多値論理研究ノート　第２９巻　第10号　2006年8月
Note on Multiple-Valued Logic in Japan
Vol.29, No.10, August 2006

BMDを用いた初等関数の表現法と数値計算回路への応用

Representations of Elementary Functions Using BMDs and Their Applications to Numerical Function Generators

永山　忍[1]　笹尾　勤[2]
Shinobu NAGAYAMA[1]　Tsutomu SASAO[2]
[1]広島市立大学 情報科学部 情報工学科
[2]九州工業大学 情報工学部 電子情報工学科
[1]Dept. of Computer Engineering, Hiroshima City University
[2]Dept. of Computer Science and Electronics, Kyushu Institute of Technology

概要　本稿は, BMD (Binary Moment Diagram)を用いて多項式関数や三角関数などの初等関数を表現する．本稿では, k 次多項式関数を表現するBMDおよびMTBDD (Multi-Terminal Binary Decision Diagram)の節点数を解析的に求め, BMDの節点数がMTBDDより少ないことを示し，実験により, BMDは，様々な初等関数をMTBDDの29%の節点数で表現できることを示す．また, BMDを用いた数値計算回路の構成法を示す.

1　はじめに

BDD (Binary Decision Diagram) [1,11]は，最もポピュラーな決定グラフであり，論理合成，形式的検証，論理シミュレータなどの様々なアプリケーションで応用されている．BDDは，多くの実用的な論理関数をコンパクトに表現できるが，乗算器を効率よく表現できないという欠点を持つ．このような整数関数をコンパクトに表現するために，ACDD (Arithmetic transform Decision Diagram)[16]，BMD (Binary Moment Diagram)[2]，*BMD (Multiplicative BMD)[2]，K*BMD (Kronecker Multiplicative BMD)[7]，TED (Taylor Expansion Diagram)[3]が提案されている．与えられた整数関数をシャノン展開で表現する MTBDD (Multi-Terminal BDD) [4]に対し，ACDDやBMDは，整数関数を算術展開で表現する．算術展開に基づくこれらの決定グラフは，乗算器などの整数関数を入力数 n の多項式サイズで表現できる[2,16]．本稿では，実数の初等関数[13]を整数関数に帰着し，その整数関数をBMDで表現する．定理と実験により，BMDは実数の初等関数もコンパクトに表現できることを示し，BMDを用いた数値計算回路の構成法を示す．

2　用語の定義

定義 1. $B = \{0, 1\}$, $P = \{0, 1, \ldots, p-1\}$, $p \geq 2$ としたとき, n入力m出力**論理関数**（**多出力論理関数**）は写像$B^n \rightarrow B^m$, **整数関数**は写像$B^n \rightarrow P$である．また, Rを実数の集合としたとき, **実数関数**は写像$R \rightarrow R$である．

定義 2. 二進**固定小数点**で表現された数値 r を

$$d_{s-1}\, d_{s-2}\, \ldots\, d_0.\, d_{-1}\, d_{-2}\, \ldots\, d_{-t}$$

と表記する．ただし, $d_i \in \{0, 1\}$，sは整数部のビット数，tは小数部のビット数である．このときrは，以下の式で計算できる．

$$r = -2^{s-1} d_{s-1} + \sum_{i=-t}^{s-2} 2^i d_i$$

本稿では，負数を2の補数で表現するため，特に指定がない限り, s は符号ビットを含む.

