{v}_2 * \ldots * \mathbf{v}_k). \tag{30}$$

Dacă notăm cele trei baze ale algebrei liniare de funcţii în variabila x_i prin:

$$\mathbf{y}_i = \begin{Vmatrix} x_i' \\ x_i \end{Vmatrix}, \quad \mathbf{y}_{i,0} = \begin{Vmatrix} 1 \\ x_i \end{Vmatrix}, \quad \mathbf{y}_{i,1} = \begin{Vmatrix} x_i' \\ 1 \end{Vmatrix}. \tag{31}$$

atunci putem rescrie relaţia (28) astfel:

$$\mathbf{Z}_n = \mathop{*}_{k=0}^{n-1} \mathbf{y}_k \ . \tag{32}$$

Fie ($\mathbf{x}_1$,$\mathbf{x}_0$) o partiţie a variabilei $\mathbf{x}$ din B^n. Atunci, o dezvoltare Taylor în raport cu variabilele cuprinse în $\mathbf{x}_0$, într-un punct $\mathbf{k}$ din B^r, unde r este numărul de componente din $\mathbf{x}_0$, revine la transformarea bazei canonice de mintermi în conformitate cu bazele corespunzătoare fiecărei variabile x_i. Astfel în acest caz relaţia (31) se scrie:

$$\mathbf{Z}_n = \mathbf{y}_{n\text{-}1} * \mathbf{y}_{n\text{-}2} * \ldots * \mathbf{y}_r * (\mathbf{A}_{k_{r\text{-}1}} \cdot \mathbf{y}_{n\text{-}1,k_{r\text{-}1}}) * \ldots * (\mathbf{A}_{k_0} \cdot \mathbf{y}_{0,k_0}). \tag{32}$$

iar în general, într-un punct $\mathbf{k}$ din B^n, relaţia (32) devine:

$$\mathbf{Z}_n = (\mathbf{A}_{k_{n\text{-}1}} \cdot \mathbf{y}_{n\text{-}1,k_{n\text{-}1}}) * (\mathbf{A}_{k_{n\text{-}2}} \cdot \mathbf{y}_{n\text{-}2,k_{n\text{-}2}}) * \ldots * (\mathbf{A}_{k_0} \cdot \mathbf{y}_{0,k_0}). \tag{33}$$

Folosindu-ne de relaţia (30) putem rescrie (33) astfel:

$$\mathbf{Z}_n = (\mathbf{A}_{k_{n\text{-}1}} * \mathbf{A}_{k_{n\text{-}2}} * \ldots * \mathbf{A}_{k_0}) \cdot (\mathbf{y}_{n\text{-}1,\ k_{n\text{-}1}} * \mathbf{y}_{n\text{-}2,\ k_{n\text{-}2}} * \ldots * \mathbf{y}_{0,k_0}) \tag{34}$$

Aceste rezultate stabilite, se poate considera ca fiind demonstrată propoziţia:

Propoziţia 4 Dezvoltarea Taylor a funcţiei f într-un punct k din B^n este dată de relaţia:

$$f(\mathbf{x}) = \begin{Vmatrix} f(0) & f(1) & \ldots & f(2^n\text{-}1) \end{Vmatrix} \cdot \left(\mathop{*}_{j=n\text{-}1}^{0} \mathbf{A}_{k_j} \right) \cdot \left(\mathop{*}_{j=n\text{-}1}^{0} \mathbf{y}_{j,\ k_j} \right) \tag{35}$$

◊

Propoziţia 4 constituie baza teoretică de calcul a dezvoltării Taylor pentru o funcţie binară dată, într-un punct $\mathbf{k}$ din domeniul său de definiţe, pornind de la o descriere punctuală (tabelară) a funcţiei (prin valorile funcţiei în fiecare din punctele domeniului său de definiţie). Procedeul este implementabil în calcul automat şi aplicabil pentru $n < 20$. Principala limitare a procedeului descris rezidă în faptul că se foloseşte o descriere tabelară completă a funcţiei pentru care se face calculul. Utilizând practic aceleaşi rezultate teoretice se poate utiliza o reprezentare prin cuburi a unei funcţii complet definite şi anume numai cuburile corespunzătoare valorii 1 (o acoperire) a respectivei funcţii.

6 Proprietăţi şi aplicaţii ale dezvoltărilor disjunctiv-exclusive.

Considerând o dezvoltare disjunctiv-exclusivă, de forma (17), a unei funcţii f se poate construi o implementare de forma sume modulo-2 de produse (intersecţii). Astfel de implementări stau la baza sintezei circuitelor combinaţionale folosind circuite SI şi circuite SAU-EXCLUSIV (sumă modulo-2) cu două intrări. Astfel de circuite sunt numite, generic, implementări de *sume exclusive de produse* (SEDP) spre deosebire de implementările clasice în două nivele, care sunt numite, deasemenea generic, implementări de *sume de produse* (SDP). Funcţiile de paritate de n variabile sunt implementabile prin SEDP având n termeni produs în timp ce aceleaşi funcţii implementate prin SDP necesită 2^{n-1} termeni produs. Au fost realizate experimente folosind funcţii generate aleator. Aceste experimente au dovedit că în medie SEDP necesită mai puţini termeni produs decât SDP. Există însă şi excepţii, care chiar dacă sunt relativ izolate sunt totuşi de luat în seamă. Astfel, există o funcţie de $2n$ variabile pentru care sunt necesari 2^n-1 termeni produs în SEDP în timp ce pentru SDP necesită doar n termeni produs. Pornind ca referinţă de la formula (17), în literatură sunt considerate în total 7 clase de expresii SI-EXOR, respectiv 7 clase de circuite:

(i) SEDP *cu polaritate pozitivă* (dezvoltare doar în zero, pentru toate variabilele). Această formă este canonică (unică pentru o funcţie precizată) deoarece este fixat punctul de dezvoltare, deci nu se poate minimiza. Pentru o funcţie de n variabile, numărul mediu de termeni produs este 2^{n-1}.

