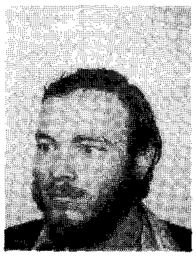
Womcodes constructed with projective geometries

« Womcodes » construits à partir de géométries projectives



Frans MERKX [1]

École Nationale Supérieure de Télécommunications (ENST), 46, rue Barrault, 75013 PARIS

Étudiant à l'Université de Technologie à Eindhoven (Pays-Bas) dans le domaine des mathématiques discrètes.

SUMMARY

We consider storage media which consist of a number of write-once bit positions (wits). A wit initially contains a "0", that may be irreversibly overwritten with a "1".

It was shown by Rivest and Shamir [5] that, by coding techniques one can reuse such a write-once memory (wom) up to a very high rate. We present two new cyclic womcodes, based on PG (2,2) and PG (3,2) respectively, which attain the RS-bound. These codes can be decoded with a decoding algorithm for Hamming codes. Some other high-rate womcodes, derived from those above, are discussed.

KEY WORDS

Error-correcting codes, finite geometry, numerical storage media.

RÉSUMÉ

Nous considérons des mémoires constituées de positions permettant l'écriture irréversible d'un bit (wits). Un wit contient initialement un zéro, qui peut être définitivement transformé en un. Nous utilisons des techniques de codage pour réutiliser ces mémoires à écriture unique avec un rendement élevé.

MOTS CLÉS

Codes correcteurs d'erreurs, géométrie finie, mémoires numériques.

[1] During this research, the author was with the École Nationale Supérieure de télécommunication, Paris.

CONTENTS

1. Introduction

We consider storage media which consist of a number of write-once bit positions (wits). A wit initially contains a "0", that may be irreversibly overwritten with a "1". We call such a storage medium a "write-once memory" or wom.

Examples of woms are punched cards or digital optical disks.

In their pioneering paper on this subject, Rivest and Shamir [5] showed that it is possible to use a wom several times, by using "womcodes".

They gave many examples of womcodes, and show that the "capacity" (defined later) of a womcode is greater than the number of wits. They also derive asymptotic results for this capacity.

The following coding scheme was their prime "motivating example".

Example 1.1: We write two times 2 bits in a memory of 3 wits, as follows:

Message x	First writing $r(x)$	Second writing $r'(x)$
0 0	0 0 0	1 1 1
0 1	1 0 0	0 1 1
1 0	0 1 0	1 0 1
1 1	0 0 1	1 1 0

This scheme must be interpreted as follows:

The first time we receive a message x, we write $r(x)$.

Later, we will receive for the second time a message, say y. If $x=y$ we don't change the memory, if $x \neq y$ we change the memory state to $r'(y)$, by only changing 0's to 1's. Without coding, we would have to use 4 wits.

Remark that after the second writing we lost the information on the first message.

2. Notation

We call a coding scheme that uses n wits to represent t times one out of v messages (i.e. write once and change $t-1$ times) a $\langle v \rangle^t/n$ womcode. More general, with a $\langle v_1, v_2, \ldots, v_t \rangle/n$ womcode the message in the i'th generation can be one of a set V_i of v_i messages.

Such a womcode must have the following properties:

1. Each memory content that occurs must determine uniquely the last received message, and

2. for each memory content $x \in \{0, 1\}^n$ occuring in the s'th generation $(s<t)$ it must be possible to encode all sequences of messages $(m_{s+1}, \ldots, m_t)$, $m_i \in V_i$, such that $x_s \leqq x_{s+1} \leqq \ldots \leqq x_t$ (componentwise).

The womcode is determined by giving for all possible memory states and new messages the new memory state (update function [5]).

$w(\langle v_1, v_2, \ldots, v_t \rangle)$ denotes the least n for which a $\langle v_1, v_2, \ldots, v_t \rangle/n$ womcode exists, and, of course, $w(\langle v \rangle^t) = w(\langle v, v, \ldots, v \rangle)$.

Rivest and Shamir [5] derive a lower bound for $w(\langle v \rangle^t)$. This can be easily extended to a lower bound for $w(\langle v_1, v_2, \ldots, v_t \rangle)$, *see* [4], which we shall refer to as the RS-bound.

The capacity C and the rate R of a $\langle v_1, v_2, \ldots, v_t \rangle/n$ womcode are defined as $C := \log(v_1 . v_2 . \ldots, v_t)$, $R := C/n$.

For example, the $\langle 4 \rangle^2/3$ womcode of example 1.1 has $C=4$ and $R=1.33\ldots(=1.33\ldots$ bit per wit).

The following two examples of womcodes are described in [5].

Example 2.1: A $\langle 5 \rangle^3/5$ cyclic womcode (Rate $=1.39\ldots$), constructed by D. Klarner.

Message 1 is represented by 10000 in the first generation, either 01001 or 00110 in the second, and one of 01111, 10110 or 11001 in the third generation.

Message i, $1 \leqq i \leqq 5$ is represented by a cyclic shift over $i-1$ positions of the words for the first message (since 5 is prime they are all distinct).