定義 3. **精度**は，実数計算における有効桁数を意味する．特に，**n ビット精度**とは，固定小数点表現された実数計算において有効桁数が n ビット，すなわち, $n = s + t$ を意味する．本稿で, n ビット精度関数$f(X)$は，変数Xがnビット精度であることを意味する．

固定小数点表現により, nビット精度の実数関数は, n入力m出力論理関数とみなすことができ，この多出力論理関数は, mビット出力の二進ベクトルを整数とみなすことで，整数関数に帰着できる．即ち，nビット精度の実数関数は，整数関

数$B^n \rightarrow P$に帰着できる．ただし, $P = \{0, 1, \ldots, 2^m - 1\}$．本稿では，実数の初等関数を$n$ビット精度の固定小数点で表現することにより，整数関数に帰着する．以下では，特に断りがない限り，初等関数は整数関数に帰着されているものとする．

定義 4. A, Bをn次正方行列とし，AとBの**クロネッカー積**を以下のn^2次正方行列で定義する．

$$A \otimes B = \begin{bmatrix} a_{11}B & a_{12}B & \cdots & a_{1n}B \\ a_{21}B & a_{22}B & \cdots & a_{2n}B \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1}B & a_{n2}B & \cdots & a_{nn}B \end{bmatrix}$$

ただし,

$$A = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{bmatrix}$$

3 算術変換

本節では，算術変換および算術スペクトラムについて簡単に述べる．詳細は, [16]を参照されたい.

定義 5. 算術変換行列 $A(n)$を

$$A(n) = \bigotimes_{i=1}^{n} A(1), \quad A(1) = \begin{bmatrix} 1 & 0 \\ -1 & 1 \end{bmatrix}$$

と定義すると，整数関数fの関数ベクトルFの**算術スペクトラムA_f** $= [a_0, a_1, \ldots, a_{2^n-1}]^t$は，以下で与えられる．

$$A_f = A(n)F$$

スペクトラムの各要素a_iを**算術係数**と呼ぶ.

例 1. 整数関数$f(x_1, x_2) = x_1 + x_2$の関数ベクトル$F = [0, 1, 1, 2]^t$であり，その算術スペクトラムは,

$$A_f = A(2)F = \begin{bmatrix} 1 & 0 & 0 & 0 \\ -1 & 1 & 0 & 0 \\ -1 & 0 & 1 & 0 \\ 1 & -1 & -1 & 1 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \\ 1 \\ 2 \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \\ 1 \\ 0 \end{bmatrix}$$

となる． **(例終り)**

算術スペクトラムから，逆算術変換により関数ベクトルを復元できる．

定義 6. 逆算術変換行列$A^{-1}(n)$は，以下で与えられる．

$$A^{-1}(n) = \bigotimes_{i=1}^{n} A^{-1}(1), \quad A^{-1}(1) = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix}$$

$A^{-1}(n)$は，Reed-Muller変換行列と同じ行列であるため，整数Reed-Muller行列とも言われる．

定義 7. 変数x_iを用いて

$$A^{-1}(1) = \begin{bmatrix} 1 & x_i \end{bmatrix}$$

とすると，**逆算術変換**は，以下のように定義される．

$$f = X_a A_f, \quad X_a = \bigotimes_{i=1}^{n} \begin{bmatrix} 1 & x_i \end{bmatrix}$$

例 2. 例 1.で得た算術スペクトラムから逆算術変換を行い，元の関数を得る．

$$f = X_a A_f = \begin{bmatrix} 1 & x_2 & x_1 & x_1 x_2 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \\ 1 \\ 0 \end{bmatrix} = x_1 + x_2$$

(例終り)

定義 8. $A^{-1}(1)$と$A(1)$を用いて整数関数は，次のように表現できる．

$$f = A^{-1}(1)A(1)F = \begin{bmatrix} 1 & x_i \end{bmatrix} \begin{bmatrix} 1 & 0 \\ -1 & 1 \end{bmatrix} \begin{bmatrix} f_0 \\ f_1 \end{bmatrix}$$
$$= \begin{bmatrix} 1 & x_i \end{bmatrix} \begin{bmatrix} f_0 \\ f_1 - f_0 \end{bmatrix} = f_0 + x_i (f_1 - f_0) \qquad (1)$$

