Correspondence

On the Number of Spurious Memories in the Hopfield Model

JEHOSHUA BRUCK, MEMBER, IEEE, AND
VWANI P. ROYCHOWDHURY

Abstract—**It is shown that the outer-product method for programming the Hopfield model is shown, which can result in many spurious stable states—exponential in the number of vectors that we want to store—even in the case when the vectors are orthogonal.**

I. Introduction

We consider the neural network model that was suggested by Hopfield in 1982 [10]. It is a discrete time system that can be represented by a weighted graph. There is a weight attached to each edge of the graph and a threshold value attached to each node (neuron) of the graph. The order of the network is the number of nodes in the corresponding graph. Let N be a neural network of order n; then N is uniquely defined by (W, T) where

- W is an $n \times n$ matrix, with element w_{ij} equal to the weight attached to edge (i, j);
- T is a vector of dimension n, where element t_i denotes the threshold attached to node i.

Every node (neuron) can be in one of two possible states, either 1 or -1. The state of node i at time t is denoted by $v_i(t)$. The state of the neural network at time t is the vector $V(t) = (v_1(t), v_2(t), \cdots, v_n(t))$.

The state of a node at time $(t+1)$ is computed by

$$v_i(t+1) = \operatorname{sgn}(H_i(t)) = \begin{cases} 1, & \text{if } H_i(t) \geq 0 \\ -1, & \text{otherwise} \end{cases} \quad (1)$$

where

$$H_i(t) = \sum_{j=1}^{n} w_{j,i} v_j(t) - t_i.$$

The next state of the network, i.e., $V(t+1)$, is computed from the current state by performing the evaluation (1) at a single node of the network. This mode of operation is known as serial or asynchronous mode. The node at which the computation is performed can be chosen at random or according to some deterministic rule.

A state $V(t)$ is called stable iff $V(t) = \operatorname{sgn}(WV(t) - T)$, i.e., the state of the network is not changing as a result of computation. The set of stable states of a network N is denoted by M_N.

Manuscript received March 28, 1989; revised August 1989. This work was supported in part by AFOSR under contract 88-0024A, the Department of the Navy, Office of Naval Research under Contract N00014-86-K-0726, the SDIO/IST, managed by the Army Research Office under Contract DAAL03-87-K-0033, Rockwell International under Contract 6G3052, and the U.S. Army Research Contract DAAL03-86-K-0045. The material in this paper was presented in part at the IEEE Information Theory Symposium, San Diego, CA, January 14–19, 1990.

J. Bruck is with the IBM Research Division, Almaden Research Center, 650 Harry Road, San Jose, CA 95120-6099.

V. P. Roychowdhury is with the Information Systems Laboratory, Stanford University, Stanford, CA 94305.

IEEE Log Number 8933772.

One of the most important properties of the model is the fact that, when it operates in a serial mode, it will always get to a stable state (provided W is a symmetric matrix with nonnegative diagonal); see [5], [7], [10] for more details on convergence properties. This property suggests the use of the model as an associative memory device. An associative memory is a device that memorizes a set M of distinct n-bit vectors. It gets as an input an n-bit vector and its output is a vector which belongs to M and is the closest (e.g., in Hamming distance) to the input vector. The idea is that a network N can implement an associative memory with $M \subseteq M_N$ and the association done by convergence to the closest stable state. There are many interesting questions related to this idea [1], [6], [11], [15].

One of the interesting issues concerning the use of the network as an associative memory is how one should program the network. Programming of a network can be defined as follows.

Consider the set $M = \{V_1, V_2, \cdots, V_s\}$ that consists of s vectors over $\{1, -1\}^n$. Construct a network $N = (W, T)$ such that $M \subseteq M_N$, i.e., the set M is a subset of the set of stable states of N. Hopfield [10] suggested computing W by the outer-product method (which is a Hebb-type rule [9]). Namely,

$$W = \sum_{i=1}^{s} \left(V_i V_i^T - I_n \right)$$

where I_n is the $n \times n$ identity matrix. Using this method, T is chosen to be the all-zero vector. Note that if the V_i's are orthogonal then

$$WV_1 = (n - s) V_1.$$

So if $n > s$ every one of the V_i's is stored. Hence, a natural question is: are there any other (spurious) stable states? Namely, what can be said about the number of stable states that are not in M?

