Test exhaustif de circuits combinatoires

Exhaustive testing of combinatorial circuits

Gérard COHEN

ENST, 46, rue Barrault, 75013 PARIS

Professeur à l'École Nationale Supérieure des Télécommunications en théorie de l'information et codes correcteurs d'erreurs. Ses activités de recherche se situent essentiellement dans ces domaines, ainsi que dans des problèmes de mathématiques discrètes et de combinatoire émanant des communications numériques (cryptographie, test de circuits, ...).

Philippe GODLEWSKI

ENST, Département Systèmes et communications, 46, rue Barrault, 75013 PARIS

Enseignant-chercheur au département Systèmes et Communications de l'E.N.S. Télécommunications, Domaines d'enseignement : communications numériques, théorie de l'information. Domaines de recherches : codes correcteurs d'erreurs, cryptographie.

M. KARPOVSKY

Université de Boston, Département de Génie Électrique, BOSTON, USA

Professeur à l'Université de Boston (USA), sa spécialité le conduit à s'intéresser à la correction d'erreurs et aux systèmes tolérant les pannes.

RÉSUMÉ

Nous présentons une méthode de construction de tableaux dits *s*-surjectifs qui permettent de tester exhaustivement tout ensemble de *s* entrées d'un circuit combinatoire. La méthode est basée sur l'emploi de codes linéaires, ce qui assure la simplicité de sa mise en œuvre. La taille (nombre de tests) des tableaux obtenus se rapproche du minimum $f(n, s)$ pour certaines valeurs des paramètres n (nombre total d'entrées du circuit) et s utiles en pratique.

MOTS CLÉS

Test de VLSI, codes correcteurs d'erreurs.

SUMMARY

We present a method for the construction of s-surjective arrays, which allows exhaustive testing of any set of s inputs in a combinatorial device. The method is based upon the use of linear codes, which implies simplicity of implementation. The size (number of tests) of the obtained arrays is close to the minimum $f(n, s)$ for values of the parameters n (total number of inputs) and s useful in practice.

KEY WORDS

VLSI testing, error-correcting codes.

TABLE DES MATIÈRES

Introduction

Avec l'augmentation du nombre de circuits contenus sur une « puce » VLSI, les problèmes de tests deviennent de plus en plus difficiles. En particulier, l'approche classique, dans laquelle l'ensemble des tests à exercer dépend d'une analyse préalable du circuit, atteint un coût et une complexité inacceptables. Une solution possible à ce problème est la construction de tests exhaustifs relativement à chaque sortie, couplée à des techniques de partitionnement destinées à limiter le nombre d'entrées dont chacune de ces sorties est fonction.

Définitions : Nous appellerons matrice $T(n, s)$ une matrice binaire à n colonnes (les sorties) et telle que dans tout ensemble ordonné de s colonnes, les 2^s s-uples binaires possibles apparaissent au moins une fois. Les lignes de $T(n, s)$ seront dénommées tests.

Le nombre minimal de lignes de $T(n, s)$ sera dénoté par $f(n, s)$. $T(n, s)$ est également désignée par : *tableau binaire s-surjectif* [1]. Avec les hypothèses de l'introduction, chaque sortie dépendant d'au plus s entrées sera testé relativement au 2^s valeurs possibles des entrées. Tout type d'erreur permanente occasionnant une modification de la table de vérité de fonction de sortie sera détecté.

Exemple 1 : $n=3$, $s=2$:

$$T(3, 2)=\begin{matrix}1&1&0\\1&0&1\\0&1&1\\0&0&0\end{matrix}$$

Le problème est résolu pour $s=2$ [2]. Pour $s\geqq 3$, on dispose de constructions récursives [5] sous optimales, c'est-à-dire nécessitant plus de $f(n, s)$ tests. On ne connaît d'ailleurs que des encadrements pour $f(n, s)$.

1. Tests linéaires

Dans ce qui suit, nous considérons la question de la génération simple de ces tests. Une première approche est celle des tests linéaires, dans laquelle les lignes de $T(n, s)$ forment un sous-espace vectoriel.

Rappelons que l'on note $C(n, k, d)$ un sous-espace vectoriel de dimension k de F_2^n, ensemble des suites binaires de taille n, dans lequel deux vecteurs quelconques (ou mots) diffèrent en au moins d composantes.