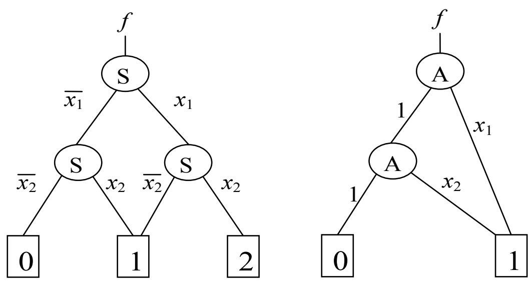
ただし, $f_0 = f(x_i = 0)$, $f_1 = f(x_i = 1)$. (1)を**算術展開**と呼び，算術展開によって得られた式を**算術式**と呼ぶ．算術変換によって得られた算術係数が，この算術式の係数に対応する．

4 BMD (Binary Moment Diagram)

定義 9. 整数関数fに算術展開$f = f_0 + x_i (f_1 - f_0)$を再帰的に繰り返し適用することにより，**BMT (Binary Moment Tree)**を得る．このBMTに，以下の二つの簡単化規則を適用することで**BMD (Binary Moment Diagram)**を得る．

1. 等価な部分グラフを共有する．
2. ある非終端節点vのx_iに対応する枝が，終端節点 0 を指すときその節点vを削除する．

例 3. 図 1(a)は，例 1.の整数関数 f の MTBDD，図 1(b)は，f のBMDを表す．図中で, Sのラベルが付いた節点はシャノン展開を表し, A は，算術展開を表す． **(例終り)**



(a) f の MTBDD.　(b) f の BMD.

図 1. $f = x_1 + x_2$ のMTBDDとBMD.

MTBDDの終端節点は，関数ベクトルFを表現しているのに対し，BMDの終端節点は，Fの算術スペクトラムA_fを表現している．そのため，MTBDDの終端節点数は，異なる関数値の数に等しいのに対し，BMDでは，異なる算術係数の数に等しい．

X^kおよびk次多項式では，精度nと次数kの値から非零算術係数の数とBMDの節点数を解析的に求めることができる．以下の補題と定理で，私たちは，nビット精度X^k関数におけるBMDの非終端節点数およびk次多項式関数におけるBMDの節点数の上界を示す．

補題 1. nビット精度の$f(X) = X^k$における，非零算術係数の数は，次の式で得られる．

$$\sum_{i=1}^{k} \binom{n}{i}$$

(証明) 付録参照．

補題 2. nビット精度k次多項式関数$f(X) = c_0 + c_1X + \ldots + c_kX^k$の非零算術係数の数は，高々

$$\sum_{i=0}^{k} \binom{n}{i}$$

(証明) 付録参照．

補題 3. nビット精度k次多項式関数において，$c_i > 0$ かつ$X \geq 0$ のとき，非零算術係数の数は

$$\sum_{i=0}^{k} \binom{n}{i}$$

(証明) 付録参照．

補題 4. nビット精度の$f(X) = X^k$を表現するBMDの非終端節点数は

$$\alpha(n,k) = \sum_{i=1}^{k} \binom{n}{i}$$

(証明) 付録参照．

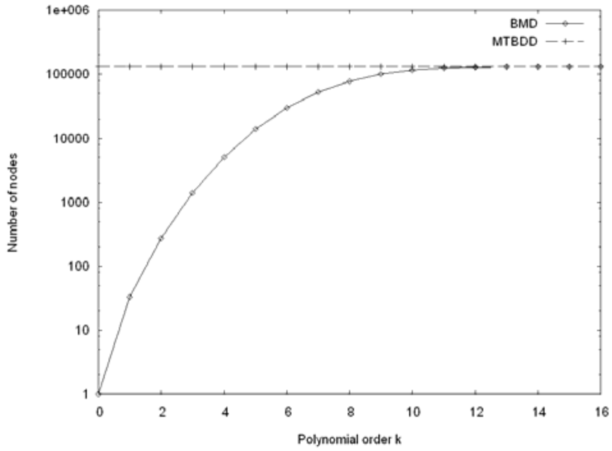


図 2. 16 ビット精度 k 次多項式関数における BMD と MTBDD の節点数.

