Discrete Mathematics 114 (1993) 105–113
North-Holland

Write-isolated memories (WIMs)*

Gérard D. Cohen and G. Zemor
ENST and CNRS URA 251, 46, rue Barrault - 75634 Paris, Cedex 13, France

Received 18 December 1988
Revised 13 June 1989

Abstract

Cohen, G.D. and G. Zemor, Write-isolated memories (WIMs), Discrete Mathematics 114 (1993) 105–113.

A write-isolated memory (WIM) is a binary storage medium on which no change of two consecutive positions is allowed when updating the information stored. We prove that the optimal rate for writing on a WIM is $\log_2(1+\sqrt{5})/2 = 0.69$. We give asymptotic constructions achieving 0.6.

1. Introduction

A write-isolated memory (WIM) is a binary storage medium on which no change of two consecutive positions is allowed when updating the information stored.

This constraint is dictated by the current technology for writing on some digital optical disks, as indicated by Vinck [14], who discovered the problem we consider here. We assume that we have a WIM with n positions, which we use for writing one message among the M possible ones. We want to be able to continue the process indefinitely, under the above constraint. The problem is: What is asymptotically the maximum achievable rate R of the WIM, defined as (log to the base 2)

$$R := (1/n)\log M \ ?$$

A similar question was considered by Kløve (which he refers to as Robinson's problem, see [13]) in relation to coding for fluorescent ink bars.

More generally, analogous problems dealing with writing on memories under constraints have been considered by many authors (see e.g. [5, 7, 12, 15]) for write-once memories, for [2, 6, 10] write-unidirectional memories, [1] for a general model of write-efficient memories.

Correspondence to: Gérard D. Cohen, Ecole Nationale Superieure des Telecomm., 46 rue Barrault, Paris 75634, Cedex 13, France.

* Presented at the French–Israeli Conference on Combinatorics and Algorithms, November 14–17, 1988, Jerusalem, Israël.

Four cases can be distinguished, arising when

> *the encoder (writer) and/or the decoder (reader) are informed or uninformed about the previous state of the memory.*

2. General bounds

Let us start with a simple example giving a lower bound on R, valid in all four cases of our problem.

Example 2.1. Take a WIM of size n, with positions numbered $1, 2, \ldots, n$, and use only odd positions for writing. Then the nonadjacency constraint is clearly satisfied. This yields a construction with $R = 1/2$ (asymptotically).

Let us give now an upper bound, also valid in all four cases. Let V_n be the set of binary n-tuples, $V_n(t)$ the set of possible states of the memory after t utilizations. Consider the following directed graph:

$$G = (V, E), \quad \text{with } V = V_n(t) \cup V_n(t+1)$$

and

$$E = \{(i, j), \text{ where } i \in V_n(t),\ j \in V_n(t+1) \text{ and } i \to j \text{ is allowed}\}.$$

Now the following is clear:

$$M \leqslant \operatorname*{Max}_i v(i), \quad i \in V_n(t),$$

where $v(i)$ is the valency of i (indeed any state i can be updated to at most $v(i)$ states j).

Set

$$F_n := \{(x_1, x_2, \ldots, x_n) \in V_n \colon x_i x_{i+1} = 0 \text{ for } i = 0, 1, \ldots, n-1\}.$$

F_n will be written F for short, when no confusion on the length can occur.

Then $|F| = f_n$ (nth Fibbonacci number with $f_0 = 1$, $f_1 = 2$) is easily checked by induction. Hence, $v(i) = f_n$, since $i \to j$ is allowed iff $i + j \in F$, where the addition is bitwise mod 2. Hence, we get $M \leqslant f_n$, and, using the well-known approximation of $f_n : f_n \cong 2^{\rho n}$, where $\rho := (1 + \sqrt{5})/2)$ is the golden ratio, we get

$$R \leqslant \log((1 + \sqrt{5})/2).$$

Let us summarize these results.

Proposition 2.2. *The optimal rate for writing on a WIM (in all four cases) satisfies*

$$1/2 \leqslant R \leqslant \log((1 + \sqrt{5})/2) \cong 0.69.$$

From now on, we consider only the following case:

> *Encoder knows the previous state of the memory (i.e. can read before writing), but decoder does not).*

3. Coding with blocks

We now present a coding strategy based on the notion of good blocks, which we define as follows.

