Explicit Constructions of Finite-Length WOM Codes

Yeow Meng Chee[*], **Han Mao Kiah**[*], **Alexander Vardy**[*†], and **Eitan Yaakobi**[‡]

[*] School of Physical and Mathematical Sciences, Nanyang Technological University, Singapore

[†] Dept. of Electrical and Computer Engineering, University of California San Diego, La Jolla, CA 92093, USA

[‡] Dept. of Computer Science, Technion — Israel Institute of Technology, Haifa, 32000 Israel

{**ymchee,hmkiah**}**@ntu.edu.sg.edu**, **avardy@ucsd.edu**, **yaakobi@cs.technion.ac.il**

***Abstract*—*Write-once memory (WOM)* is a storage device consisting of binary cells which can only increase their levels. A t-write *WOM* code is a coding scheme which allows to write t times to the WOM without decreasing the levels of the cells. The sum-rate of a WOM code is the ratio between the total number of bits written to the memory and the number of cells. It is known that the maximum sum-rate of a t-write WOM code is $\log(t+1)$. This is also an achievable upper bound both by information theory arguments and explicit WOM code constructions. While existing constructions of WOM codes were targeted to increase the sum-rate, we consider here two more figures of merit in evaluating the constructions. The first one is the *complexity* of the encoding and decoding maps of the code. The second one is called the *convergence rate*, and is defined to be the minimum code length $n(\epsilon)$ in order to reach ϵ close to a point in the capacity region. One of our main results in the paper is a specific capacity achieving construction for two-write WOM codes which has polynomial complexity and relatively short block length to be ϵ close to the capacity. Using these two-write WOM codes, we obtain three-write WOM codes that approach sum-rate 1.809 with relatively short block lengths. Finally, we provide another construction of three-write WOM that achieves sum-rate 1.71 by using only 100 cells.**

I. Introduction

Write-once memory (*WOM*) is a storage medium consisting of cells that can only increase their level. WOM codes were first introduced by Rivest and Shamir [23] in 1982 and were motivated by storage medium like punch cards and optical disks. These media are comprised of storage elements, called *cells*, which have a special asymmetric programming attribute. In the binary version, it is only allowed to irreversibly program each cell from level zero to level one. If a cell can accommodate more than two levels, then on each programming operation, it is only possible to increase the cell's level. A WOM code is a coding scheme which allows to store multiple messages in the WOM, while assuring that cells can only increase their levels on each write. One of the famous examples of a WOM code was presented by Rivest and Shamir for storing two bits twice using only three cells [23]. In this work they presented more constructions of WOM codes and several more families of WOM codes were studied later in the 1980's and 1990's; see e.g., [6], [13], [20].

In the last decade, WOM codes have attracted tremendous interest due to their applicability to flash memories [2], [9], [17], [25], [26], [29]–[32]. Flash memory is another example of a WOM where its cells are charged with electrons and thus represent multiple levels [3]. Increasing a cell level is fast and easy; however, in order to decrease its level, its entire containing block of cells has to be erased first. This does not only affect the writing speed of the flash memory but also significantly reduces its lifetime [3]. Therefore, reducing the number of block erasures is crucial in order to improve the lifetime of flash memories. The implementation of WOM codes in flash memories was recently demonstrated in several works, e.g. [18], [21], [34], as well as its benefits in improving the memory lifetime [33].

Assume the WOM, consisting of n binary cells, is required to accommodate t messages. For $1 \leqslant i \leqslant t$, let M_i be the message size on the i-th write. The *rate* of the i-th write is defined to be $\mathcal{R}_i = \frac{\log M_i}{n}$, and the *sum-rate* is $\mathcal{R} = \sum_{i=1}^{t} \mathcal{R}_i$. The capacity region of a t-write WOM is the set of all achievable rate tuples. For the binary case, the capacity region was found in [12], [15], [23]. It was also proved that the maximum achievable sum-rate for a binary WOM code with t writes is $\log(t+1)$. These results were generalized in [12] for non-binary WOM and the maximum sum-rate for WOM with q-ary cells was shown to be $\log\binom{t+q-1}{t}$.

The main goal in designing a WOM code is to achieve, by explicit code constructions, all rate tuples in the capacity region, and in particular high sum-rate. For the two-write case, such constructions were studied in [26] and [31] and for multiple writes in [25]. There are also capacity achieving constructions but for the ϵ-error case, that is, successive writes are not guaranteed in the worst case [2], [9], [17], [29]. In this work, we are only interested in the zero-error case, i.e. each write succeeds in the worst case. There are two more figure of merits which we will investigate in this work when evaluating WOM codes constructions. The first one, which we call the *complexity*, is the complexity of the encoding and decoding maps of the WOM code as a function of code length n. The second one, called the *convergence rate*, is the minimum code length $n(\epsilon)$ in order to be ϵ-close to a point in the capacity region.