The main contribution of this correspondence is proving that in certain cases the number of spurious memories (vectors which are in M_N but not in M) can depend exponentially on s. Our results hold for the three following cases that cover all the possibilities for s:

1) s is small: $1 \leq s \leq \log n$.
2) s is big: $n - \log n \leq s < n$.
3) The intermediate cases: $s = 2^k$, where $0 \leq k < \log n$.

The results are the first constructive evidence for the results in [1], [14], [15], [18] where such a phenomenon was suggested based on probabilistic arguments.

The paper is organized as follows. In Section II we prove the main combinatorial theorems which provide examples of networks with exponentially many spurious memories. The main tool for proving the results is based on treating vectors over $\{1, -1\}^n$ as Boolean functions of $\log n$ variables. A way to do it is by using the polynomial representation of Boolean functions that is presented in the Appendix. We advise the reader to read the Appendix before going through the proofs in the next section.

0018-9448/90/0300-0393$01.00 ©1990 IEEE

II. Combinatorial Results

We now prove that the number of spurious memories can be exponentially big in the number of memories we are trying to store using the outer-product method. Intuitively, we should suspect something like this to happen when the number of vectors that we want to store is close to n; surprisingly, this also happens in cases when the number of vectors is small. The first result (Theorem 1) provides an example in which the number of vectors, say s, to be stored is less than or equal to $\log n$; and the number of stable states in the resulting network is exponential in s. This question is also addressed in [2], where the result is proved for $s = 3$, 5, and 7. Here we present a proof for an arbitrary odd s. The idea in the proof is to represent vectors over $\{1,-1\}^n$ by polynomials. For example, consider vectors over $\{1,-1\}^8$; then any vector can be represented by a polynomial of three variables. The vectors that correspond to the three variables are

$$X_1 = (1,-1,1,-1,1,-1,1,-1),$$
$$X_2 = (1,1,-1,-1,1,1,-1,-1),$$

and

$$X_3 = (1,1,1,1,-1,-1,-1,-1).$$

Theorem 1: Let $s \geq 1$ be odd. Let $n = 2^s$. Consider the s vectors $X_1, X_2, \cdots, X_s$ over $\{1,-1\}^n$ that correspond to the Boolean functions defined by the x_i's. Let W be the matrix that is computed by the outer-product method, i.e.,

$$W = \sum_{i=1}^{s} \left(X_i X_i^T - I_n \right).$$

Consider the vectors of the form

$$U_\beta = \operatorname{sgn}\left(\sum_{i=1}^{s} \beta_i X_i \right)$$

where $\beta = (\beta_1, \beta_2, \cdots, \beta_s)$. Then: 1) For all $\beta \in \{1,0,-1\}^s$, such that the support of β (number of nonzero entries in β) is odd, the vector U_β is stable in the network $N = (W, T)$ (T is the all-zero vector). 2) All the $(3^s+1)/2$ U_β's that correspond to β with odd support are distinct.

Proof: The idea in the proof is to compute WU_β and to show that $\operatorname{sgn}(WU_\beta) = U_\beta$. First we prove that, for β being the all-1 vector and for all $1 \leq i \leq s$, we have

$$X_i^T U_\beta = 2\binom{s-1}{\frac{s-1}{2}}.$$

Using the notation in Appendix I, $X_1^T U_\beta = 2^s a_{1,0,\cdots,0}$. Namely, the inner product of U_β with X_1 is just the corresponding spectral coefficient times 2^s. Without loss of generality we prove the result for $i = 1$. Notice that for β being the all-1 vector the vector U_β corresponds to the majority function of s variables. To compute the spectral coefficient that corresponds to X_1 consider the vector

