semi-enumeration of 8-variable bent functions*

Meng qingshu Zhang huanguo Cui jingsong Yang min
Computer school,Wuhan university, Hubei China 430072

Abstract

By almost classification of R(4,7)/R(2,7) under the action of general affine group, and by a sieve algorithm, we gave almost all cosets of R(2,7) which can be used to construct 7-variable plateaued functions of degree 4. And an efficient algorithm is given to construct bent functions from a plateaued function. Based on above results, we almost enumerate all 8-variable bent function.
keywords: Reed-Muller code, group action, bent functions

1 Introduction

Since the concept of bent function was proposed by Rothaus in 1976[1], there are many papers[2, 3, 4] discussing bent function. A function $f(x) : F_2^n \to F_2$ is called a bent function if the value of Walsh transform of $f(x)$ are always $\pm 2^{n/2}$(the definition of Walsh transform will be defined in section 2). However only in 6 variables case, we know all 3 equivalent classes[1]. For 8 variables, only bent functions of degree 3 is known in [5]. In paper [6] an algorithm theoretically can construct all bent functions, like the enumeration of all 8 variables homogenous bent functions of degree 3 and all homogenous rotation symmetric bent functions of 10 variables, but it is not practical to construct all 8 variables bent functions.

In this paper, we almost classified the Reed-Muller code $R(4,7)/R(2,7)$ under the action of general affine group using invariant theory. For each of the cosets, using the sieve algorithm in paper[6], we got all cosets which can be used to construct plateaued functions[7]. For each plateaued function, an efficient algorithm is given to construct all bent functions from the plateaued function. Based on the above results, it is possible to enumerate all 8-variable bent functions if given certain computation.

2 Preliminary

For each subset $s \subseteq \{1, 2, \cdots, n\}$, there exists a corresponding vector $(s_1, s_2, \cdots, s_n)$ of dimension n by letting $s_i = 1$ if element i is in s else letting $s_i = 0$. And the vector $(s_1, s_2, \cdots, s_n), s_i \in \{0, 1\}$ for $i = 1, 2, \cdots, n$ can be denoted by an integer s whose 2-adic expansion is just the vector

*funded by National Natural Science of China(66973034, 90104005, 60373087)

$(s_1, s_2, \cdots, s_n)$. Obviously, the set, the vector and the integer are isomorphic. In this paper, if confusion is not caused, we will use the three notations for description convenience. Denote by F_2 the Galois field with two elements $\{0, 1\}$ and denote by F_2^n the vector space over F_2. Denote by $p_n = F_2[x_1, x_2, \cdots, x_n]/(x_1^2 - x_1, \cdots, x_n^2 - x_n)$ the algebra of all functions $F_2^n \to F_2$. For each subset $s \subseteq \{1, 2, \cdots, n\}$, denote $\prod_{i \in s} x_i \in p_n$ by x^s. The algebraic normal form of a Boolean function $F_2^n \to F_2$ can be written as $f(x) = \sum_{s=0}^{2^n-1} a_s x^s$, where $a_s \in F_2$. Define

$$deg(f) = \max_{s \in \{0,1,\cdots,2^n-1\}, a_s \neq 0} H(s),$$

where $H(s)$ is the Hamming weight of vector s. Let $R(r, n)$=$\{f(x)|deg(f) \leq r\}$ and $R(r, n)/R(s, n)$=$\{f(x) + R(s, n)|s < deg(f) \leq r\}$.

Denote by $GL(n, 2)$ the set of all nonsingular matrix of order n, i.e. the general linear group. Denote by $AGL(n, 2)$ the general affine group $\{(A, b)|A \in GL(n, 2), b \in F_2^n\}$.

Two functions $f(x), g(x) \in R(r, n)/R(s, n)$ are called equivalent if there exists $(A, b) \in AGL(n, 2)$ such that $f(x) = g(xA + b) \mathrm{mod} R(s, n)$. Two equivalent functions are in one class. An invariant of $R(r, n)/R(s, n)$ is a mapping M from $R(r, n)/R(s, n)$ to a set such that for any two equivalent functions $f(x), g(x) \in R(r, n)/R(s, n)$, $M(f) = M(g)$ holds. Suppose the number of all classes of R(r,n)/R(s,n) is N, an invariant is called a discriminant if it takes exact N distinct values.