Prior to this work, the capacity achieving constructions for two-write WOM cannot achieve these two figures of merit simultaneously. Namely, the one from [31] is beneficial for the convergence rate but suffers from high complexity, while the second one in [26] has opposite attributes and requires an exponential block length in order to be ϵ-close to the capacity. One of our main results in this paper is an explicit construction of two-write WOM codes which provides both low complexity and short block length to be ϵ-close to the capacity. Furthermore, it is possible to derive explicit rate tuples under this construction which improve upon the previously best known sum-rate of 1.4928 in [31]. WOM codes with short block length are attractive due to their efficient encoding and decoding procedures [4].

For three-write WOM codes, the best known sum-rate of an explicit WOM code is 1.61 [31], while the upper bound is 2. The best sum-rate which could be achieved by the construction from [31] is 1.66, and in [26] this result was improved to sum-rate 1.809. Then, another improvement was given in [32]

978-1-5090-4096-4/17/$31.00 ©2017 IEEE

and the best sum-rate was 1.885. As mentioned before, the construction from [25] is a capacity achieving construction for any number of writes, and in particular three writes. However, in order to get close to the capacity the block length has to be extremely large and significantly larger than the one from [26], [31], [32]. Using our two-write WOM codes, we modify the construction in [26] to obtain three-write WOM codes that approach sum-rate 1.809 faster than the former. In addition, we provide another construction of three-write WOM that achieves sum-rate 1.71 by using only 100 cells.

The rest of the paper is organized as follows. In Section II, we review the definitions of WOM codes and define the figure of merits of complexity and convergence rate that we be used in evaluating WOM code constructions. In Section III, we review state of the art results for the constructions of two-write WOM codes and present our construction with polynomial complexity and fast convergence rate. In Section IV, we present our constructions of explicit three-write WOM codes. These constructions will improve upon state of the art sum-rate results of explicit known WOM codes. Finally, Section V concludes the paper.

II. Definitions and Basic Properties

In this work, we assume that the memory consists of n binary cells, where initially all of them are in the zero state. On each write it is only possible to increase the level of each cell to level one. A vector $\boldsymbol{x} = (x_1, \ldots, x_n) \in \{0,1\}^n$ will be called a *cell-state vector*. For two cell-state vectors $\boldsymbol{x}$ and $\boldsymbol{y}$, we say that $\boldsymbol{x} \leqslant \boldsymbol{y}$ if $x_i \leqslant y_i$ for all $1 \leqslant i \leqslant n$. For a positive integer n, we use the notation $[n]$ to define the set of integers $\{1, \ldots, n\}$. If x represents a bit value then its complement is $\overline{x} = 1 - x$, and for a binary vector $\boldsymbol{x} = (x_1, \ldots, x_n)$, $\overline{\boldsymbol{x}} = (\overline{x}_1, \ldots, \overline{x}_n)$. For any map $f : A \to B$, $Im(f)$ is the image of the map f. The binary entropy function is defined for every probability $0 < p < 1$ as $h(p) = -p\log(p) - (1-p)\log(1-p)$.

We follow the formal definition of WOM codes from [32].

Definition 1. *An $[n, t; M_1, \ldots, M_t]$ t-write WOM code is a coding scheme comprising of n binary cells and is defined by t pairs of encoding and decoding maps $(\mathcal{E}_i, \mathcal{D}_i)$, for $1 \leqslant i \leqslant t$. The encoding map $\mathcal{E}_i$ is defined by*

$$\mathcal{E}_i : [M_i] \times Im(\mathcal{E}_{i-1}) \to \{0,1\}^n,$$

where, by definition, $Im(\mathcal{E}_0) = \{(0, \ldots, 0)\}$, such that $\mathcal{E}_i(m, \boldsymbol{c}) \geqslant \boldsymbol{c}$ for all $(m, \boldsymbol{c}) \in [M_i] \times Im(\mathcal{E}_{i-1})$. Similarly, the decoding map $\mathcal{D}_i$ is defined by

$$\mathcal{D}_i : Im(\mathcal{E}_i) \to [M_i],$$

such that for all $(m, \boldsymbol{c}) \in [M_i] \times Im(\mathcal{E}_{i-1})$, $\mathcal{D}_i(\mathcal{E}_i(m, \boldsymbol{c})) = m$. The rate on the i-th write is defined by $\mathcal{R}_i = \frac{\log M_i}{n}$, and the sum-rate is $\mathcal{R}_{sum} = \sum_{i=1}^{t} \mathcal{R}_i$.