L'orthogonal de C pour le produit scalaire

$$\langle x, y\rangle=\sum_{i=1}^{n} x_i y_i \bmod 2,$$

où :

$$x=(x_i)_{i=1}^n, y=(y_i)_{i=1}^n$$

sont des vecteurs de F_2^n, est noté $\bar{C}$ et a pour paramètres $(n, n-k, d)$ [3].

Proposition 1 : *Soit* C *un code de paramètres* (n, k, d). *Alors les* 2^{n-k} *mots de* $\bar{C}$ *sont les lignes d'une matrice de test* $T(n, d-1)$.

Preuve : $\bar{C}$ possède en fait une propriété plus forte : c'est un tableau orthogonal de force $d-1$, c'est-à-dire que dans tout $(d-1)$-uple de colonnes, tous les $(d-1)$-uplets binaires apparaissent exactement $2^{n-k-d+1}$ fois [3].

Dans l'exemple 1, on a pris $C(3, 1, 3)$. Les tests linéaires sont de construction simple, mais minimaux seulement dans les cas triviaux $(s=n, n-1, 1)$.

Par exemple la classe des codes BCH $(2^m-1, 2^m-1-mt, 2t+1)$ conduit à des matrices $T(n, s)$ ayant $\sim n^{s/2}$ tests, ce qui trop grand, comparé à $f(n, s)\simeq 2^s \log n$ obtenu par des méthodes non constructives [2], et à $\log n^{2\log s}$ [5].

2. Concaténation de tests linéaires

L'idée est de construire un tableau dont l'ensemble des lignes soit une union de codes linéaires de façon à conserver la simplicité de génération des lignes, tout en améliorant les performances (en augmentant s par exemple). Un tel tableau sera dit *concaténé des codes linéaires*. Nous dirons qu'un s-uple de colonnes d'un tableau est *mauvais* s'il ne contient pas tous les s-uples binaires au moins une fois comme lignes. Par exemple, dans :

$$\begin{matrix}0&1&0\\1&0&1\\0&1&1\\0&0&0\end{matrix}$$

obtenu à partir de l'exemple 1 en changeant le premier bit du premier test, le couple des deux premières colonnes est mauvais, car ne contenant pas (1 1). Les colonnes sont indicées par leur position i, $1 \leqq i \leqq n$.

Reformulons à présent la proposition 1.

Proposition 2 : *Le tableau $2^{nk} \times n$ constitué des mots du code $\bar{C}(n, n\text{-}k)$ contient un mauvais s-uple de colonnes $P=\{p_{i_1}, p_{i_2}, \ldots, p_{i_s}\}$ si et seulement si il existe un mot non nul de C dont le support soit contenu dans l'ensemble $\{i_1, i_2, \ldots, i_s\}$ de positions.*

Preuve : Soit :

$$x \in F_2^n,\ x=(x_1, x_2, \ldots, x_n),\quad \mathrm{supp}(x)=\{i,\ x_i=1\},$$

et P_i les colonnes de $\bar{C}$. $|\mathrm{supp}(x)|$ est le poids de x, noté $|x|$.

Alors, d'après la définition de la dualité,

$$x \in C \Leftrightarrow \sum_{i,\ x_i=1} P_i = 0$$

$$\Rightarrow \text{ rang } \{P_i,\ i \text{ tel que } x_i=1\} < |x|$$

$$\Rightarrow \bar{C} \text{ contient un mauvais } |x|\text{-uple de colonnes.}$$

Réciproquement, s'il existe un mauvais s-uple de colonnes $P=\{P_1, P_2, \ldots, P_s\}$, alors $\mathrm{rang}(P) \leqq s\text{-}1$, il existe $S \subset P$ tel que $\sum_{i \in s} P_i = 0$, et S est le support d'un mot de C.