For $x \in V_n$, set $F(x) := F + x := \{f + x, f \in F\}$. Hence, $F(0) = F$. In words, $F(x)$ is the set of states reachable from x. We call $F(x)$ the *F-set centered at x*.

A block $B \subset V_n$ is called *good* if

$$\bigcup_{b \in B} F(b) = V_n. \tag{1}$$

That is, V_n is covered by F-sets centered on the elements of B.

Proposition 3.1. *If a block B is good, any translate $B+t$, $t \in V_n$, of B is also good.*

Proof. $$\bigcup_{b' \in B+t} F(b') = \left(\bigcup_{b \in B} F(b) \right) + t = V_n. \qquad \square$$

With a different phrasing, the following result is already in [3].

Proposition 3.2. *If B is good, then*

$$\forall x \in V_n \quad \exists b \in B \; \exists f \in F\colon \; x + f = b.$$

In other words, starting from any state x of the memory, there exists an allowed transition f which transforms x into an element of B (say b).

Proof. By (1), for all x, there is an i s.t. x is in $F(b_i)$, i.e. $x = b_i + f$ for some f in F. $\square$

Proposition 3.3. *If $B_0, B_1, \ldots, B_{M-1}$ are pairwise disjoint good blocks, they yield a WIM-code of size M.*

Proof. Put the M messages to be coded in 1–1 correspondence with the blocks. By Proposition 3.2, whatever the state of the WIM is, updating will be possible to any message. $\square$

Example 3.4. $n = 3$. Set $B_0 = \{000, 111\}$. Then B_0 is good, since

$$F(000) = F = \{000, 001, 010, 100, 101\},$$

$$F(111) = \{111, 110, 101, 011, 010\} = \bar{F}$$

and $F \cup \bar{F} = V_3$; so, (1) holds.

By Proposition 3.1, the following blocks are also good:

$$B_1 = B_0 + 001 = \{001, 110\}, \qquad B_2 = B_0 + 010 = \{010, 101\},$$

$$B_3 = B_0 + 100 = \{100, 011\}.$$

This yields, by Proposition 3.3, since B_0, B_1, B_2, B_3 form a partition of V_3, a WIM-code with 4 code words, i.e. rate 2/3. Unfortunately, we could not use this example for an infinite construction.

Let us visualize how the coding works: suppose we are in state $x=010$, representing message 2. The following are the allowed transitions for writing messages 0, 1, 2, 3, respectively:

$$010\to000,\ 010\to110,\ 010\to010,\ 010\to011.$$

4. $R=\log\rho$

We shall now close the gap between the two bounds in Proposition 2.2 by showing that the actual achievable rate is $R=\log\rho\cong0.69$.

This will a fortiori give the achievable rate in the more favorable case when writer and reader know the previous state. This result is not difficult to prove in a probabilistic (nonconstructive) way. We shall rather give here a 'semi-constructive' proof, which also helps in obtaining good codes.

In view of Proposition 3.3, it is intuitive to look for 'small' good blocks, so as to be able to pack many of them (e.g. by translation) in V_n.

In fact, we shall first prove the existence of small *good subgroups* of V_n (i.e. good blocks which are groups). Then the second step, finding pairwise disjoint good blocks, becomes simple: if G is a good subgroup, $|G|=2^k$, then there are 2^{n-k} pairwise disjoint good blocks, namely the cosets of G (see Example 3.4). To that end, we use Theorem 1 of [4], which is established for coverings of V_n by Hamming spheres centered on the elements of a group (group coverings). Its extension to group covering by tiles other than spheres is easy and already mentioned in [4]; so, we shall not give its proof, which is based on a 'group' greedy algorithm:

Proposition 4.1. *There exists a group covering G of V_n with 2^k sets $F(g_i)$, $g_i\in G$, with*

$$k\leqslant n-\log f_n+\log n+O(1).$$

This gives

$$k\leqslant n(1-\rho)+\log n+O(1)$$

and

$$M=2^{n-k}\geqslant 2^{\rho n}/n\,O(1).$$

Corollary 4.2. $R=\log\rho$.