In [12], [15], the capacity region of a binary t-write WOM was found to be

$$C_t = \Big\{(\mathcal{R}_1, \ldots, \mathcal{R}_t) | \mathcal{R}_1 \leqslant h(p_1), \mathcal{R}_2 \leqslant (1-p_1)h(p_2), \ldots, \mathcal{R}_{t-1} \leqslant \Big(\prod_{i=1}^{t-2}(1-p_i)\Big)h(p_{t-1}), \mathcal{R}_t \leqslant \prod_{i=1}^{t-1}(1-p_i), \text{ where } 0 \leqslant p_1, \ldots, p_{t-1} \leqslant 1/2\Big\},$$

and $\log(t+1)$ was proved to be the maximum sum-rate. Even though it is known that all rate tuples in the capacity region are achievable, the problem of finding efficient code construction remains a challenge. According to [31], we assume that the write number on each write is known since this side information does not affect the achievability of rate tuples in the capacity region

We will evaluate the constructions efficiency according to the following two figures of merit, which we next define for the constructions of WOM codes.

1) *Complexity*: the construction complexity is defined to be the complexity of the encoding and decoding maps as a function of code length n.
2) *Convergence rate*: the minimum code length $n(\epsilon)$ in order to be ϵ-close to a rate tuple $(\mathcal{R}_1, \ldots, \mathcal{R}_t)$ in the capacity region.

More rigorously, the second figure of merit states that the convergence rate of a WOM codes construction to a rate tuple $(\mathcal{R}_1, \ldots, \mathcal{R}_t) \in C_t$ is $n(\epsilon)$ if there exists a WOM code by the construction of length $n(\epsilon)$ with rate tuple at least $(\mathcal{R}_1 - \epsilon, \ldots, \mathcal{R}_t - \epsilon)$. Since we will be mostly interested to determine whether $n(\epsilon)$ is polynomial or exponential with $1/\epsilon$, we say that a construction approaches a rate tuple $(\mathcal{R}_1, \ldots, \mathcal{R}_t) \in C_t$ with polynomial, exponential rate if $n(\epsilon)$ is polynomial, exponential with $1/\epsilon$, respectively. Similarly we that say a construction approaches a sum-rate $\mathcal{R}_{\text{sum}}$ with polynomial or exponential rate, and for capacity achieving constructions with polynomial or exponential rate.

III. Two-Write WOM Codes Construction

In this section we present an explicit construction of two-write WOM codes. As opposed to existing constructions, we will prove that this construction will have both polynomial complexity and polynomial convergence rate.

A. Background and State of the Art Results

The capacity region of a two-write WOM is given by

$$C_2 = \big\{(\mathcal{R}_1, \mathcal{R}_2) | \mathcal{R}_1 \leqslant h(p), \mathcal{R}_2 \leqslant (1-p), 0 \leqslant p \leqslant 1/2\big\},$$

and the maximum sum-rate is $\log 3 \approx 1.58$. On the other hand, the best sum-rate reported in the literature is 1.4928 [31]. There are three explicit constructions which achieve the capacity C_2 of two-write WOM. The first one to accomplish this task was presented in [31]. Shortly after, in [26] Shpilka presented another capacity achieving construction for two writes and a general construction for t writes in [25]. There are several other works which achieve the ϵ-error capacity of two-write WOM. Here, ϵ-error implies that the second write does not succeed in the worst case but only with high probability. These constructions are based on polar codes [2], LDPC codes [9], [17], or random matrices [29].

All the aforementioned constructions use a very similar principle. For a given probability $0 < p < 1/2$, if the WOM has n cells, then on the first write at most pn cells are programmed (but not necessarily all such patterns). Thus, it is possible to store roughly $nh(p)$ bits and the rate approaches $h(p)$. The challenge on the second write is to store roughly $(1-p)n$ more bits on the remaining $(1-p)n$ cells.

The construction in [31] accomplishes this task by restricting the cell-state vectors that can be programmed on the first write, such that any pattern of $n(1-p)$ bits can be stored on the second write. This approach significantly simplifies the encoding and decoding on the second write however it incurs an extremely high complexity on the first write since not all vectors of weight at most pn can be programmed. While the convergence rate of this construction is polynomial its complexity can be exponential since it may require a lookup table for the first write encoding and decoding.