$$V = X_2 + X_3 + \cdots + X_s.$$

Clearly, V is zero in

$$2\binom{s-1}{\frac{s-1}{2}}$$

entries. For example, for $s = 3$, $X_2 + X_3$ is zero in four entries—see the example preceding the theorem. There is a symmetry in the values of the other entries, half of them being negative and half positive. Hence, in $X_1^T \operatorname{sgn}(X_1 + V)$, the positive and negative entries cancel each other and we get as a result

$$2\binom{s-1}{\frac{s-1}{2}}.$$

By similar arguments we get the result for general β with odd support w

$$X_i^T U_\beta = \beta_i 2^{s-w+1}\binom{w-1}{\frac{w-1}{2}}.$$

Hence for β with odd support w we have

$$WU_\beta = 2^{s-w+1}\binom{w-1}{\frac{w-1}{2}} \sum_{i=1}^{s} \beta_i X_i - sU_\beta.$$

From the foregoing equation it follows that the sign of WU_β is dominated by the sign of

$$\sum_{i=1}^{s} \beta_i X_i.$$

Thus, for all odd $s \geq 1$ and $\beta \in \{1,0,-1\}^s$, such that the support of β is odd, we have $\operatorname{sgn}(WU_\beta) = U_\beta$. The U_β's are distinct because they all have a distinct polynomial representation. □

A natural question is whether such phenomena hold for other values of s. The next result (Theorem 2) shows that indeed, for any $s = 2^k$ where $0 \leq k < \log n$, there are sets of s orthogonal vectors that will result in an exponential number of spurious memories.

A note regarding the technique: in the foregoing theorem the spurious memories are a nonlinear function of the vectors that we want to store. In the results to follow we use a different technique to prove that a state is stable: we prove that it is in the linear span of the vectors that were stored. By the following lemma, every state that is in the linear span of the stored vectors is also stable.

Lemma 1: Let $1 \leq s < n$. Let $V_1, V_2, \cdots, V_s$ be a set of orthogonal vectors where $V_i \in \{1,-1\}^n$ for all $1 \leq i \leq s$. Let

$$W = \sum_{i=1}^{s} \left(V_i V_i^T - I_n \right).$$

Then a vector $V \in \{1,-1\}^n$ that is in the linear span of the V_i's corresponds to a stable state in $N = (W, T)$ with $T = 0$.

Proof: V is in the linear span, hence there exist γ_i's such that

$$V = \sum_{i=1}^{s} \gamma_i V_i.$$

Thus, $WV = (n-s)V$. So if $s < n$ we get that $\operatorname{sgn}(WV) = V$. □

Theorem 2: Let $n = 2^s$. For every $0 \leq k < s$ there exists a set of 2^k orthogonal vectors $\{V_1, V_2, \cdots, V_{2^k}\}$, where $V_i \in \{1,-1\}^n$, such that when W is computed by the outer-product method, i.e.,

$$W = \sum_{i=1}^{2^k} \left(V_i V_i^T - I_n \right),$$

the network $N = (W, 0)$ has 2^{2^k} stable states.

Proof: The idea in the proof is to choose the vectors V_i's to be the basis functions of the Boolean functions with k variables (there are 2^k basis functions). For example, for $k = 2$ we consider

the vectors that correspond to 1, X_1, X_2 and X_1X_2. Clearly, all the Boolean functions of k variables (there are 2^{2^k} of those) are in the linear span of those vectors. Hence by Lemma 1, all of those are stable in N. □

The next interesting case is the case in which s, the number of vectors to be stored, is very close to n: i.e., $n-\log n \le s < n$. In turns out that for $n-1$, $n-2$ and $n-3$ orthogonal vectors we can count exactly the number of vectors in the linear span. In the next theorem we give counting results for these cases. Note that those results hold for n for which there exist Hadamard matrices (not just for n being a power of 2 as in previous results).