Dropping the group condition, one can obtain still smaller good blocks, but this will, of course, not improve the rate. We shall nevertheless give some details, since they shed more light on the possible constructions.

Let $B=\{b_1, b_2, \ldots, b_m\}$ be a good block of minimal size (i.e., by (1), a minimal covering of V_n by F-sets). Consider the hypergraph $H=(V,E)$, where $V=V_n$ and

$E=\{F(x), x\in V_n\}$. Then H is clearly f_n-uniform and f_n-regular (i.e. $|\{x:y\in F(x)\}|=f_n$ for all y). Thus, by a Theorem of Lovàsz [11], there exists a covering with

$$2^n/f_n \leqslant m \leqslant (2^n/f_n)(1+\log f_n), \tag{2}$$

where the lower bound is the well-known covering bound.

5. Explicit constructions

5.1. Length $n=6$

We have $F_6=21$; So, by (2), a good block has size at least $\lceil 2^6/21 \rceil=4$. An exhaustive search for minimal good blocks of length 6 was done by Busson [3] with a computer; it turned out that exactly 64 minimal good blocks exist. One good subgroup is

$$B_6=\begin{matrix}000000\\100110\\011001\\111111.\end{matrix}$$

Now to get a WIM-code, just take the $16=2^4$ cosets of this block. Its rate is $R=4/6\cong 0.66$.

The next natural question (from a constructive point of view) is: How do we build WIM-codes of arbitrary length? One answer is to try and concatenate small WIMs. For instance, we can get a WIM-code of length $n=7k$ by dividing a memory of length $7k$ into k blocks of size 6, and use the above length-6-WIM-code on every block; unfortunately, we need to sacrifice a position between each block to ensure the 'isolation' condition.

Still, with this somewhat unrefined approach, we achieve a rate of $R=4/7=0.57$ for an explicitly constructed WIM-code of arbitrary length.

5.2. A more efficient concatenation

We will show here a method for concatenating the length-6 example to obtain an explicit code with rate 0.6.

Let εF_n ($F_n\varepsilon'$; $\varepsilon F_n\varepsilon'$) denote the set of elements of F_n starting with ε ($\varepsilon=0$ or $\varepsilon=1$) (ending with ε'; respectively, starting with ε and ending with ε'). If x and y are two words of lengths, say n and n', we denote by $x{:}y$ the concatenation of x and y (so that $x{:}y\in V_{n+n'}$).

In a similar fashion if A and B are 2 subsets of V_n and $V_{n'}$, we denote by $A{:}B$ the set of concatenated words $x{:}y$, where $x\in A$ and $y\in B$.

In the following proofs we shall have to bear in mind the obvious.

Proposition 5.1. $F_{n+n'}=(F_n0{:}F'_n)\cup(F_n1{:}0F'_n)=(F_n{:}0F_{n'})\cup(F_n0{:}1F'_n)$.

The purpose of the following construction is to obtain a good block in length $5k$, with $4k$ elements.

To do so, we start by stripping the above B_6 in length 6 of its first column to obtain a good subgroup B_5 in length 5:

$$B_5 = \begin{matrix} 00000 \\ 00110 \\ 11001 \\ 11111 \end{matrix} := \begin{matrix} \beta_0 \\ \beta_1 \\ \beta_2 \\ \beta_3 . \end{matrix}$$

From now on, elements in V_5 will be denoted by greek letters.

Now we want a procedure that will give us a good block of length $n+5$, starting from a good one of length n. We will not do it for any good block, but for blocks B_n satisfying the following 3 properties:

(i) B_n is a group,

(ii) the last 5 coordinates of all the elements of B_n coincide with a word β_i of B_5,

(iii) for every $\beta_i \in B_5$, the set of words of B_n ending with β_i is, when restricted to its $n-5$ first coordinates, a good block in length $n-5$.

Given a block B_n, let B_{n+5} be defined by

$$B_{n+5} := \begin{matrix} B_n : \beta_0 \\ (t+B_n) : \beta_1 \\ B_n : \beta_2 \\ (t+B_n) : \beta_3 , \end{matrix}$$

where $t := (00..0:11000)$ is in V_n.