On the other hand, in [26] (almost) any pattern of at most pn cells can be programmed and the second write uses a set of "average" MDS codes which are derived from a *Wozencraft ensemble* [16], [19]. This collection of codes guarantees the success of encoding roughly $(1-p)n$ bits on the second write. The encoding and decoding complexities of this construction are polynomial with the code length n. However in order to achieve high sum-rate, the block length has to be extremely large, and in particular the convergence rate of this construction is exponential.

The constructions from [31] and [26] introduce a tradeoff between the complexity and convergence rate. While the first one suffers from high complexity but achieves polynomial convergence rate, the second one has opposite attributes, i.e., low complexity but exponential convergence rate. However, we will show that there is no such tradeoff by presenting a construction which accomplishes these two goals simultaneously.

B. The Construction

We are now ready to present our two-write WOM codes construction. This construction is motivated by a recent one [5], which is based on the notion of *spreads* in projective geometry. A specific example for this construction was first developed by Dumer for codes with stuck-at cells [8]. This structure is defined formally as follows.

Definition 2. *A collection $V_1, V_2, \ldots, V_M$ of τ-dimensional subspaces of $\mathbb{F}_2^n$ is said to be a* τ-partial spread *of $\mathbb{F}_2^n$ of size M if*

$$V_i \cap V_j = \{\mathbf{0}\} \text{ for all } i \neq j$$

In addition, if $\bigcup_{i=1}^M V_i = \mathbb{F}_2^n$, then the collection is said to be a τ-spread *of $\mathbb{F}_2^n$.*

Spreads were widely studied in the literature and it is well known that τ-spreads exist if and only if τ divides n; see e.g. [1], [10], [24], [27]. In this case it also holds that

$$M = \frac{2^n - 1}{2^\tau - 1} > 2^{n-\tau}.$$

The case where n is not a multiple of τ was studied in [11] and it was shown that a τ-partial spread of $\mathbb{F}_2^n$ of size $2^{n-\tau}$ exists, where $\tau \leqslant n/2$.

The construction of two-write WOM codes is carried as follows.

Theorem 3. *For all τ and n such that $\tau + 1 \leqslant n/2$, there exists an $[n, 2; M_1, M_2]$ two-write WOM code, where*

$$M_1 = \sum_{i=0}^{\tau} \binom{n}{i}, M_2 = 2^{n-(\tau+1)}.$$

Moreover the construction complexity is polynomial $O(n^3)$ and it is capacity achieving with polynomial rate.

Proof: We shall prove this theorem by explicitly presenting its encoding and decoding on each write. The first write is simple. We encode one of M_1 messages by simply programming one of the vectors of weight at most τ. The encoding and decoding complexities for this step can be implemented for example by the enumerative coding scheme by Cover [7].

For the second write we use a $(\tau+1)$-partial spread, given by the collection of M_2 subspaces $V_1, \ldots, V_{M_2}$. Let $m \in [M_2]$ be the message to be written on the second write, and let S be the set of indices of the programmed cells on the first write, so $|S| \leqslant \tau$ but for the simplicity of the proof we assume that $|S| = \tau$. The encoding of this write is performed by programming a vector $\boldsymbol{v}$ such that for all $i \in S$, $v_i = 1$.

Assume that $\boldsymbol{v}_1, \boldsymbol{v}_2, \ldots, \boldsymbol{v}_{\tau+1}$ is a basis for V_m, and let $\boldsymbol{v}_1', \boldsymbol{v}_2', \ldots, \boldsymbol{v}_{\tau+1}'$ be their projections on the set S. Since for $i \in [\tau+1]$, $\boldsymbol{v}_i' \in \mathbb{F}_2^\tau$, these $\tau+1$ vectors belong to a τ-dimensional vector space and therefore there exist non-trivial coefficients $a_1, \ldots, a_{\tau+1}$ for which

$$\sum_{i=1}^{\tau+1} a_i \boldsymbol{v}_i' = \mathbf{0}.$$

Finally, let $\boldsymbol{x} = \sum_{i=1}^{\tau+1} a_i \boldsymbol{v}_i$, and the programmed cell-state vector is $\boldsymbol{c} = \overline{\boldsymbol{x}}$. Note that since for all $i \in S$, $x_i = 0$, we get that $c_i = 1$, as required. Given a codeword $\boldsymbol{c}$, we are able to determine $\boldsymbol{x}$. Since the subspaces in the partial spread intersect trivially, we are able to identify the subspace that the nonzero vector $\boldsymbol{x}$ belongs to and hence decode the value of m. This completes the decoding.