SEDP *cu polaritate mixtă dar unică* (corespunzător (17)). Există 2^n astfel de forme pentru o funcţie cu n variabile. Problema minimizării se pune în termenii găsirii acelui punct k în jurul căruia să se facă dezvoltarea astfel încât numărul de termeni produs să fie minim. Sunt cunoscute două categorii de metode de minimizare ale SEDP cu polaritate mixtă dar unică. Astfel, pentru o funcţie cu n variabile, metodele de minimizare din prima categorie au o complexitate a spaţiului de memorie şi a timpului de calcul $O(2^n)$ respectiv $O(2^{2n})$, în timp ce metodele de minimizare din a doua categorie necesită spaţiu de memorie şi timp de calcul $O(3^n)$. Există dezvoltate un număr important de programe de minimizare pentru astfel de circuite. Din cauza complexităţii algoritmilor aplicaţiile acestor programe se mărginesc la funcţii având pâna la 15 variabile. Eforturile de cercetare din ultimii ani s-au concentrat asupra minimizării numărului de termeni produs pentru funcţiile de variabile multi-valorice incomplet specificate (subiect teoretic foarte avansat) şi asupra funcţiilor simetrice clasice. Funcţiile simetrice sunt o categorie foarte mult aplicată în circuitele combinaţionale din co-procesoare (circuite aritmetice), în circuitele corectoare de erori, în circuitele combinaţionale utilizate în telecomunicaţii etc. Programul Sympathy, în conformitate cu rezultatele publicate, are bune performanţe reuşind minimizări ale unor funcţii cu până la 30 de variabile într-un timp absolut rezonabil (circa 2 secunde, pe o staţie de lucru Sun Spark 1+).

Printre alte proprietăţi la circuitelor implementând SEDP se numără şi proprietăţile de testare simplă.

(iii) *expresii Kronecker* (în care pentru fiecare variabilă sunt efectuate dezvoltări de tip Taylor fie în zero fie în unu, sau dezvoltări exclusiv-disjunctive). Astfel de expresii au primit numele Kronecker de la faptul că pot fi reprezentate prin produsele cu acelaşi nume. Pentru o funcţie de n variable există 3^n expresii Kronecker. Se cunoaşte un algoritm de minimizare pentru numărul de termeni produs din astfel de expresii. Algoritmul are o complexitate $O(n3^n)$ a memoriei ocupate şi o complexitate a timpului de execuţie $O(3^n)$.

(iv) *pseudo* SEDP *cu polaritate mixtă dar neunică*. Pentru o funcţie cu n variabile există, în total, 2^{2^n-1} astfel de forme distincte. Nu există un interes deosebit, până în prezent, pentru aceste expresii.

(v) *pseudo-expresii Kronecker* (dezvoltările se fac arbitrar, nesistematic) ;
expresii corespunzătoare unor dezvoltări de tip Taylor generalizate. Mai sunt cunoscute şi sub denumirea de forme canonice inconsistente (T.Sasao). Aceste expresii sunt caracterizate prin faptul că permit ambele polarităţi pentru o aceeaşi variabilă. Unei funcţii de *n* variabile se pot asocia $2^{n2^{n-1}}$ astfel de expresii. Există un procedeu de minimizare heuristică al acestor expresii aparţinând unui grup de cercetători condus de M. A. Perkowski.
Dacă termenii produs sunt arbitrari (sume neexclusive de produse) atunci se obţine cea mai generală formă de SEDP. Nu sunt, încă, cunoscute metode eficiente de minimizare a acestor expresii şi se folosesc metode iterative pentru obţinerea unor soluţii minimale. In literatură este citat un algoritm exact, aparţinând grupului de cercetare condus de M. A. Perkowski. Din nefericire acest algoritm este intens consumator de resurse, motiv pentru care este aplicabil doar în cazuri de complexitate foarte redusă. Combinarea minimizării clasice multi-nivel (instrumentul interactiv MIS II sau, mai nou, SIS-1.2) cu unele procedee heuristice iterative de minimizare a SEDP (KGPXORMIN) a condus la crearea instrumentului KGPMIN. KGPMIN este un instrument interactiv eficient cu performanţe deosebite. Astfel, comparativ cu varianta clasică MISII/SIS, instrumentul KGPMIN reuşeşte, în medie, să sintetizeze circuite având aria cu 45,16% mai mică, iar comparativ cu programul de minimizare iterativă a SEDP, KGPXORMIN, rezultatele arată o scădere a suprafeţei, în medie, cu 34,32%. Evaluările au fost făcute pe o staţie de lucru Sun3/280 operând la 16.67 MHz şi cu o memorie de 16MB. A fost utilizat benchmark-ul de circuite combinaţionale multi-nivel de la MCNC.