Example 2.2: A $\langle 7 \rangle^4/7$ cyclic womcode (rate $=1.60\ldots$), constructed by D. Leavitt, by extending the method of example 2.1 (to appear).

We will present a new cyclic $\langle 7 \rangle^4/7$ womcode. The first, second and fourth generations are equivalent to these of Leavitt's code, but the third generation is different (private communication). We will use a projective geometry PG (2, 2) or 2-(7, 3, 1) symmetric block design (Steiner triple system) or Fano plane.

These are different names for the same object, described in Figure 2.3.

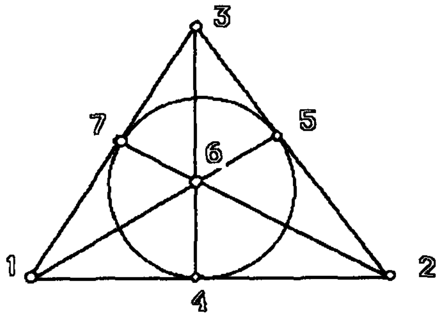


Fig. 2.3. – The Fano plane.

This picture should be interpreted as follows: The plane consists of seven points (numbered 1, 2, . . . , 7) and seven lines, each containing three points (the six lines together with the circle in Figure 2.3). Remark that any pair of points is on exactly one line, and any two lines intersect in exactly one point.

With the enumeration as in Figure 2.3, the incidence vectors of lines with the points of the plane are the cyclic shifts of 1101000.

In what follows, we shall identify the seven points of the plane with the wits of a seven-wit memory.

As a consequence, every memory content can be identified with a configuration of points in the plane (i.e. the points for which the corresponding wit contains a 1).

3. A $\langle 7 \rangle^4/7$ womcode

The $\langle 7 \rangle^4/7$ womcode which we propose is described in terms of configurations of points in the Fano plane in Figure 3.1 below.

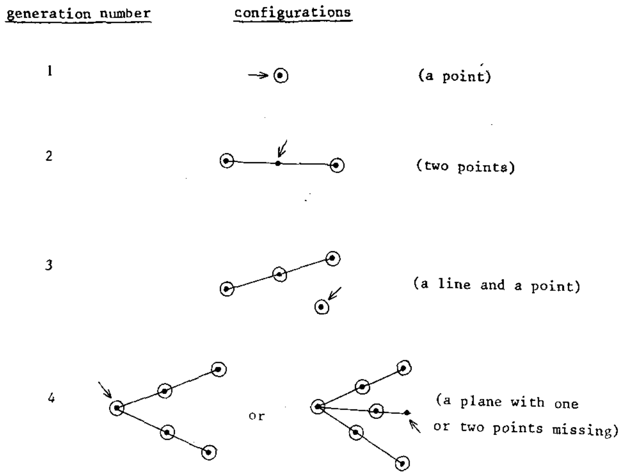


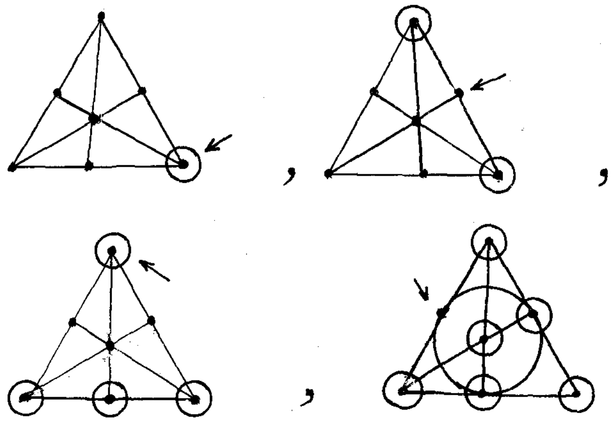
Fig. 3.1. – A description of a $\langle 7 \rangle^4/7$ womcode.

Figure 3.1 should be interpreted as follows: possible memory states of the womcode are described by their corresponding configurations [The encircled points in (3.1)]. Since there are seven messages, they are identified with the seven points in the Fano plane. For each configuration the message (point) represented is indicated by an arrow.

Using the properties of the Fano plane given after (2.3) it is easy to check that Figure 3.1 describes indeed a $\langle 7 \rangle^4/7$ womcode:

1. Each memory state determines uniquely a message point, and
2. For each configuration in generation i, and for each message point received, it is possible, by adding one or two points, to find a configuration in generation $i+1$ representing the received message point. (NB: if the same message is received twice, the memory state is not changed.)

Example 3.2: Suppose we receive the message sequence 2, 5, 3, 7. Using the womcode of (3.1), we obtain the following sequence of configurations representing them:



Corresponding to the sequence of memory contents 0100000, 0110000, 1111000, 1111110.

Decoding

For the decoding, we use the fact that the codewords of the [7, 4] binary perfect Hamming code are all the linear combinations modulo 2 of the lines of the Fano plane (*see*, for instance [3]). So the code words are 0^7, lines, symmetric differences of two lines, and the whole plane. Configurations of the womcode in (3.1) are never Hamming codeword, so they are at Hamming distance 1 from exactly one Hamming code word. Moreover, by inspection of (3.1) we see that this Hamming codeword is obtained by adding (mod 2) the message point to the configuration. So, if we denote the memory content as a Hamming codeword, the error vector yields exactly the message

(i. e. using syndrome decoding, each message corresponds with one of the seven possible non-zero syndromes).