Soit maintenant $\bar{C}_i(n, n\text{-}k_i)$, $i=1, 2, \ldots, R$, R codes linéaires. Notons T leur concaténé, de dimensions $\sum_{i=1}^{R} 2^{nk_i} \times n$. Soient :

$$x=(x_i)_{i=1, 2, \ldots, n} \qquad y=(y_i)_{i=1, 2, \ldots, n}$$

deux éléments de F_2^n, notons :

$$x \vee y = (x_i \vee y_i)_{i=1, 2, \ldots, n}$$

Ici $x_i \vee y_i$ est le « ou » logique. Alors la proposition 2 s'étend à :

Proposition 3 : *Si le concaténé* T *des* R *codes* $\bar{C}_i(n, n\text{-}k_i)$, $i=1, 2, \ldots, R$ *contient un mauvais s-uple de colonnes* P, *alors il existe* R *vecteurs non nuls* $c_i \in C_i$ *pour* $i=1, 2, \ldots, R$ *tels que :*

$$\mathrm{supp}(c_i \vee c_2 \vee \ldots \vee c_R) \subset P.$$

Preuve : Un mauvais s-uple pour T doit être mauvais pour tous les $\bar{C}_i$, impliquant l'existence des c_i d'après la proposition 2 tels que $\mathrm{supp}(c_i) \subset P$. D'où $\cup \mathrm{supp}(c_i) = \mathrm{supp}(c_1 \vee c_2 \vee \ldots \vee c_R) \subset P$.

La réciproque est fausse, un s-uple mauvais pour tous les $\bar{C}_i$ ne l'étant pas nécessairement pour T.

Soient maintenant $C_1, C_2, \ldots, C_R$ des codes (n, k_i).

Posons :

$$r = \min \left| \bigvee_{i=1}^{R} c_i \right|.$$

Le minimum étant pris lorsque les c_i parcourent les $C_i \setminus \{0\}$, $1 \leqq i \leqq R$.

Alors la concaténée des $\bar{C}_i$ est $(r\text{-}1)$-surjective et :

Corollaire : $f(n, r\text{-}1) \leqq \sum_{i=1}^{R} 2^{nk_i}$.

3. Le cas de deux codes

Supposons dans ce paragraphe $R=2$. Notons $\lceil\ \rceil$ la partie entière par excès.

L'égalité :

$$|c_1 \vee c_2| = \frac{1}{2}[|c_1| + |c_2| + d(c_1, c_2)]$$

est immédiate. Distinguons deux possibilités :

– si $c_1=c_2=c$, alors $c \in C_1 \cap C_2$ et

$$|c_1 \vee c_2| \geqq d(C_1 \cap C_2),$$

où $d(C)$ désigne la distance minimale du code C;

– si $c_1 \neq c_2$, alors :

$$|c_1 \vee c_2| \geqq \frac{1}{2}[d(C_1) + d(C_2) + d(C_1 \oplus C_2)],$$

où :

$$C_1 \oplus C_2 = \{c,\ \exists c_1 \in C_1,\ \exists c_2 \in C_2,\ c=c_1+c_2\}.$$

Nous avons ainsi montré :

Proposition 4 : *Le tableau concaténé de* $\bar{C}_1(n, n\text{-}k_1)$ *et* $\bar{C}_2(n, n\text{-}k_2)$, *duaux de* $C_1(n, k_1, d_1)$ *et* $C_2(n, k_2, d_2)$ *respectivement, est* $(r\text{-}1)$*-surjectif, avec :*

$$r \geqq \min\left\{ d(C_1 \cap C_2),\ \frac{1}{2}[d_1 + d_2 + d(C_2 \oplus C_2)] \right\}$$

Exemple 2 : $n=31$, $k_1=k_2=10$.

Soit α une racine primitive 31-ième de l'unité, de polynôme minimal $m_\alpha(x)=x^5+x^2+1$. Notons $m_{\alpha^i}(x)$ le polynôme minimal de α_i. Prenons pour C_1 et C_2 les codes cycliques engendrés par

$$g_1(x) = \frac{x^{31}-1}{m_\alpha(x)\, m_{\alpha^3}(x)}$$

et

$$g_2(x) = \frac{x^{31}-1}{m_{\alpha^3}(x)\, m_{\alpha^5}(x)},$$

respectivement. Alors C_1 et C_2 ont pour paramètres (31, 10,12) (*cf.* [4]),

$C_1 \cap C_2$ est un code simplexe (31, 5, 16).

$C_1 \oplus C_2$ est un code de paramètres (31, 15, 8).