Theorem 3: The number of vectors over $\{1,-1\}^n$ in the linear span of

1) $(n-1)$ orthogonal vectors is

$$\binom{n}{\frac{n}{2}},$$

2) $(n-2)$ orthogonal vectors is

$$\binom{\frac{n}{2}}{\frac{n}{4}}^2,$$

and

3) $(n-3)$ orthogonal vectors is

$$\sum_{j=0}^{\frac{n}{4}} \binom{\frac{n}{4}}{j}^4.$$

Proof: Throughout the proof we consider only vectors over $\{1,-1\}^n$. Counting the number of vectors in the linear span is the same as counting the number of vectors that are orthogonal to the null-space. Namely, we have to count the number of vectors over $\{1,-1\}^n$ that are orthogonal to a single vector (for 1)), two vectors (for 2)) and three vectors (for 3)).

Without loss of generality, assume that the single vector is the all-1 vector. Clearly, the vectors that are orthogonal to this vector are those that consist of $(n/2)$ 1's and $(n/2)$-1's. Hence we get 1): the number of vectors in the linear span of any $(n-1)$ orthogonal vectors is $\binom{n}{n/2}$.

For 2), we consider, without loss of generality, the all-1 vector to be denoted by U_1 and a vector in which half of the entries are 1 and the other half are -1 to be denoted by U_2. We introduce some notation: let $N^{++}(U,V)$ be the number of entries in which both U and V are 1. Similarly, let $N^{+-}(U,V)$ be the number of entries in which there is a 1 in U and a -1 in V, and let N^{-+} and N^{--} denote the other two cases. Assume that V is orthogonal to both U_1 and U_2; then we have

$$N^{++}(U_2,V)+N^{+-}(U_2,V)=\frac{n}{2},$$

$$N^{++}(U_2,V)+N^{-+}(U_2,V)=\frac{n}{2},$$

$$N^{++}(U_2,V)+N^{--}(U_2,V)=\frac{n}{2}$$

and

$$N^{+-}(U_2,V)+N^{-+}(U_2,V)=\frac{n}{2}.$$

From these equations we get the following necessary and sufficient condition for V to be orthogonal to both U_1 and U_2:

$$N^{++}(U_2,V)=N^{--}(U_2,V)=N^{+-}(U_2,V)=N^{-+}(U_2,V)=\frac{n}{4}.$$

Hence, there are $\binom{n/2}{n/4}^2$ vectors that are orthogonal to both U_1 and U_2.

For 3) we consider three orthogonal vectors U_1, U_2, and U_3. We choose, without loss of generality, U_1 and U_2 as in 2). From 2) we know that there is a unique canonical form for U_3 in which the first quarter is 1, the second is -1, the third quarter is 1 and the fourth quarter is -1. Consider a vector V that is orthogonal to U_1, U_2, and U_3. Let j be the number of 1's in the first quarter of V. By similar arguments as in 2) we get that the number of -1's in the second quarter is j, the number of -1's in the third quarter is j and the number of 1's in the fourth quarter is j. Again this is a necessary and sufficient condition for a vector to be orthogonal to U_1, U_2 and U_3. Hence the number of vectors in a linear span of $(n-3)$ orthogonal vectors is

$$\sum_{j=0}^{\frac{n}{4}} \binom{\frac{n}{4}}{j}^4.$$

□

So by Lemma 1 we get that number of stable states is indeed exponential for the cases of $n-1$, $n-2$ and $n-3$ orthogonal vectors. An important remark is that those are the only stable states, namely, there are no other stable states besides those in the linear span.

The foregoing approach for counting vectors in the linear span does not work for $n-4$: we do not have a canonical form any more (this phenomenon is related to the question of existence of Hadamard matrices of order n that is divisible by four [13]). Hence, we assume that n is a power of 2 (we consider the Sylvester-type Hadamard matrix) and we would like to count the number of high-frequency Boolean functions—functions which have a zero constant term and zero linear terms in the polynomial representation. Counting the number of high-frequency Boolean functions is the same as counting the number of vectors in the linear span of $n-\log n$ orthogonal vectors (those that correspond to higher order terms in the polynomial representation). In the following theorem we provide a lower bound on this number.

Theorem 4: Let $n=2^s$. Consider the set of vectors $S=\{1, X_1,\cdots, X_s\}$ over $\{1,-1\}^n$ that corresponds to the Boolean functions defined by $1, x_1,\cdots, x_s$. The number of vectors in $\{1,-1\}^n$ that are orthogonal to the vectors in S is at least $2^{n/4}$.