3 Basic transform and invariant

This part is all from paper [8]. It is presented here for readers' convenience.

3.1 Walsh Transform and Autocorrelation Function

Definition 1: Define

$$s_{(f)}(w) = \sum_{x \in F_2^n} (-1)^{f(x)} (-1)^{w \cdot x}$$

be the Walsh spectrum of $f(x)$ at vector w, where $f(x) \in p_n, w \in F_2^n$.

The transform is called the Walsh transform of $f(x)$.

Definition 2: Let $c_f(s) = \sum_{x=0}^{2^n-1} (-1)^{f(x)} (-1)^{f(x+s)}$ be the autocorrelation function of $f(x)$, where $f(x) \in p_n, s \in F_2^n$.

The following two propositions are well known.

Proposition 1: Let $f(x), g(x) \in p_n$ be two functions such that $g(x) = f(xA + b) + lx$, then for any $w \in F_2^n$, $s_{(g)}(w) = (-1)^{(l+w) \cdot bA^{(-1)}} s_{(f)}((l + w)A^{-1T}$

Corollary 1: The Walsh spectrum of $f(x)$ at i is equal to the Walsh spectrum of $g(x)$ at j, where $j = l + iA^T$. Therefore the deficiency of the rank of vectors with same spectrum between two equivalent functions is at most 1. The distribution of absolute value of Walsh spectra of $f(x)$ is same to that of $g(x)$.

Proposition 2: Let $f(x), g(x) \in p_n$ be two functions such that $g(x) = f(xA + b) + lx$, then for any given $s \in F_2^n$, $c_g(s) = (-1)^{l \cdot s} c_f(sA)$.

Corollary 2: The autocorrelation function of $f(x)$ at j is equal to the autocorrelation function of $g(x)$ at i, where $j = iA$. Therefore the ranks of vectors with same absolute autocorrelation function value are same between two equivalent functions. The distribution of absolute value of autocorrelation function of $f(x)$ is same to that of $g(x)$.

3.2 Derivation

For any boolean function $f(x) \in R(r, n)$, define its derivation function as $D_a(f) = f(x) + f(x + a)$. Similarly we can define two-order derivation function as $D_{a,b}(f) = f(x)+f(x+a)+f(x+b)+f(x+a+b)$. By definition, it is easy to get following properties:

Property 1[9]: $D_{a,b}(f) = D_a(f) + D_b(f) + D_{a+b}(f)$.

Property 2[9]: $D_a(f \circ B) = D_{aA}(f) \circ B$, where $B \in AGL(n, 2)$. similarly, $D_{a,b}(f \circ B) = D_{aA,bA}(f) \circ B$, where $B \in AGL(n, 2)$.

Proposition 3: If $f(x) \in R(r, n)/R(s, n)$, then $D_a(f \circ B) = (D_{aA}(f)) \circ B \bmod R(s - 1, n)$, where $B = (A, b) \in AGL(n, 2)$. If M is an invariant of $R(r - 1, n)/R(s - 1, n)$, then $M(D_a(f \circ B)) = M((D_{aA}(f)) \circ B)$, so $\{M(D_a(f)|a \in F_2^n\}$ is an invariant of $R(r, n)/R(s, n)$.

Remark: The derivation function is used by Hou [10] in classification of $R(3, 7)/R(2, 7)$ and by Brier[9] in classification of $R(3, 9)/R(2, 9)$. Proposition 3 is an extension of their result.

3.3 Decomposition

Proposition 4: Let $f(x), g(x) \in R(r, n)$ be two functions such that $g(x) = f(xA + b) \bmod R(s, n)$. If $f(x) = (x_1 + 1)f_0(x') + x_1 f_1(x')$, where $x' = (x_2, \cdots, x_n)$, then $g(x) = (x \cdot r_1 + b_1 + 1)f_0(x'') + (x \cdot r_1 + b_1)f_1(x'')$, where $r_1, r_2, \cdots, r_n$ is the row of the matrix A, and $x'' = (x \cdot r_2 + b_2, \cdots, x \cdot r_n + b_n)$. Obviously, $f_0(x'), f_1(x')$ are affinely equivalent to $f_0(x''), f_1(x'')$ respectively. Similar result holds for two-vector based decomposition.