Now the point of all this is to obtain the following result.

Proposition 5.2. *If B_n is good, in length n, and verifies* (i)–(iii), *then B_{n+5} is also good and verifies* (i)–(iii).

Proof. B_{n+5} verifies (i)–(iii) (easy).

We must now prove that B_{n+5} is good, that is, $B_{n+5} + F_{n+5} = V_{n+5}$.

Point 1: Check that $B_5 + 0F_5 = V_5 \backslash A$, where $A = \{\theta, \alpha, \bar{\theta}, \bar{\alpha}\}$, with

$\theta = 10010,$

$\alpha = 10100,$

$\bar{\theta} = 01101,$

$\bar{\alpha} = 01011.$

Point 2: All elements of the form $x{:}g$, where $x \in V_n$ and $g \in V_5 \backslash A$, are in $B_{n+5} + F_{n+5}$. To check that, recall that B_n and $t + B_n$ are good, and that $F_n{:}0F_5 \subset F_{n+5}$. We can, therefore, write, for $g \in V_5 \backslash A$,

$$\gamma = b_i + \lambda_5, \quad \text{with } \beta_i \text{ in } B_5 \text{ and } \lambda_5 \text{ in } 0F_5.$$

Set

$$\varepsilon_0 = \varepsilon_2 = 0,\ \varepsilon_1 = \varepsilon_3 = 1.$$

Then write $x + \varepsilon_i t = b + f$ for some b in B_n and f in F_n (remember B_n is good). Then

$$x:\gamma = ((\varepsilon_i t + b):\beta_i) + (f:\lambda_5), \quad \text{with } (\varepsilon_i t + b):\beta_i \in B_{n+5} \text{ and } f:\lambda_5 \in F_{n+5}.$$

Point 3: The last case to study is $x \in V_n$ and $\gamma \in A$, for which we prove that

$$x:\gamma \in B_{n+5} + F_{n+5}.$$

Let us do it first for $\gamma = \theta$ and $\gamma = \alpha$. Note that

$$\begin{aligned} &\theta \in \beta_0 + F_5 \quad \text{and} \quad \theta \in \beta_1 + F_5, \\ &\alpha \in \beta_0 + F_5 \quad \text{and} \quad \alpha \in b_1 + F_5. \end{aligned} \tag{3}$$

Since B_n is good, $x \in B_n + F_n$.

The problem is that concatenating an element of F_n with an element of F_5 will not necessarily yield an element of F_{n+5}. To avoid this, we prove that

$$x \in B_n + F_n 0 \quad \text{or} \quad x \in (t + B_n + F_n 0).$$

The only x's for which this is not obvious are the elements of $B_n + F_n 1$ (the others are in $B_n + F_n 0$). Since B_n is a group, we need only show that any element of $F_n 1$ belongs to $B_n + F_n 0$ or to $t + B_n + F_n 0$. Now the last 5 coordinates of every element of $F_n 1$ must be in $F_5 1$, where

$$F_5 1 = \{00001, 10001, 0001, 00101, 10101\}.$$

Since B_n verifies (iii) and since $F_{n-5}:0F_5 0 \subset F_n 0$, we will be finished when we show that any element of $F_5 1$ can be written as an element of $B_5 + 0F_5 0$ or $\tau + B_5 + 0F_5 0$, where $\tau = (11000)$ (the last 5 coordinates of t).

That is checked easily:

$$\begin{aligned} 00001 &= \tau + b_2 \\ 10001 &= b_2 + 01000 \\ 01001 &= \tau + b_2 + 01000 \\ 00101 &= \tau + b_2 + 00100 \\ 10101 &= b_3 + 01010. \end{aligned}$$

Summarizing, we have proved using properties (i)–(iii), that the troublesome elements of $F_n 1$ can be expressed as elements of $B_n + F_n 0$ or of $t + B_n + F_n 0$. Therefore, any element of V_n is in $B_n + F_n 0$ or $t + B_n + F_n 0$, so that with property (iii) of θ and α we get

$$x:\theta \in (B_n + F_n 0):(\beta_0 + F_5)$$

or

$$x:\theta \in (t + B_n + F_n 0):(\beta_1 + F_5).$$

This means $x:\theta \in B_{n+5} + F_{n+5}$, and the same holds for $x:\alpha$.