La concaténée de $\bar{C}_1$ et de $\bar{C}_2$ est un tableau 15-surjectif d'après la proposition 4. En supprimant dans T les tests apparaissant deux fois, c'est-à-dire les mots de $\bar{C}_1 \cap \bar{C}_2$ qui est un (31, 16) code, on obtient un tableau à 2^{22}-2^{16} lignes, généré à partir de $g_1(x)$ et $g_2(x)$, et proche de la taille minimale $f(31, 15)$.

Remarque : En concaténant trois codes $\bar{C}_1$, $\bar{C}_2$, $\bar{C}_3$ de paramètres (31, 21), avec $g_1(x)$, $g_2(x)$ comme plus haut et

$$g_3(x) = \frac{x^{31}-1}{m_a(x)\, m_{a^3}(x)},$$

on construit un tableau 16 surjectif.

Exemple 3 : $n=7$, $s=3$.

C_1 et C_2 sont deux codes cycliques (7, 4, 3) de Hamming, engendrés par $g_1(x)=(X^3+X+1)$ et $g_2(x)=(X^3+X^2+1)$ respectivement. Un calcul simple donne $r=4$. Concaténant $\bar{C}_1$ et $\bar{C}_2$ et supprimant la ligne de $\bar{C}_1+\bar{C}_2$ et supprimant la ligne de $\bar{C}_1 \cap \bar{C}_2=\{0\}$, on obtient une matrice T(7, 3) à 15 lignes :

$$T(7,3) = \begin{bmatrix} 1&1&1&0&1&0&0 \\ 0&1&1&1&0&1&0 \\ 0&0&1&1&1&0&1 \\ 1&0&0&1&1&1&0 \\ 0&1&0&0&1&1&1 \\ 1&0&1&0&0&1&1 \\ 1&1&0&1&0&0&1 \\ 0&0&0&0&0&0&0 \\ 0&1&1&1&0&0&1 \\ 1&0&1&1&1&0&0 \\ 0&1&0&1&1&1&0 \\ 0&0&1&0&1&1&1 \\ 1&0&0&1&0&1&1 \\ 1&1&0&0&1&0&1 \\ 1&1&1&0&0&1&0 \end{bmatrix} \begin{matrix} \\ \\ \\ \\ \\ \\ \\ \\ \\ \\ \\ \\ \\ \\ \end{matrix} \begin{matrix} \bar{C}_1 \\ \\ \\ \\ \\ \\ \\ \bar{C}_2 \\ \\ \\ \\ \\ \\ \end{matrix}$$

On vérifie que trois colonnes quelconques de T(7, 3) contiennent au moins une fois les huit triplets binaires possibles.

Exemple 4 : $n=7$, $s=4$.

C'est le dual du précédent; on concatène C_1 et C_2, obtenant de façon analogue un tableau T(7, 4) à 30 lignes.

Conclusion

Nous avons présenté ici une méthode de construction de tableaux testant exhaustivement tout ensemble de s entrées d'un circuit combinatoire. Elle est basée sur l'emploi de codes linéaires. La taille (nombre de test) des tableaux obtenus se rapproche du minimum $f(n, s)$ pour certaines valeurs des paramètres n et s utiles en pratique. Le caractère linéaire des codes assure la simplicité de la mise en œuvre. Il suffit en effet de stocker les matrices génératrices (ou le polynôme générateur dans le cas cyclique) des codes qui interviennent dans la construction et non le tableau dans sa totalité.

BIBLIOGRAPHIE

[1] A. Chandra, L. Kou, G. Markovsky et S. Zaks, *On sets of boolean n-vectors with all k-projections surjective*, IBM Research Report RC 8936, July 1981.

[2] J. Kleitman et J. Spencer, Families of k-independent sets, *Discrete Math.*, 6, 1973, p. 255-262.

[3] J. Macwilliams et N. J. A. Sloane, *The Theory of Error Correcting Codes*, North-Holland, 16, 1977.

[4] D. V. Sarvate et M. B. Pursley, Correlation properties of pseudorandom and related sequences, *Proc. of the IEEE*, 68, n° 5, May 1980.

[5] D. T. Tang et C. L. Chen, Iterative Exhaustive Pattern Generation for Logic Testing, *IBM J. Res. and Devel.*, 28, n° 2, March 1984.