By proposition 4, if $f(x)$ is decomposed into two subfunctions at vector b (like $b = (1, 0, \cdots, 0)$), then $g(x)$ can be decomposed into two subfunctions at vector $a = bA$(like the $a = bA = r_1$)such that the two subfunctions of $f(x)$ are equivalent to those of $g(x)$.

Proposition 5: If M is an invariant of $R(r, n-1)/R(s, n-1)$, then the set $\{\{M(f_{ax=0}), M(f_{ax=1})\}|a \in F_2^n\}$ is an invariant of $R(r, n)/R(s, n)$.

Remarks: The basic idea of the decomposition of a function can be found early in Maiorana's paper[11], which made the classification of R(6,6)/R(1,6) possible early in 90s in 20th century. And recently it is used by Brier[9] to classify R(3,9)/R(2,9).

3.4 The Modification of Truth Table

Definition 3[12]: For a function $f(x)$, define its 1-local connection functions as

$$f_i(x) = \{ \begin{array}{ll} f(x) & x \neq i \\ f(x) + 1 & x = i \end{array} , i = 0, 1, \cdots, 2^n - 1.$$

similarly 2-local connection functions can be defined.

Proposition 6[12]: Let $f(x), g(x) \in R(r,n)$ be such that $g(x) = f(xA+b)+lx$, then $g_j(x) = f_i(xA+b)+lx$, where $jA = (i+b), i = 0,1,\cdots,2^n-1$. Similar result holds for two-local connection functions.

Proposition 7: If M is an invariant of $R(n,n)/R(1,n)$, then $\{M(f_i(x)|i \in F_2^n\}$ is an invariant of $R(r,n)/R(1,n)$.

4 Almost classification of R(4,7)/R(2,7)

In paper[8],some invariants are given. A discriminant is given to classify R(3,7)/R(1,7) based on proposition 5 and corollary 1 and 2. A discriminant is given to classify R(4,6)/R(2,6) based on proposition 3 and corollary 1. Here we will use them to almost completely classify the Reed-Muller code R(4,7)/R(2,7) under the action of $AGL(7,2$.

algorithm 1

By Hou's result[10], $R(4,,7)/R(3,7)$ can be classified into 12 cosets, denote by $g_i(x)+R(3,7), deg(g_i) = 4, i = 1,2,\cdots,12$, which can got by complement the 12 cosets of $R(3,7)/R(2,7)$. So $R(4,7)/R(2,7)$ can be first classified into 12 sets of forms: $g_i(x)+R(3,7)/R(2,7), i = 1,2,\cdots,12$. We can classify the 12 sets one by one. For a given set, say $g_i(x)+R(3,7)/R(2,7)$, do the following steps.

For any a function $f(x) \in g_i(x)+R(3,7)/R(2,7)$,

1. Decompose the function $f(x)$ based on one vector a into two subfunctions $f_{ax=0}(x), f_{ax=1}(x) \in R(4,6)/R(2,6)$. As the discriminant for R(4,6)/R(2,6),denote by $D^6_{4,2}$, is known in [8], an invariant, like $D1_a(f) = \{D^6_{4,2}(f_{ax=0}), D^6_{4,2}(f_{ax=1})\}$, is calculated for these two subfunctions. Now the distribution $\{D1_a|a \in F_2^n, a \neq 0\}$ is an invariant of $g_i(x)+R(3,7)/R(2,7)$.
2. Let $f_a(x) \in R(3,7)/R(1,7)$ be a derivative function of $f(x)$ based on one vector a. As the discriminant for $R(3,7)/R(1,7)$, denote by $D^7_{3,1}$, is known, the distribution $\{D^7_{3,1}(f_a)|a \in F_2^n, a \neq 0\}$ is an invariant of $g_i(x)+R(3,7)/R(2,7)$.