補題 5. n ビット精度 k 次多項式関数を表現する BMD の非終端節点数は，高々

$$\sum_{i=1}^{k} \binom{n}{i}$$

(証明) 付録参照．

定理 1. n ビット精度 k 次多項式関数を表現する BMD の非終端節点数および終端節点数の和は，高々

$$2\sum_{i=0}^{k} \binom{n}{i} - 1$$

(証明) 付録参照．

補題 6. nビット精度関数$f(X)$において，任意のα, βで関係 $\alpha \neq \beta \rightarrow f(\alpha) \neq f(\beta)$ が成立するとき, $f(X)$を表現するMTBDDの節点数は, $2^{n+1} - 1$ である．

(証明) 付録参照．

系 1. nビット精度k次多項式関数を表現するMTBDDの節点数は，高々$2^{n+1} - 1$ である．

例 4. 図2は, 16 ビット精度のk次多項式関数におけるBMDとMTBDDの節点数の上界を示す．図3は, n ビット精度の3次多項式関数におけるBMDとMTBDDの節点数の上界を示す．　**(例終り)**

精度nを固定したとき，k次多項式関数におけるBMDの節点数の上界は，kの値の増加とともに増加する．一方，MTBDDは, kの値とは無関係に節点数の上界が $2^{n+1} - 1$ に決まるため，kの値が小さいとき，BMDの節点数の上界は，MTBDDより小さい．

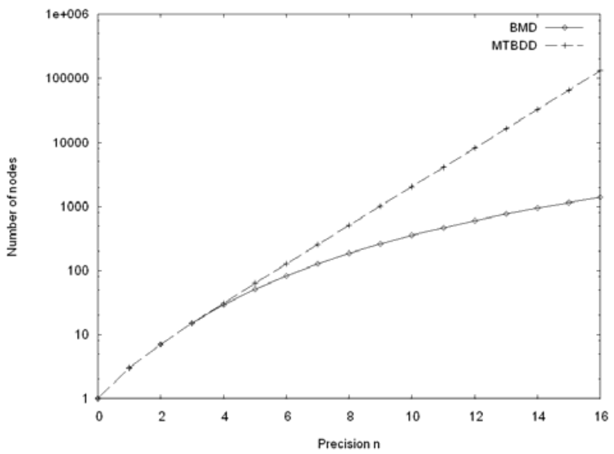


図 3. *n* ビット精度 3 次多項式関数における BMD と MTBDD の節点数.

表 1: *n* ビット精度の 4 次多項式関数の非零算術係数の数

精度	算術係数の数		
n	上界	非零係数	異なる係数
7	99	99	78
8	163	163	111
9	256	256	152
10	386	386	202
11	562	562	262
12	794	794	333
13	1,093	1,093	416
14	1,471	1,471	512
15	1,941	1,941	622
16	2,517	2,517	747

関数値は丸めていない．即ち，丸め誤差なし．

多項式の次数 k を固定したとき，n ビット精度多項式関数における BMD の節点数の上界は，MTBDD よりなだらかに増加するため，精度を上げれば上げるほど, BMD と MTBDD の節点数の上界の差は大きくなる．

以上のことから，n ビット精度 k 次多項式関数では，多項式の次数 k が精度 n より十分小さいとき, BMD の節点数の上界は MTBDD より小さくなる．通常，多項式の次数 k は，精度 n より十分に小さい．そのため，実用的な多項式関数において, BMD は MTBDD よりコンパクトであるといえる．