Proof: The number of Boolean functions (vectors over $\{1,-1\}^n$) that are functions only of the first $s-2$ variables is $2^{2^{s-2}}=2^{n/4}$. Let $f(x_1,\cdots,x_s)$ be one of this functions, namely, f is a function only of the first $(s-2)$ variables. Let

$$g(x_1,\cdots,x_s)=x_{s-1}\oplus x_s\oplus f(x_1,\cdots,x_s).$$

where $\oplus$ is exclusive OR. Consider the polynomial representation of g

$$g(X)=x_{s-1}x_s f(X).$$

Since f is dependent only on the first $s-2$ variables, in the polynomial representation of $g(X)$, we have $a_\alpha=0$ for all α of weight (number of 1's) less or equal 1. In other words, the polynomial representation of g has no constant term nor has it

linear terms. In the language of vectors: we exhibited a set of $2^{n/4}$ vectors that are orthogonal to S. □

Again, by Lemma 1 all the vectors in the linear span are stable.

III. Conclusion

We proved that the number of spurious memories resulting from using the outer-product method is in many cases exponentially big. A few remarks follow.

1) Bad Networks: In the discussion in the previous section we were interested in lower bounds on the number of spurious memories. It is also possible to give a description of the networks associated with the constructions and compute the exact number of spurious memories. For example, consider the construction in Theorem 2, for $n/2$ vectors we get

$$W = \begin{bmatrix} 0 & I_{n/2} \\ I_{n/2} & 0 \end{bmatrix},$$

where $I_{n/2}$ is an $n/2 \times n/2$ identity matrix. The network associated with W consists of $n/2$ pairs of nodes each connected with an edge of weight 1. Since every pair of nodes is isolated and has two stable states we get a total of $2^{n/2}$ stable states. For $s = 2^k$ we get a network that consists of 2^k subnetworks each of which is fully connected with all the weights being one. Since there are two stable states in each subnetwork, we get a total of 2^{2^k} stable states.

2) Being more General: Although our results are for some specific sets of vectors, it is not hard to see that they provide evidence for a more general phenomenon. In particular, Theorem 3 holds for any set of $n-1$, $n-2$ and $n-3$ orthogonal vectors. Also any set of orthogonal vectors that contains in its linear span a subset of one of the bad sets (those that result in many spurious memories) of vectors that we exhibited will also be bad.

3) Boolean functions: The main tool in proving the results was the polynomial representation of Boolean functions. One of results in the correspondence is a lower bound of $2^{n/4}$ on the number of high frequency Boolean functions. In fact, we can also prove a better result—a lower bound of $2^{n-\log^2 n}$, but that is beyond the scope of this paper. Computing the exact number of high frequency Boolean functions is left as an open problem.

Appendix
Polynomial Representation of Boolean Functions

The representation of Boolean functions as polynomials over the field of rational numbers is presented. This representation was first suggested by Muller [16] and then was used by Ninomiya [17] and Golomb [8] to get results on counting the number of equivalent Boolean functions. See [3], [4], [12] for more details.

A Boolean function f of n variables is a mapping,

$$f: \{1,-1\}^n \mapsto \{1,-1\}.$$

Note that we use the multiplicative representation of $\{0,1\}$ via the transformation $a \mapsto (-1)^a$.

Definition: Given a Boolean function f of order n, p is a polynomial (with coefficients over the field of rational numbers) equivalent to f iff for all $X \in \{1,-1\}^n$:

$$f(X) = p(X).$$

As an example, let $f = x_1 \oplus x_2$; that is, f is the XOR function of two variables. It is easy to check that in the $\{1,-1\}$ representation $p(x_1, x_2) = x_1 x_2$. Notice that for every Boolean function f, the polynomial p is linear in each of its variables because $x^2 = 1$ for $x \in \{-1,1\}$. It is known that every Boolean function has a unique representation as a polynomial [12]. This representation is derived by using the Hadamard matrix, as described by Theorem 5 below.