Theoretically the direct product of the above two invariants is an invariant of $g_i(x)+R(3,7)/R(2,7)$, but it is not practical or too expensive in computation as there are 2^{35} functions in $g_i(x)+R(3,7)/R(2,7)$. A practical method is needed.

algorithm 2: practical one

There are 35 monomial of degree 3 in $R(3,7)/R(2,7)$, they can be represented as $x^s, H(s) = 3$, where $H(s)$ is the Hamming weight of vector s. They can be numbered as $0,1,\cdots,34$ according to the value of s in a way that the x^s is numbered as 0 if s is least, and x^s is numbered as 34 if s is the largest. Now we can construct a one to one corresponding between a homogeneous function and a 35-bit unsigned integers as follows: if the ith monomial is in the function, then the ith bit of the integer is 1 else is 0.

With above description, we divide the 35 monomials into four groups, named as $G1, G2, G3, G4$, of size 10,10,10,5 respectively, that is the first 10 least monomials in $G1$, and so on. And denote by $FG1, FG2, FG3, FG4$

the set of homogeneous functions generated by $G1, G2, G3, G4$ respectively.

For a given set $g_i(x) + R(3,7)/R(2,7)$,

1. Use the algorithm 1 to classify the set $g_i(x) + FG1/R(2,7)$, and denote by $RG1$ the set of equivalent classes.
2. Use the algorithm 1 to classify the set $\{h(x)+l(x)|h(x) \in RG1, l(x) \in FG2\}$. Denote by $RG2$ the set of equivalent classes.
3. Use the algorithm 1 to classify the set $\{h(x)+l(x)|h(x) \in RG2, l(x) \in FG3\}$. Denote by $RG3$ the set of equivalent classes.
4. Use the algorithm 1 to classify the set $\{h(x)+l(x)|h(x) \in RG3, l(x) \in FG3\}$. Denote by $RG4$ the set of equivalent classes.

The functions in $RG4$ are not affinely equivalent. That is , the coset $g_i(x) + R(3,7)/R(2,7)$ is classified into at least $|RG4|$ classes. Here we use "at least" for the following two reasons:

1. We don't know if the invariant used in algorithm 1 is a discriminant for $g_i(x) + R(3,7)/R(2,7)$.
2. Even the invariant used in algorithm 1 is indeed a discriminant, by the proof of fact 1 in paper [6], theoretically some classes can be lost by algorithm 2.

Using algorithm 2, the 12 sets are classified into 12, 63, 285, 474, 686, 185, 108, 6371, 1013, 33598, 1298, 23987 classes respectively. The sum of all these classes is 68080, only $68447 - 68080 = 367$ classes are lost. So we almost classified the 12 sets $g_i(x) + R(3,7)/R(2,7), i = 1, 2, \cdots, 12$.

5 Functions in R(4,7)/R(2,7) which can be expanded into bent functions

In paper[6], a novel algorithm is given to search bent function more efficiently. We give a short description related to our work.

Lemma 1 [13]. Let

$$f(x_1, x_2, \cdots, x_n) = \sum_{i=0}^{2^k-1} \delta_{a_i}(x') f_i(x''),$$

where $x' = (x_1, x_2, \cdots, x_k)$, $x'' = (x_{k+1}, x_{k+2}, \cdots, x_n)$, $f_i(x'') : F_2^{n-k} \to F_2, i = 0, 1, \cdots, 2^k - 1$, the integer representation of $a_i \in F_2^k$ is i, $\delta_{a_i}(x') = \begin{cases} 1, & a_i = x' \\ 0, & a_i \neq x' \end{cases}$, then

$$[s_{(f_0)}(w''), s_{(f_1)}(w''), \cdots, s_{(f_{2^k-1})}(w'')]^T$$

$$= [s_{(f)}(a_0, w''), s_{(f)}(a_1, w''), \cdots, s_{(f)}(a_{2^k-1}, w'')] H_k/2^k, \qquad (1)$$

where $w = (w', w''), w'' \in F_2^{n-k}$, H_k is a Hadamard matrix.

lemma 2[6]. If

$$f(x_1, x_2, \cdots, x_n) = \sum_{i=0}^{2^k-1} \delta_{a_i}(x') f_i(x''),$$

is a bent function, then every spectrum $s_{(f_i)}(w'')$ can take the following $2^k + 1$ values:

$$\{(2^k - j)2^{n/2} - j2^{n/2}\}/2^k = (2^k - 2j)2^{n/2-k}, j = 0, 1, \cdots, 2^k.$$

All these values are called the k-th Granted-value.