5　実験結果

5.1　算術係数の数

多項式関数: 補題 2 の実用性を調べるために，n ビット精度 4 次多項式関数 $f(X) = c_4X^4 + c_3X^3 + c_2X^2 + c_1X + c_0$ をランダムに生成し，補題 2 で与えられた上界と生成した多項式関数の非零算術係数の数を比較した．係数 c_i ごとに，$|c_i| \leq 2^{15}$ を満たす一様乱数を生成し，乱数関数を得た．表 1 に比較結果を示す．各精度において, $X \geq 0$ として, 10 個の 4 次多項式をランダムに生成したが，非零算術係数の数は，すべて上界と一致した．表 1 より，補題 2 は，非零算術係数の数を正確に計算することがわかる．また，多項式関数の算術係数は，多くの係数が零になるだけでなく，値の等しい係数が多く存在するため，異なる係数の数は非零係数の数よりも少ないことがわかる．

表 2: 16 ビット精度の初等関数の異なる算術係数の数

初等関数	異なる値の数		比率
	関数値	算術係数	[%]
$2^x - 1$	59,895	148	0.25
$1 / \mathrm{sqrt}(x + 1) - 0.707$	19,196	174	0.90
$\ln(x + 1)$	45,427	165	0.36
$\log_2(x + 1)$	59,895	160	0.27
$\mathrm{sqrt}(x + 1) - 1$	27,147	138	0.50
$2 / (x + 1) - 1$	54,292	180	0.33
$\sin(x)$	55,147	141	0.25

関数値は 16 ビット精度．関数の定義域は, $0 \leq x < 1$.
比率 ＝ 算術係数 / 関数値 * 100 [%].

(非多項式) 初等関数: 表 2 に，16 ビット精度の(非多項式)初等関数における異なる関数値の数と異なる算術係数の数を示す．表 2 より，初等関数は非常にコンパクトな算術スペクトラムに変換できることがわかる．$1 / \mathrm{sqrt}(x + 1) - 0.707$ と $\mathrm{sqrt}(x + 1) - 1$ は，他の関数に比べ値域が狭いので，異なる関数値の数が少ないが，ほとんどの関数で 2^{16} に近い数の異なる関数値がある．一方，異なる算術係数の数は，関数の値域とは無関係に，関数値よりはるかに小さい．異なる関数値と算術係数の数は，それぞれ，MTBDDおよびBMDの終端節点数に対応するため，BMDは，MTBDDよりはるかに少ない終端節点で初等関数を表現できる．

表 3: *n* ビット精度の 4 次多項式関数を表現する BMD と MTBDD の節点数

n	MTBDD		BMD		比率
	上界	節点数	上界	節点数	[%]
7	255	255	197	176	69
8	511	511	325	273	53
9	1,023	1,023	511	407	40
10	2,047	2,047	771	587	29
11	4,095	4,095	1,123	823	20
12	8,191	8,191	1,587	1,126	14
13	16,383	16,383	2,185	1,508	9
14	32,767	32,767	2,941	1,982	6
15	65,535	65,535	3,881	2,562	4
16	131,071	131,071	5,033	3,263	2

関数値は丸めていない．即ち，丸め誤差なし．
比率 ＝BMD の節点数 / MTBDD の節点数 * 100 [%].

5.2　BMD の節点数

多項式関数: 表 3 は，ランダムに生成した n ビット精度 4 次多項式関数を表現する BMD と MTBDD の節点数を比較している. BMD と MTBDD の上界は，定理 1 と系 1 から計算した．表 1 で示したように，多項式関数の多くの算術係数は，

表 4: 16 ビット精度の初等関数を表現する BMD と MTBDD の節点数

初等関数	節点数		比率
	MTBDD	BMD	[%]
$2^x - 1$	122,659	29,634	24
$1 / \mathrm{sqrt}(x + 1) - 0.707$	58,412	28,446	49
$\ln(x + 1)$	100,880	28,442	28
$\log_2(x + 1)$	122,542	29,553	24
$\mathrm{sqrt}(x + 1) - 1$	73,406	26,149	36
$2 / (x + 1) - 1$	114,093	28,348	25
$\sin(x)$	115,450	22,638	20

関数値は 16 ビット精度. 関数の定義域は, $0 \leq x < 1$.
比率 = BMD / MTBDD * 100 [%].