4. More womcodes from projective geometries

Example 4.1: Consider the projective geometry PG (3, 2). It contains 15 points, 35 lines of 3 points and 15 Fano planes of 7 points.

The [15, 11] Hamming code can be seen as the collection of all linear combinations mod 2 of lines in PG (3, 2).

Now the approach of sections 3 can be generalised: it is possible to construct a $\langle 15 \rangle^7/15$ womcode which can be "decoded" by the method described in section 3, i. e. the message represented by each memory state in the womcode is just its Hamming code error, and is obtained by computing the syndrome (*see* [4] for details).

Example 4.2: Fix a line in the Fano plane. The configurations of the first two generations in (3.1), restricted to those on this line, describe a $\langle 3 \rangle^2/3$ womcode. Since a line is a PG (1, 2), this womcode could be considered as the first code of a class of womcodes, based on PG (n, 2), all having the property that they can be denoted with syndrome decoding for the Hamming code. Of course, the second code of this class is the code in section 3, the third is (4.1). These three codes are optimal in the sense that $w(\langle 3 \rangle^2) \geqq 3$, $w(\langle 3 \rangle^3) \geqq 4$, $w(\langle 7 \rangle^4) \geqq 7$, $w(\langle 7 \rangle^5) \geqq 8$, $w(\langle 15 \rangle^7) \geqq 15$, $w(\langle 15 \rangle^8) \geqq 16$, all by the RS-bound.

The next code in this class is a $\langle 31 \rangle^t/31$ womcode. The RS-bound yields $w(\langle 31 \rangle^{14}) \geqq 31$, $w(\langle 31 \rangle^{15}) \geqq 32$. We have constructed a $\langle 31 \rangle^{10}/31$ womcode, so that some sequences of length 11 cannot be encoded. However, we think that by further selection of the configurations it is possible to construct at least a $\langle 31 \rangle^{12}/31$ womcode.

An important feature of the codes described is that messages correspond to a coset of a linear error-correcting code with $(n-k) \times n$ parity check matrix H.

Encoding a new message m in a memory with state $x \in \{0, 1\}^n$ is equivalent to finding a $y \in \{0, 1\}^n$ such that $m = y . \mathrm{H}^{\mathrm{T}}$ and $y \geqq x$, componentwise. This is also described in [2], together with a dynamic programming algorithm for finding a y with minimal weight.

However, other constraints for y than having minimal weight, can be posed, such as "y must be one of the configurations of (3.1)".

In [4] some constraints are described to construct $\langle 2^{n-k}-1 \rangle^t/2^{n-k}-1$ womcodes based on the corresponding Hamming codes.

For some of these constraints the exact value of t is determined.

If turns out that, using only the minimal weight constraint, does not always (and probably only for $n=3$ or $n=7$) yield the maximal t.

5. Extensions

For more details of these extensions, *see* [4].

Extension 5.1: Consider again the code of (4.2). If we allow a 4-th message, represented by the empty set and the whole line, in the first resp. second generation, a $\langle 4 \rangle^2/3$ womcode is obtained. This is the womcode of example 1.1.

Extension 5.2: Consider the code of section 3. If, in the first, third and fourth generation, we allow an eighth message, represented by the empty set, a line or the whole plane respectively, we obtain a $\langle 8, 7, 8, 8 \rangle/7$ womcode (rate is 1.69...).

Extension 5.3: Adding to the configurations in the third generation all those consisting of three non-colinear points, it is possible to represent three more messages, giving an $\langle 8, 7, 11, 8 \rangle/7$ womcode (rate is 1.75...). 11 is best possible here.

Extension 5.4: By extending the memory of the code of section 3, respectively example 4.1 with one wit, it is possible to construct an $\langle 8, 14, 11, 8 \rangle/8$ (rate is 1.66...) and a $\langle 16 \rangle^7/16$ (rate is 1.75...) womcode, respectively.

Conclusion

We have considered memories (woms) which consist of a number of Write-Once bit positions (wits).

Using projective geometries, we have constructed codes, which make it possible to use these woms several times.

It turned out that the message, represented by the memory-content, can be seen as the syndrome of the binary Hamming code.

Acknowledgements

We would like to thank Henk Hollmann for his great help.

REFERENCES

[1] A. Fiat and A. Shamir, Generalized Write-Once Memories, *IEEE Transactions on Information Theory*, IT-30, 1984, n° 3, May 1984, pp. 470-480.

[2] C. Heegard, *An efficient encoder for optical disk codes*, preprint.

[3] J. H. Van Lint, *Introduction to coding theory*, Springer Verlag, New York, 1982.

[4] F. Merkx, *Codes for Write-Once Memories*, Rapport Interne, École Nationale Supérieure des Télécommunications, Paris, 1984.

[5] R. L. Rivest and A. Shamir, (1982), How to reuse a Write-Once Memory, *Information and Control*, 55, 1982, pp. 1-19.