For example, let $k = 1$, a bent function $f(x)$ is divided into two sub-functions $f_0(x''), f_1(x'')$. The Walsh spectra the two sub-functions can take are $0, \pm 2^{n/2}$. The two sub-functions is called complementary plateaued functions. Let $k = 2$, we get five values: $0, \pm 2^{n/2-1}, \pm 2^{n/2}$. Similarly we can let $k = 3, 4, \cdots, n/2 - 1$.

Now we consider a concrete case: the number of variables $n = 8$, with $k = 1$, the set of the first Granted-value is $\{0, \pm 16\}$. With $k = 2$, the set of the 2nd Granted-value is $\{0, \pm 8, \pm 16\}$. With $k = 3$, the set of the 3rd Granted-value is $\{0, \pm 4, \pm 8, \pm 12, \pm 16\}$.

With above lemmas 1,2 , we can check whether the functions in $\{f(x) + g(x) | f(x) \in RG4, g(x) \in R(2,7)/R(1,7)\}$ could be expanded into bent functions. This is easy. There are 68080×2^{21} functions, and check if their Walsh spectra take the first Granted-value. If there exists function $g(x) \in R(2,7)/R(1,7)$ such that the Walsh spectra of $f(x) + g(x)$ take the first Granted-Value, then the function $f(x) \in RG4$ is reserved. The exact number of reserved function is as follows: $(12, 6), (63, 24), (285, 128), (474, 156), (686, 327), (185, 55), (108, 44), (6371, 3306), (1013, 501), (33598, 16851)$, (1298,658), (23987,11993). Here the first number in bracket is the number of all classes and the second number is the number of reserved classes .

An interesting observation is that there are 34049, about half of the total number 68080, functions in $RG4$ that can be expanded into bent functions, which reminds me of 6 variables case. There are 6 classes in $R(3,6)/R(2,6)$ under the action of $AGL(6,2)$, and half of them could be used to construct bent functions.

Denote by HB the set of all 34049 reserved functions. For any a function in HB, it is easy to construct a plateaued functions. By corollary 2 in paper[14],from a balanced plateaued function we can construct plateaued function with more variables.

If some readers want the 34049 reserved functions, contact me by email.

6 Algorithm to enumerate 8-variable bent functions

As in last part, it is easy to get a plateaued function from a function in HB. In this part an algorithm is given by which it is efficient to construct all bent functions from a plateaued function.

lemma 3[1] : Let $f(x)$ be a bent function. $\widetilde{f(x)}$ be such that $s_{(f)}(w) = 2^{n/2}(-1)^{\widetilde{f(x)}}$, then $\widetilde{f(x)}$ is a bent function, and called the dual function of f(x).

lemma 4

$$s_{(f_0)}(w'') = (s_{(f)}(a_0, w'') + s_{(f)}(a_1, w'') + \cdots + s_{(f)}(a_{2^k-1}, w''))/2^k,$$

$$= ((-1)^{\widetilde{f(a_0,w'')}} + (-1)^{\widetilde{f(a_1,w'')}} + \cdots + (-1)^{\widetilde{f(a_{2^k-1},w'')}})/2^{n/2-k} \quad (2)$$

especially let k=1, then

$$s_{(f_0)}(w'') = (s_f(a_0, w'')+s_f(a_1, w''))/2 = ((-1)^{\widetilde{f(a_0,w'')}}+(-1)^{\widetilde{f(a_1,w'')}})/2^{n/2-1}.$$