Complexity: The complexity of the encoding and decoding maps is $O(n^3)$. For more details, we refer the reader to [5].

Convergence rate: We now turn to prove that the convergence rate of the construction is polynomial. For all $\tau + 1 \leqslant n/2$, the rates of the construction are given by

$$\mathcal{R}_1 = \frac{\log\left(\sum_{i=0}^{\tau} \binom{n}{i}\right)}{n}, \mathcal{R}_2 = \frac{n - (\tau+1)}{n},$$

and thus the sum-rate is

$$\mathcal{R} = \frac{\log\left(\sum_{i=0}^{\tau} \binom{n}{i}\right) + n - (\tau+1)}{n}.$$

For each $0 < p < 1/2$, let us choose $\tau = \lceil np \rceil$. According to [22, Lemma 4.8] we use the inequality

$$\sum_{i=0}^{\tau} \binom{n}{i} \geqslant \frac{1}{n+1} 2^{nh\left(\frac{\tau}{n}\right)},$$

and conclude that

$$\begin{aligned}
\mathcal{R} &\geqslant \frac{\log\left(\frac{1}{n+1} 2^{nh\left(\frac{\tau}{n}\right)}\right) + n - (\tau+1)}{n} \\
&= h\left(\frac{\tau}{n}\right) + 1 - \frac{\tau+1}{n} - \frac{\log(n+1)}{n} \\
&\geqslant h(p) + 1 - p - \frac{2 + \log(n+1)}{n}.
\end{aligned}$$

Therefore if $n = (1/\epsilon)^2$ then $\mathcal{R} \geqslant h(p) + 1 - p - \epsilon$ and in particular, $\mathcal{R}_1 \geqslant h(p) - \epsilon$ and $\mathcal{R}_2 \geqslant 1 - p - \epsilon$. ■

The codes we derived in Theorem 3 are not only capacity achieving. The construction also provides WOM codes with explicit sum-rates. For example, if we choose $n = 168, \tau = 55$ we will get sum-rate $\mathcal{R} = 1.561$.

IV. Three-Write WOM Codes Construction

In this section, we present two constructions of three-write WOM codes and we focus on obtaining codes with high sum-rates. Suppose a family of three-write WOM codes approaches sum-rate $\mathcal{R}_{\text{sum}}$. As before, we consider the minimum code length $n(\epsilon)$ in order to be ϵ-close to $\mathcal{R}_{\text{sum}}$ and determine whether $n(\epsilon)$ is polynomial or exponential with $1/\epsilon$.

The first construction is a simple modification of a three-write WOM code in [26] and has sum-rates approaching 1.809. The WOM codes from this construction have polynomial complexity and approach 1.809 at polynomial rate. The second construction however is non-explicit but provides three-write WOM codes with high sum-rates for moderate values of n. In fact, using the second construction, we obtain three-write WOM codes with sum-rate 1.71 using only 100 cells. The previously best known sum-rate of explicit WOM codes is 1.61 in [31].

A. First Construction

Our first construction is based on a three-write WOM code in [26]. On the first two writes, Shpilka used a modification of the Rivest-Shamir two-write WOM code [23] that guarantees that at most a certain number of cells are programmed after the second write. On the third write, Shpilka then applies his two-write WOM code (derived from a Wozencraft ensemble).

Theorem 4. *[26] Let m and ℓ be integers such that $\ell \leqslant 4m$. Set $n = 12m + 5$ and $\tau = \lceil 8m - 5\ell/4 \rceil$. There exists an $[n, 3; M_1, M_2, M_3]$ three-write WOM, where*

$$M_1 = \sum_{i=\ell}^{4m} \binom{4m}{i} 3^{4m-i},\ M_2 = 4^{4m},\ M_3 = 2^{12m-\tau}.$$

After the second write, at most τ cells are programmed. Furthermore, if we set $\ell = \lfloor 1.768m \rfloor$, these codes approach sum-rate 1.809.

Since these codes by Shplika use a Wozencraft ensemble, the sum-rate of the three-write WOM codes approaches 1.809 with exponential rate. A simple modification to this scheme is to apply the two-write WOM code from Theorem 3. Since at most τ cells are programmed after the second write and if we set $\tau + 1 \leqslant n/2$, or $\lceil 8m - 5\ell/4 \rceil \leqslant 6m + 1$, we are able to apply Theorem 3 to encode $2^{n-(\tau+1)}$ messages. Since the two-write WOM code has polynomial convergence rate, the sum-rate of the modified three-write WOM code approaches 1.809 with polynomial rate. We summarize this discussion in the following theorem.