Definition: A Hadamard matrix of order m, to be denoted by H_m, is an $m \times m$ matrix of $+1$'s and -1's such that

$$H_m H_m^T = m I_m \tag{2}$$

where I_m is the $m \times m$ identity matrix. This is equivalent to saying that any two rows of H are orthogonal.

Hadamard matrices of order 2^k exist for all $k \geq 0$. The so called Sylvester construction is as follows [13]:

$$H_1 = [1]$$

$$H_2 = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$$

$$H_{2^{n+1}} = \begin{bmatrix} H_{2^n} & H_{2^n} \\ H_{2^n} & -H_{2^n} \end{bmatrix}. \tag{3}$$

Theorem 5: Let f be a Boolean function of order n. Let p be a polynomial equivalent to f. Let A_{2^n} denote the vector of coefficients of p. Let P_{2^n} denote the vector of the 2^n values of p (and f). Then

1) The polynomial p always exists and is unique.
2) The coefficients of p are computed as follows,

$$A_{2^n} = \frac{1}{2^n} H_{2^n} P_{2^n}.$$

Proof (idea): The proof is constructive. The idea is to compute A_{2^n} by solving a system of linear equations. □

Example: Consider the function $f(x_1, x_2) = x_1 \wedge x_2$. Then $f(1,1) = 1$, $f(1,-1) = 1$, $f(-1,1) = 1$ and $f(-1,-1) = -1$. By Theorem 5

$$f(x_1, x_2) = \frac{1}{2}(1 + x_1 + x_2 - x_1 x_2).$$

Notation: The entries of the vector A are denoted by $\{a_\alpha | \alpha \in \{0,1\}^n\}$ and are called the spectral representation of a function. Note that a_α is the coefficient of X^α in the polynomial representation where $X^\alpha = x_1^{\alpha_1} x_2^{\alpha_2} \cdots x_n^{\alpha_n}$. Hence, every Boolean function can be written as

$$f(X) = \sum_{\alpha \in \{0,1\}^n} a_\alpha X^\alpha.$$

The vectors corresponding to X^α will be denoted by $X_1^{\alpha_1} X_2^{\alpha_2} \cdots X_n^{\alpha_n}$.

Acknowledgment

The authors would like to thank Amir Dembo for introducing us to the problem.

References

[1] Y. S. Abu-Mostafa and J. M. St. Jacques, "Information capacity of the Hopfield model," *IEEE Trans. Inform. Theory*, IT-31 no. 4, pp. 461–464, July 1985.
[2] M. A. Akra, "On the analysis of the Hopfield network: a geometric approach," Master's thesis, Massachusetts Inst. of Technol., Cambridge, 1988.
[3] J. Bruck, "Computing with networks of threshold elements," Ph.D. thesis, Stanford Univ., Stanford, CA, 1989.
[4] J. Bruck, "Harmonic analysis of polynomial threshold functions," *SIAM J. Discrete Mathematics*, Feb. 1990.
[5] J. Bruck and J. W. Goodman, "A generalized convergence theorem for neural networks," *IEEE Trans. Inform. Theory*, vol. 34, no. 5, pp. 1089–1092, Sept. 1988.