零であり, 非零係数においても値の等しい係数が多く存在するため, 簡単化規則により, BMDの節点数は, 定理 1が与える上界より小さくなる. 一方, MTBDDの節点数は, ランダムに生成したすべての多項式関数で補題 6 が成立し, その上界 $2^{n+1} - 1$ に一致した.

(非多項式) 初等関数: 表4は, 16 ビット精度の初等関数を表現する BMD と MTBDD の節点数を比較している. 表 2 で示したように, BMD の終端節点数は, MTBDD より, はるかに小さいため, 総節点数も MTBDD より小さくなる. 特に BMD は, 様々なアプリケーションでしばしば利用される初等関数 sin(x)を MTBDD の 20%の節点数で表現できる.

6 数値計算回路への応用

BMD の各非終端節点は, 図 4 に示されているように, 加算器と AND ゲートで実現できる. 算術係数(終端節点)をメモリで実現することにより, BMD は, メモリと加算木(adder tree)で実現できる.

初等関数を BMD でコンパクトに表現し, その BMD をメモリと加算木で実現することにより, 私たちは, システマチックな手法で, 初等関数の数値計算回路を生成できる.

7 結論とコメント

本稿は, 多項式関数や三角関数などの初等関数を BMD と MTBDD を用いて表現した. k 次多項式関数を表現する BMD の節点数を解析的に求め, BMD は MTBDD より少ない節点数で k 次多項式関数を表現できることを示した. 特に次数 k に対して精度 n が十分に大きいとき, BMD と MTBDD の節点数の差は大きくなる. 16 ビット精度の様々な初等関数を用いた実験により, 主要な初等関数を表現する BMD の節点数は, MTBDD の 20%であることを示した.

本稿は, 算術変換を用いることで初等関数をコンパクトな決定グラフで表現できることを示した. 算術変換に基づく ACDD, *BMD, K*BMD などの他の決定グラフでも同様

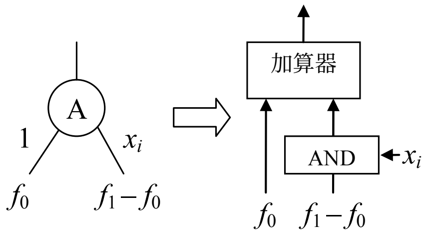


図 4. BMD 節点の実現.

の結果が得られるだろう. 多くの論文[5,6,8,9,10,12,14, 17,18,19]が, 算術変換について研究しているが, 私たちの知る限り, 算術変換による初等関数の表現法について考慮したのは, 私たちが最初である.

謝辞

本研究の一部は, 平成 18 年度科学研究費補助金(若手研究(B))課題番号 18700048 および平成 18 年度広島市立大学教員特定研究費(一般研究)課題番号 6101 による.