Theorem 5. *Let m and ℓ be integers such that $\ell \leqslant 4m$ and $\lceil 8m - 5\ell/4 \rceil \leqslant 6m + 1$. Set $n = 12m + 5$ and $\tau = \lceil 8m - 5\ell/4 \rceil$. There exists an $[n, 3; M_1, M_2, M_3]$ three-write WOM, where*

$$M_1 = \sum_{i=\ell}^{4m} \binom{4m}{i} 3^{4m-i},\ M_2 = 4^{4m},\ M_3 = 2^{12m-(\tau+1)}.$$

Moreover, the construction complexity is polynomial $O(n^3)$ and if we set $\ell = \lfloor 1.768m \rfloor$, these codes approach sum-rate 1.809 with polynomial rate.

B. Second Construction

Our second construction for three-write WOM codes is non-explicit. Nevertheless, the construction yields the best known sum-rate for block lengths up to 100. For that we first show a modification for the two-write construction from Theorem 3.

Observe that in Theorem 3, we require $\tau + 1 \leqslant n/2$. In the next theorem, we propose a two-write WOM code, where τ is possibly greater than $n/2$. While we do not obtain new points in the capacity region, these codes will be useful in the constructions of three-write WOM codes in this section. These codes are similar to the codes constructed in [5] and hence, the proof is omitted.

Theorem 6. *Let n, τ, r, k, and d be integers such that*

1) $\tau + r \leqslant (n + k)/2$,
2) *there exists an $[n, k, d]$ linear code, and*
3) *there does* not *exist an $[n - \tau, r, d]$ linear code.*

Then there exists an $[n, 2; M_1, M_2]$ two-write WOM code, where

$$M_1 = \sum_{i=0}^{\tau} \binom{n}{i}, M_2 = 2^{n-(\tau+r)}.$$

Moreover, on the first write, at most τ cells are programmed.

Example 1. Let $n = 100$, $\tau = 76$, $r = 3$, $k = 58$, and $d = 14$, so it holds that $\tau + r \leqslant (n + k)/2$. From [14], we know that an $[100, 58, 14]$ linear code exists, while an $[24, 3, 14]$ linear code does not exist. Therefore, we have a $\left[100, 2; \sum_{i=0}^{76} \binom{100}{i}, 2^{21}\right]$ two-write WOM code. Here, the rates are given by $\mathcal{R}_1 \approx 1$ and $\mathcal{R}_2 = 0.21$. □

We are now ready to present the construction for three-write WOM codes. As with most three-write WOM constructions, the key idea is to minimize the number of programmed cells after the second write. To do so, instead of using all subspaces in the spread, we only choose certain good ones to encode as the messages for the second write.

Theorem 7. *Let τ_1 and τ_2 be integers such that*

1) $\tau_1 + 1 \leqslant n/2$,
2) $\sum_{i=0}^{n-\tau_2-1} \binom{n}{i} < 2^{n-(\tau_1+1)}$, *and*
3) *Theorem 6 yields an $[n, 2; \sum_{i=0}^{\tau_2} \binom{n}{i}, M_3]$ two-write WOM.*

Then, there exists an $[n, 3; M_1, M_2, M_3]$ three-write WOM, where

$$M_1 = \sum_{i=0}^{\tau_1} \binom{n}{i},\ M_2 = 2^{n-(\tau_1+1)} - \sum_{i=0}^{n-\tau_2-1} \binom{n}{i}.$$

Proof: Consider a $(\tau_1 + 1)$-partial spread, given by the collection of $M = 2^{n-(\tau_1+1)}$ subspaces $V_1, \ldots, V_M$. We say that a subspace is *bad* if it contains a nonzero vector of weight at most $n - \tau_2 - 1$. Since there are at most $\sum_{i=0}^{n-\tau_2-1} \binom{n}{i}$ words of weight at most $n - \tau_2 - 1$, we have at most $\sum_{i=0}^{n-\tau_2-1} \binom{n}{i}$ bad subspaces. We expurgate these bad subspaces and use the remaining M_2 subspaces to encode the second write.

Note that any nonzero vector $\boldsymbol{x}$ in these subspaces have weight at least $n - \tau_2$. Since we program the vector $\bar{\boldsymbol{x}}$, there are at most τ_2 programmed cells after the second write.