Remark: The number of w'' where $s_{(f_0)}(w'') = 0$ depends on the Hamming weight of the derivative function of $\widetilde{f(x)}$ at the vector $(1, 0, \cdots, 0)$. As $\widetilde{f(x)}$ is a bent function by lemma 3, the first order derivative function at vector (1,0,,0)is a balanced function. So there are 2^{n-2} w'' such that $s_{(f_0)}(w'') = 0$. Similarly, we can discuss the case k=2,and k=3. But when k get bigger, the problem gets more complex.

algorithm 3

If $f(x) = (x_1 + 1)f_0(x') + x_1 f_1(x'), x = (x_1, x_2, \cdots, x_n,$ be a bent function in 8 variables, then by paper [7], the two subfunctions are called complementary plateaued functions. The distribution of their Walsh spectra would be of following forms respectively:

$$\overbrace{a, \cdots, a}^{n_1}, \overbrace{b, \cdots, b}^{n_2}, \overbrace{a \cdots, a}^{n_3}, \cdots \quad (3)$$

$$\overbrace{b, \cdots, b}^{n_1}, \overbrace{a', \cdots, a'}^{n_2}, \overbrace{b \cdots, b}^{n_3}, \cdots \quad (4)$$

,where a is ± 16 ,a' is $\pm 16$and $b = 0$. The spectra of $f(x)$ is of form like

$$\overbrace{a, \cdots, a}^{n_1}, \overbrace{a', \cdots, a'}^{n_2}, \overbrace{a \cdots, a}^{n_3}, \cdots, \overbrace{a, \cdots, a}^{n_1}, \overbrace{-a', \cdots, -a'}^{n_2}, \overbrace{a, \cdots, a}^{n_3}, \cdots \quad (5)$$

If 1 is substituted for 16 and -1 for -16, then the sequence resulted from the substitution should be a truth table of another bent function by lemma 3, Denote by sequence S.

Now given a plateaued function(that is, a is known), to expand it into a bent function means to determine the value of a'. The number of a' is 64 by Lemma 4 and usually the distribution of a' is not very uneven by lemma 4.

1. The sequence S is of 256 length. It can be divided into 8 equal blocks, each of which is a truth table of a 5-variable subfunction. Suppose there are $m1, m2, m3, m4$ a's in first,second,third and fourth block respectively, then $m_1 + m_2 + m_3 + m_4 = 64$. The 5-8th blocks depend on the 1-4th blocks respectively by formula 5. For each of the 8 blocks, substitute -1 or 1 for a', and check if the Walsh spectra of the block take the 3rd Granted-value. If they do take the 3rd

Granted value, then the substitution is right, else the substitution is not proper, discard this substitution. The number of substitutions in this step is $2^{m_1} + 2^{m_2} + 2^{m_3} + 2^{m_4}$. After this step, suppose there are $N1, N2, N3, N4$ substitutions are reserved.

2. Divide the sequence 3 into 4 blocks, each of which is then a truth table of a 6 variables subfunction. There are $N_1 \times N2$ substitutions in first block, and $N3 \times N4$ in second block. The third and the fourth block depend on the first and second block respectively. For each of the four blocks, check if the Walsh spectra take the second Granted-value. Suppose there are M_1, M_2 substitutions are reserved in first and second block respectively.
3. Now there are $M_1 \times M_2$ substitutions, check if the sequence S are bent function.

The dual function of sequence S is the bent function we searched. By the following lemma 5, sequence S can also be taken as the bent function we searched.

lemma 5: Let $f(x), g(x) \in p_n$ be two bent functions such that $g(x) = f(xA + b)$, then $\widetilde{g(x)} = x \cdot bA^{-1} + f(\widetilde{xA^{-1T}})$.

Proof: it is from proposition 1 and lemma 3.