The cases of $x{:}\bar{\theta}$ and $x{:}\bar{\alpha}$ are easily deducible from the above by noting that

$$\{\bar{\theta},\bar{\alpha}\}=\{\theta,\alpha\}+(11111)$$

and

$$\{\beta_2,\beta_3\}=\{\beta_0,\beta_1\}+(11111),$$

which concludes the proof that B_{n+5} is good. □

We can, therefore, use this theorem to construct inductively, starting from B_5 — check that it verifies (i)–(iii) — a good subgroup of length $5k$ and size $4k$. It has 2^{3k} cosets, which gives by Proposition 3.3, a WIM-code of rate

$$R=3k/5k=0.6.$$

6. Nonlinear WIM-coding and perspectives for further research

Until now we considered only WIM-codes obtained by taking the cosets of a good subgroup of V_n.

Now suppose that we have a good block B, which is not a group (obtained, say, with the help of a computer). To construct a WIM-code, we need disjoint good blocks, and the only systematic way we see of getting those from B is to search for a set of pairwise disjoint translates $B+t$ of B.

In other words, we see easily that a WIM-code can be constructed with:

(1) a good block B,

(2) a set of translations $T\subset V_n$ such that $B+B$ and $T+T$ are disjoint.

Note that looking for such a set T can be thought of as a classical coding problem: indeed, T is a code correcting a set B of parasite noise elements.

This general coding problem was considered by Deza [8], where he proved that the sets B of noises (of a given cardinality), for which the largest B-correcting codes T exist are either the most 'scattered', or the most 'dense' sets, that is, they are either included in a subgroup of V_n, or in a Hamming sphere.

In fact, one can easily convince oneself that B must have diameter at least $\lfloor n/2\rfloor$, since F has diameter at most $\lceil n/2\rceil$. In conclusion, small good blocks should 'resemble' subgroups.

Acknowledgment

We thank F. Busson for helpful discussions and for letting us include his computer results, a referee for improving Section 6 and AFIRST for financial support.

References

[1] R. Ahlswede and Z. Zhang, Coding for write-efficient memories, Inform. and Control 83 (1989) 80–87.
[2] J.M. Borden, Coding for write-unidirectional memories, preprint.

[3] F. Busson, Un code pour une WEM, rapport de stage ENST.
[4] G. Cohen and P. Frankl, Good coverings of Hamming spaces with spheres, Discrete Math. 56 (1985) 125–131.
[5] G. Cohen, P. Godlewski and F. Merkx, Linear block codes for write-once memories, IEEE Trans. Inform. Theory IT-29 (1983) 731–739.
[6] G. Cohen and G. Simonyi, Coding for write-unidirectional memories and conflict resolution, Discrete Appl. Math, 24 (1989) 103–114.
[7] G. Cohen and G. Zemor, An application of combinatorial group theory to coding, Ars combin. 3A (1987) 81–89.
[8] M. Deza, Best additive arbitrary noise with given size, Research report CORR 75/20, University of Waterloo, Canada.
[9] T. Kløve, On Robinson's coding problem, IEEE Trans. Inform. Theory IT-29 (1983) 450–454.
[10] A. Kuznetsov, On the capacity of write-unidirectional memory in the presence of random errors, preprint.
[11] L. Lovàsz, On the ratio of optimal integral and fractional covers, Discrete Math. 13 (1975) 383–390.
[12] R.L. Rivest and A. Shamir, How to reuse a "write-once" memory, Inform. and Control 55 (1982) 1–19.
[13] J.P. Robinson, An asymmetric error-correcting ternary code, IEEE Trans. Inform. Theory IT-24, (1978) 258–261.
[14] A. Vinck, personal communication.
[15] J.K. Wolf, A.D. Wyner, J. Ziv and J. Korner, Coding for write-once memory, AT & T Bell Lab.-Tech. J. 63 (1984) 1089–1112.