For the third write, we use the $\left[n,2;\sum_{i=0}^{\tau_2}\binom{n}{i},M_3\right]$ two-write WOM from Theorem 6. This completes the proof. ■

The next example demonstrates this construction.

Example 2. Let $n=100$, $\tau_1=23$, and $\tau_2=76$. Then

$$M_1=\sum_{i=0}^{23}\binom{100}{i}\approx 2^{74.9}, \text{ and}$$

$$M_2=2^{76}-\sum_{i=0}^{23}\binom{100}{i}\approx 2^{75.1}.$$

From Example 1, we have a $\left[100,2;\sum_{i=0}^{76}\binom{100}{i},2^{21}\right]$ two-write WOM code, and we set

$$M_3=2^{21}.$$

Therefore, the sum-rate is approximately $0.749+0.751+0.21=1.71$.

In contrast, we consider the three-write WOM codes from Theorem 5. For $1\leqslant m\leqslant 8$, or $n\leqslant 101$, the best sum-rate achievable is 1.68. Therefore, Theorem 7 yields the best known sum-rate for lengths up to 100. □

V. Conclusions and Open Problems

In this paper we studied efficient constructions of WOM codes. In the two-write case, we provided a capacity achieving construction which has polynomial complexity and polynomial convergence rate. For three writes, we gave WOM codes which improved upon the best known sum-rate of explicit WOM codes.

While the results in the paper provide a significant contribution in the area of WOM codes, there are still several interesting problems which are left open. Namely, finding better explicit constructions for the three-write case and capacity achieving construction with polynomial complexity and polynomial convergence rate. These problems are even more challenging and stimulating for more than three writes, where much less is known.

VI. Acknowledgement

The research of Y. M. Chee was supported in part by the Singapore Ministry of Education under Research Grant MOE2015-T2-2-086. The research of H. M. Kiah was supported in part by the Singapore Ministry of Education under Research Grants MOE2015-T2-2-086 and MOE2016-T1-001-156. The research of Alexander Vardy was supported in part by the National Science Foundation under Grant CCF-1405119. The research of Eitan Yaakobi was supported in part by the Israel Science Foundation (ISF) Grant 1624/14.