参考文献

[1] R. E. Bryant, ``Graph-based algorithms for Boolean function manipulation,'' *IEEE Trans. on Computer*, Vol. C-35, No. 8, pp. 677-691, Aug. 1986.
[2] R. E. Bryant and Y-A. Chen, ``Verification of arithmetic circuits with binary moment diagrams,'' *Design Automation Conference*, pp. 535-541, 1995.
[3] M. J. Ciesielski, P. Kalla, Z. Zheng, and B. Rouzeyre, ``Taylor expansion diagrams: A compact canonical representation with applications to symbolic verification,'' *Design, Automation and Test in Europe (DATE2002)*, pp. 285-289, 2002.
[4] E. M. Clarke, K. L. McMillan, X. Zhao, and M. Fujita, ``Spectral transforms for extremely large Boolean functions,'' *IFIP WG 10.5 Workshop on Applications of the Reed-Muller Expansion in Circuit Design,* pp. 86-90, Sept. 1993.
[5] B. J. Falkowski, `` A note on the polynomial form of Boolean functions and related topics,'' *IEEE Trans. on Computer*, Vol. 48, No. 8, pp. 860-864, Aug. 1999.
[6] K. D. Heidtmann, ``Arithmetic spectrum applied to fault detection for combinational networks,'' *IEEE Trans. on Computer*, Vol. 40, No. 3, pp. 320-324, Mar. 1991.
[7] S. Höreth and R. Drechsler, ``Formal verification of word-level specifications,'' *Design, Automation and Test in Europe (DATE1999)*, pp. 52-58, 1999.
[8] S. L. Hurst, D. M. Muller, and J. C. Muzio, *Spectral Techniques in Digital Logic*, Academic Press, Bristol, 1985.
[9] J. Jain, ``Arithmetic transform of Boolean functions,'' Chapter 6 in [15].
[10] S. K. Kumar and M. A. Breuer, ``Probabilistic aspects of Boolean switching functions via a new transform,'' *Journal of the ACM*, 28 (3), pp. 502-520, 1981.
[11] C. Meinel and T. Theobald, *Algorithms and Data Structures in VLSI Design: OBDD – Foundations and Applications*, Springer, 1998.
[12] V.-D. Malyugin, *Paralleled Calculations by Means of Arithmetic Polynomials*, Physical and Mathematical Publishing Company, Russian Academy of Sci., Moscow, Russia, 1997.
[13] J.-M. Muller, *Elementary Functions: Algorithms and*

Implementation, Birkhauser Boston, Inc., Secaucus, NJ, 1997.
[14] S. G. Papaioannou and W. A. Barrett, ``The real transform of a Boolean function and its application,'' *Computer and Electronic Engineering*, Vol. 2, pp. 215-224, Pergamon Press, 1975.
[15] T. Sasao and M. Fujita (eds.), *Representations of Discrete Functions*, Kluwer Academic Publishers, 1996.
[16] R. Stankovic, T. Sasao, and C. Moraga, ``Spectral transform decision diagrams,'' Chapter 3 in [15].
[17] R. Stankovic and T. Sasao, ``A discussion on the history of research in arithmetic and Reed-Muller expressions,'' *IEEE Trans. on CAD*, Vol. 20, No. 9, pp. 1177-1179, Sept. 2001.
[18] R. Stankovic and J. Astola, *Spectral Interpretation of Decision Diagrams*, Springer Verlag, New York, 2003.
[19] M. A. Thornton, R. Drechsler, and D. M. Muller, *Spectral Techniques in VLSI CAD*, Springer, 2001.

付録

補題 1 の証明: 算術変換の性質より，非零算術係数の数は，

$$X^k = (-2^{n-1}x_{n-1} + 2^{n-2}x_{n-2} + \ldots + 2^1 x_1 + 2^0 x_0)^k$$

を展開整理した式の積項数に等しい．ただしx_i ($i = 0, 1, 2, \ldots, n-1$)は，ブール変数なので$x_i^2 = x_i$である．展開整理後，リテラル数が 1 個の項(即ち，x_iの形をした項)は，${}_nC_1$個存在する．リテラル数が 2 個の項(即ち，$x_i x_j$ ($i < j$)の形をした項)は，${}_nC_2$個存在し，同様にリテラル数がk個の項は，${}_nC_k$個存在する．従って，総積項数は，それらの和で求まる． **(証明終り)**