[6] J. Bruck and J. Sanz, "A study on neural networks," *International Journal of Intelligent Systems*, vol. 3, pp. 59–75, 1988.
[7] E. Goles, F. Fogelman, and D. Pellegrin, "Decreasing energy functions as a tool for studying threshold networks," *Discrete Applied Mathematics*, vol. 12, pp. 261–277, 1985.
[8] S. W. Golomb, *Shift Register Sequences*. Laguna Hills, CA: Aegean Park Press, 1982.
[9] D. O. Hebb, *The Organization of Behavior*. New York: Wiley, 1949.
[10] J. J. Hopfield, Neural networks and physical systems with emergent collective computational abilities," *Proc. National Academy of Sciences USA*, vol. 79, 1982, pp. 2554–2558.
[11] J. Komolós and R. Paturi, "Effect of connectivity in associative memory models," in *Proc. 29th IEEE Symp. on Foundations Comput. Sci.*, 1988, pp. 138–149.
[12] R. J. Lechner, "Harmonic analysis of switching functions," A. Mukhopadhyay, Ed., *Recent Development in Switching Theory*. New York: Academic, 1971.
[13] F. J. MacWilliams and N. J. A. Sloane, *The Theory of Error-Correcting Codes*. New York: North-Holland, 1977.
[14] R. J. McEliece and E. C. Posner, "The number of stable points of an infinite-range spin glass," unpublished.
[15] R. J. McEliece, E. C. Posner, E. R. Rodemich, and S. S. Venkatesh, The capacity of the Hopfield associative memory, *IEEE Trans. Inform. Theory*, vol. 33, no. 4, pp. 461–482, July 1987.
[16] D. E. Muller, "Application of Boolean algebra to switching circuit design and to error detection," *IRE Trans. Elect. Computers*, vol. 3, pp. 6–12, 1954.
[17] I. Ninomiya, "A theory of coordinate representation of switching functions," *Memoirs of the Faculty of Engineering*, Nagoya University, Nagoya, Japan, vol. 10, 1958, pp. 175–190.
[18] F. Tanaka and S. F. Edwards, "Analytic theory of the ground state properties of a spin glass: I. ising spin glass," *J. Phys. F: Metal Phys.*, vol. 10, pp. 2769–2778, 1980.

The True Dimension of Certain Binary Goppa Codes

MARCEL VAN DER VLUGT

***Abstract*—By applying a result from algebraic geometry, due to E. Bombieri, the true dimension of certain binary Goppa codes is calculated. The results lead in many cases to an improvement of the usual lower bound for the dimension.**

I. Introduction

Let $L = \boldsymbol{F}_{2^m} = \{P_1, P_2, \cdots, P_{2^m}\}$ and $g(z)$ a polynomial over $\boldsymbol{F}_{2^m}$ of degree $r(\geq 1)$ without zeros in L. We consider the binary Goppa code $\Gamma(L, g)$ of length $n = 2^m$, consisting of words $(c_1, c_2, \cdots, c_n)$ such that

$$\sum_{i=1}^{n} \frac{c_i}{z - P_i} \equiv 0 \,(\mathrm{mod}\, g(z)).$$

For the dimension k of $\Gamma(L, g)$ we have the inequality $k \geq n - mr$ (see [1], [2], [3]). Let $g(z) = g_1^2(z)g_2(z)$ be the unique factorization with $g_1(z), g_2(z) \in \boldsymbol{F}_{2^m}[z]$ of degree r_1, r_2 respectively and $g_2(z)$ squarefree. It is well known (see [1],[2],[3]) that the minimum distance $d_{\min}$ satisfies $d_{\min} \geq 2(r_1 + r_2) + 1$. The purpose of this article is to prove that for polynomials $g(z)$ with

$$(-2 + r + t) < \frac{2^m + 1}{\sqrt{2^m}}$$

where t is the number of distinct roots of $g(z)$ in an algebraic closure of $\boldsymbol{F}_2$, the following holds:

$$\dim \Gamma(L, g) = n - m(r_1 + r_2).$$

Manuscript received October 24, 1988; revised July 7, 1989.
The author is with the Department of Mathematics and Informatics, Leiden University, P.O. Box 9512, 2300 RA Leiden, The Netherlands.
IEEE Log Number 8933843.

This result is similar to a result for BCH codes (see [1], Ch. 9, Sect. 3, Cor. 8).

II. The Trace Operation and Bombieri's Inequality

First we consider the dual code $\Gamma(L, g)^{\perp}$.

The code $\Gamma(L, g)^{\perp}$ is closely connected with a generalized Reed–Solomon code, namely the code that has as words

$$\left(\frac{f(P_1)}{g(P_1)}, \frac{f(P_2)}{g(P_2)}, \cdots, \frac{f(P_n)}{g(P_n)}\right)$$

where $f(z) \in \boldsymbol{F}_{2^m}[z]$ of degree $< r$.