7 Conclusion

By almost classification of R(4,7)/R(2,7), and by the algorithm in paper[6] ,we give all functions in $R(4,7)/R(2,7)$ under action of $AGL(7,2)$, which can be expanded into 8-variable bent functions. Some functions in R(4,7)/R(2,7), which can be expanded into bent function, may be lost. And only an efficient algorithm is given to construct all bent functions from a plateaued function. That is, we don't give all bent functions directly. Therefor it is called semi-enumeration, and it is not a perfect result. We only announce the news of this result in the first Chinese Conference on Trusted Computing and Information Security[15]. However our result is useful in constructing 8 variables bent functions. Just recently, Dobbertin [16] published a toolkit in constructing 8-bit bent functions. We still have not study their result. It would be great if the combination of the two results leads to the complete enumeration and complete classification of of all 8-variable bent functions.

References

[1] Rothaus, O. S., On "Bent" Functions, J. Combin. Theory Ser. A.,1976, 20, 300305.

[2] J. F. Dillon. Elementary Hadmard Difference Sets. Ph. D, Dissertation, Unv. Maryland, 1974

[3] Claude Carlet, generalized partial spreads, ieee Trans.on I.T. Vol 41,No.5, 1482-1487, september, 1995.

[4] Claud Carlet, Philippe Guillot, a characterization of binary bent functions. journal of combinatorial theory, series A 76, 328-335 (1996)

[5] xiang-dong hou, cubic bent functions, discrete mathematics,1998,189,149-161.

[6] qing-shu meng, huanguo zhang, min yang, jingsong cui, A novel algorithm enumerating of bent functions, http://eprint.icar.org, 2004/274.

[7] yuliang zheng, xianmo zhang, relationships between bent functions and complementary plateaued functions, proc. 2nd inter. Conf. Information security and cryptology, lncs 1787,1999,60-75

[8] qing-shu meng, Min Yang, Huanguo zhang, Yuzhen Liu, Analysis of affinely equivalent boolean functions. http://eprint.icar.org, 2005/025.

[9] ERIC BRIERR and PHILLIPPE LANGEVIN: 'Classification of Boolean cubic forms in nine variables', 2003 Ieee Information Theory Workshop, pp.179-182

[10] xiang-dong hou, GL(m,2) acting on R(r,m)/R(r-1,m), discrete mathematics, 149(1996) 99-122.

[11] MAIORANA.J.A: 'A classification of the cosets of the reed-muller code R(1,6)', Math. Comp.1991,57,pp.403-414

[12] FULLER.J. and MILLAN.W.: 'Linear redundancy in S-box', In: Fast Software Encryption, LNCS 2887, Springer-Verlag, 2003, pp.74-86

[13] Qing-shu Meng, Huan-guo Zhang,Zhang-yi Wang,etc. Designing bent functions using evolving computing. Acta electronica sinica, 2004, No.11 1901-1903.

[14] Xiangyong Zeng, Lei Hu. A composition construction of bent-like boolean functions from quadratic polynomials. http://eprint.iacr.org,2003/204.

[15] huanguo zhang, the first Chinese conference on trusted computing and information security'04. Wuhan university journal of natural sciences, Vol. 10,No.1,2005.

[16] H. Dobbertin, G.Leander, cryptographer's toolkit for construction of 8-bit bent functions. http://eprint.iacr.org, 2005/089.

appendix Appendix A:

1. f1=0
2. f2=x4x5x6x7
3. f3=x4x5x6x7+x1x3x6x7
4. f4=x4x5x6x7+x1x2x3x7
5. f5=x4x5x6x7+x1x3x6x7+x1x2x5x7
6. f6=x4x5x6x7+x2x3x6x7+x1x3x5x7+x1x2x4x7+x1x2x3x7
7. f7=x3x4x5x6+x1x2x5x6+x1x2x3x4
8. f8=x4x5x6x7+x1x2x3x7+x2x3x5x6
9. f9=x4x5x6x7+x1x3x6x7+x1x2x5x7+x2x3x5x6
10. f10=x4x5x6x7+x1x2x3x7+x2x3x5x6+x1x3x4x6
11. f11=x4x5x6x7+x2x3x6x7+x1x3x5x7+x1x2x4x7+x1x2x3x7+x2x3x4x5
12. f12=x4x5x6x7+x2x3x6x7+x1x3x5x7+x1x2x4x7+xx1x2x3x7x2x3x4x5+x1x3x5x6