References

[1] T. Bu, "Partitions of a vector space," Discrete Math., vol. 31, pp. 79–83, Jan. 1980.
[2] D. Burshtein and O. Strugatski, "Polar write once memory codes," *IEEE Trans. Inform. Theory*, vol. 59, no. 8, pp. 5088–5101, Aug. 2013.
[3] P. Cappelletti, C. Golla, P. Olivo, and E. Zanoni, *Flash Memories*, Boston: Kluwer Academic, 1999.
[4] Y. Cassuto and E. Yaakobi, "Short q-ary fixed-rate WOM codes for guaranteed re-writes and with hot/cold write differentiation," *IEEE Trans. Inform. Theory*, vol. 60, no. 7, pp. 3942–3958, Jul. 2014.
[5] Y.M. Chee, T. Etzion, H.M. Kiah, and A. Vardy, "Cooling codes: Thermal-management coding for high-performance interconnects," *preprint*.
[6] G.D. Cohen, P. Godlewski, and F. Merkx, "Linear binary code for write-once memories," *IEEE Trans. Inform. Theory*, vol. 32, no. 5, pp. 697–700, Oct. 1986.
[7] T.M. Cover, "Enumerative source encoding," *IEEE Trans. Inform. Theory*, vol. 19, no. 1, pp. 73–77, Jan. 1973.
[8] I. I. Dumer, "Asymptotically optimal codes correcting memory defects of fixed multiplicity," *Probl. Peredachi Inform.*, vol. 25, no. 4, pp. 3-10, Oct. 1989.
[9] E. En Gad, W. Huang, Y. Li, and J. Bruck, "Rewriting flash memory by message passing," *Proc. IEEE Int'l Symp. Information Theory*, Hong Kong, pp. 646–650, Jun. 2015.
[10] T. Etzion, "Perfect byte-correcting codes," *IEEE Trans. Inform. Theory*, vol. 44, no. 7, pp. 3140–3146, Nov. 1998.
[11] T. Etzion and A. Vardy, "Error-correcting codes in projective space," *IEEE Trans. Inform. Theory*, vol. 57, no. 2, pp. 1165–1173, Feb. 2011.
[12] F. Fu and A.J. Han Vinck, "On the capacity of generalized write-once memory with state transitions described by an arbitrary directed acyclic graph," *IEEE Trans. Inform. Theory*, vol. 45, no. 1, pp. 308–313, Jan. 1999.
[13] P. Godlewski, "WOM-codes construits à partir des codes de Hamming," *Discrete Math.*, vol. 65, no. 3, pp. 237–243, Jul. 1987.
[14] M. Grassl, "Bounds on the minimum distance of linear codes and quantum codes." Online available at http://www.codetables.de. Accessed on 2016-12-27.
[15] C. Heegard, "On the capacity of permanent memory," *IEEE Trans. Inform. Theory*, vol. 31, no. 1, pp. 34–42, Jan. 1985.
[16] J. Justesen, "A class of constructive asymptotically good algebraic codes, *IEEE Trans. on Inform. Theory*, vol. 18, no. 5, pp. 652–656, Sep. 1972
[17] S. Kumar, A. Vem, K. Narayanan, and H.D. Pfister, "Spatially-coupled codes for write-once memories," *Proc. 53rd Annual Allerton Conference on Communication, Control, and Computing*, pp. 125–131, Sep. 2015.
[18] F. Margaglia, G. Yadgar, E. Yaakobi, Y. Li, A. Schuster, and A. Brinkmann, "The devil is in the details: Implementing flash page reuse with WOM codes," *Usenix FAST*, Santa Clara, CA, Feb. 2016.
[19] J.L. Massey, Threshold decoding Res. Lab. Electron., Massachusetts Inst. Technol., Cambridge, MA, USA, Tech. Rep. 410, 1963.
[20] F. Merkx, "Womcodes constructed with projective geometries," *Traitement du signal*, vol. 1, no. 2-2, pp. 227–231, 1984.
[21] S. Odeh and Y. Cassuto, "NAND flash architectures reducing write amplification through multiwrite codes," *30th Symp. on Mass Storage Systems and Technologies (MSST)*, pp. 110, May 2014.
[22] R.M. Roth, *Introduction to Coding Theory*, Cambridge, U.K.: Cambridge Univ. Press, 2005.
[23] R.L. Rivest and A. Shamir, "How to reuse a write-once memory," *Inform. and Contr.*, vol. 55, no. 1–3, pp. 1–19, Dec. 1982.
[24] M. Schwartz and T. Etzion, "Codes and anticodes in the Grassmann graph," *J. Combin. Theory, ser. A*, vol. 97, pp. 27–42, 2002.
[25] A. Shpilka, "Capacity achieving multiwrite WOM codes," *IEEE Trans. Inform. Theory*, vol. 60, no. 3, pp. 1481–1487, Mar. 2014.
[26] A. Shpilka, "New constructions of WOM codes using the Wozencraft ensemble," *IEEE Trans. on Inform. Theory*, vol. 59, no. 7, pp. 4520–4529, Jul. 2013.
[27] S. Thomas, "Designs over finite fields," *Geometriae Dedicata*, vol. 21, pp. 237–242, 1987.
[28] J.K. Wolf, A.D. Wyner, J. Ziv, and J. Korner, "Coding for a write-once memory," *AT&T Bell Labs. Tech. J.*, vol. 63, no. 6, pp. 1089–1112, 1984.
[29] Y. Wu, "Low complexity codes for writing write-once memory twice," *Proc. IEEE Int. Symp. Inform. Theory*, Austin, Texas, Jun. 2010.
[30] Y. Wu and A. Jiang, "Position modulation code for rewriting write-once memories," accepted by *IEEE Trans. Inform. Theory*, vol. 57, no. 6, pp. 3692–3697, Jun. 2011.
[31] E. Yaakobi, S. Kayser, P.H. Siegel, A. Vardy, and J.K. Wolf, "Codes for write-once memories," *IEEE Trans. Inform. Theory*, vol. 58, no. 9, pp. 5985–5999, Sep. 2012.
[32] E. Yaakobi and A. Shpilka, "High sum-rate three-write and non-binary WOM codes," *IEEE Trans. Inform. Theory*, vol. 60, no. 11, pp. 7006–7015, Nov. 2014.
[33] E. Yaakobi, A. Yucovich, G. Maor, G. Yadgar, "When do WOM codes improve the erasure factor in flash memories?," *Proc. IEEE Int'l Symp. Information Theory*, Hong Kong, pp. 2091–2095, Jun. 2015.
[34] G. Yadgar, E. Yaakobi, A. Schuster, "Write once, get 50% free: Saving SSD erase costs using WOM codes," *Usenix FAST*, Santa Clara, CA, Feb. 2015.