例 A. 4 ビット精度の$f(X) = X^2$を考える．$X = -8x_3 + 4x_2 + 2x_1 + x_0$であるため，

$$\begin{aligned}X^2 &= (-8x_3 + 4x_2 + 2x_1 + x_0)^2 \\ &= (64x_3^2 + 16x_2^2 + 4x_1^2 + x_0^2) + 2(-32x_2x_3 - 16x_1x_3 \\ &\quad -8x_0x_3 + 8x_1x_2 + 4x_0x_2 + 2x_0x_1) \\ &= 64x_3 + 16x_2 + 4x_1 + x_0 - 64x_2x_3 - 32x_1x_3 \\ &\quad -16x_0x_3 + 16x_1x_2 + 8x_0x_2 + 4x_0x_1\end{aligned}$$

従って，積項数は 10 となる．一方，補題 1 より非零算術係数の数は，${}_4C_1 + {}_4C_2 = 4 + 6 = 10$ となり，上の結果と一致していることがわかる． **(例終り)**

補題 2 の証明: 補題 1 の証明から明らかなように，X^i ($1 \le i < k$)の非零算術係数の集合は，X^kの非零算術係数の集合の真部分集合になっている．また，$c_0 \ne 0$ のとき，定数項に対応する係数${}_nC_0$を追加する必要がある．従って，補題が成立する． **(証明終り)**

補題 3 の証明: 補題 1 の証明から明らかなように，$X \ge 0$ のとき，X^i (i=1, 2, …, k)のすべての非零算術係数は正である．$c_i > 0$ より明らかに，非零算術係数の数はその上界に一致する． **(証明終り)**

補題 4 の証明: X^kを表現するBMTにおいて，変数が根から，x_0, x_1, …, x_{n-1}の順に並んでいるものとする．X^kの算術式は，補題 1 の証明からわかるように，x_iのリテラルがk個以下の項から成っている．X^kの算術式は，これらのすべての項を表現する木構造で表現できる．

木構造の節点数を $\beta(n, k)$とすると，$\beta(n, k)$は，次の関係を満たす．

$$\beta(n,k) = 1 + \beta(n-1,k) + \beta(n-1,k-1) \qquad (A)$$

ここで，1 は根節点の個数を，$\beta(n-1, k)$は左部分木の節点数を，$\beta(n-1, k-1)$は右部分木の節点数を示す．

次に，関数 $\alpha(n, k)$が関係(A)を満たすことを示す．

$$\begin{aligned}&1 + \alpha(n-1,k) + \alpha(n-1,k-1) \\ &= 1 + \sum_{i=1}^{k}\binom{n-1}{i} + \sum_{i=1}^{k-1}\binom{n-1}{i} \\ &= \sum_{i=1}^{k}\binom{n-1}{i} + \sum_{i=0}^{k-1}\binom{n-1}{i} \\ &= \sum_{i=1}^{k}\left[\binom{n-1}{i} + \binom{n-1}{i-1}\right] = \sum_{i=1}^{k}\binom{n}{i} = \alpha(n,k)\end{aligned}$$

従って，定理が成立する． **(証明終り)**

補題 5 の証明: $f(X)$の算術式は，補題 2 の証明からわかるように，x_iのリテラルがk個以下の項から成っている．$f(X)$の算術式は，これらのすべての項を表現する木構造で表現できる．BMTの性質より，BMTでは，定数項の値に関わらず定数項が常に表現されている．従って，定数項c_0の値に関わらず，$f(X)$は高々

$$\sum_{i=1}^{k}\binom{n}{i}$$

の非終端節点で表現できる． **(証明終り)**

定理 1 の証明: 補題 2 より$f(X)$を表現する BMD の終端節点数は，高々

$$\sum_{i=0}^{k}\binom{n}{i}$$

である．また，補題 5 より $f(X)$を表現する BMD の非終端節点数は，

$$\sum_{i=1}^{k}\binom{n}{i}$$

である．以上のことより，定理が成立する． **(証明終り)**

補題 6 の証明: Xが 2^n個の異なる値をとるとき，$f(X)$も 2^n個の異なる値をとる．このとき，$f(X)$を表現するMTBDDは，完全二分木となり，その節点数は $2^{n+1} - 1$ である． **(証明終り)**