This is a $(n = 2^m, r, 2^m - r + 1)$ code over $\boldsymbol{F}_{2^m}$.

Now if we look at the words

$$\left(\mathrm{tr}\left(\frac{f(P_1)}{g(P_1)}\right), \mathrm{tr}\left(\frac{f(P_2)}{g(P_2)}\right), \cdots, \mathrm{tr}\left(\frac{f(P_n)}{g(P_n)}\right)\right)$$

where tr is the trace mapping from $\boldsymbol{F}_{2^m}$ to $\boldsymbol{F}_2$, we get a binary linear code. We call this binary code the trace code induced by the generalized Reed–Solomon code. From [1] (ch. 12, sect. 3, Th. 5) it follows that this trace code is precisely $\Gamma(L, g)^{\perp}$. So to determine the dimension of $\Gamma(L, g)^{\perp}$ it suffices to know how many different words we get by performing the trace operation to the words of the generalized Reed–Solomon code.

This calculation relies heavily on Bombieri's result [4]:

IF $f(z)/g(z)$ (f and g as previously mentioned) has poles Q_i, $i = 1, \cdots, l$, with multiplicities n_i and IF $Y^2 - Y = f(z)/g(z)$ has no solutions in $\bar{\boldsymbol{F}}_2(z)$, the field of rational functions over an algebraic closure of $\boldsymbol{F}_2$, THEN

$$\left|\sum_{P \in \boldsymbol{P}(\boldsymbol{F}_{2^m})} (-1)^{\mathrm{tr}(f(P)/g(P))}\right| \leq \left(-2 + l + \sum_{i=1}^{l} n_i\right)\sqrt{2^m}$$

where the summation runs over all points P of the projective line $\boldsymbol{P}(\boldsymbol{F}_{2^m})$ over $\boldsymbol{F}_{2^m}$.

III. The Main Result

By simple substitution we can prove Lemma 1.

Lemma 1: If $f(z) = g_2(z)(h^2(z) - g_1(z)h(z))$ where $h(z) \in \boldsymbol{F}_{2^m}[z]$ of degree $< r_1$, then $h(z)/g_1(z) \in \boldsymbol{F}_{2^m}(z)$ is a solution of $Y^2 - Y = f(z)/g(z)$.

For different $h(z)$ we get different $f(z)$. Now for the $f(z)$ as in Lemma 1 it follows that

$$\mathrm{tr}\left(\frac{f(P)}{g(P)}\right) = \mathrm{tr}\left(\left(\frac{h(P)}{g_1(P)}\right)^2\right) - \mathrm{tr}\left(\frac{h(P)}{g_1(P)}\right) = 0$$

for all $P \in \boldsymbol{F}_{2^m}$.

So at least $(2^m)^{r_1}$ polynomials $f(z)$ induce the zero word in the trace code. This leads to Corollary 1.

Corollary 1: Let $\Gamma(L, g)$ be a binary Goppa code with Goppa polynomial $g(z) \in \boldsymbol{F}_{2^m}[z]$ and $g(z) = g_1^2(z)g_2(z)$ with $g_1(z)$, $g_2(z) \in \boldsymbol{F}_{2^m}[z]$ of degree r_1, r_2 respectively and $g_2(z)$ squarefree, then

$$\dim \Gamma(L, g)^{\perp} \leq m(r_1 + r_2) \text{ and } \dim \Gamma(L, g) \geq n - m(r_1 + r_2).$$

The last inequality is already an improvement of the usual lower bound if $r_1 > 0$ and $g_1(z)$ has multiple roots. For the other inequality for the dimension of $\Gamma(L, g)^{\perp}$ we first prove Lemma 2.

Lemma 2: If $f(z) \in \boldsymbol{F}_{2^m}[z]$, with $f \neq 0$ of degree $< r$, and $Y^2 - Y = f(z)/g(z)$ is solvable in $\bar{\boldsymbol{F}}_2(z)$ then degree $f \geq r_1 + r_2$.

0018-9448/90/0300-0397$01.00 ©1990 IEEE