Trading Partially Stuck Cells with Errors

Haider Al Kim, Sven Puchinger, Ludo Tolhuizen, Antonia Wachter-Zeh
haider.alkim@tum.de, sven.puchinger@tum.de, ludo.tolhuizen@philips.com, antonia.wachter-zeh@tum.de


table

Introduction

The dominance of non-volatile memories and PCMs (phase change memories) as memory solutions for variety of applications have become obvious due to their advantages as permanent storage devices [1].

Problem Description: PCMs may face failures in changing their states, in turn, its cells hold only one phase and they become stuck (defective). On the other hand, random errors may occur on these defective memories.

Solution: A mechanism called masking is used to determine a word whose entries coincide with writable levels at the (partially) stuck cells.

Reliable cells store any value.
The value that can store ∈ Z/4Z.

Cell Levels

<table>
<thead>
<tr>
<th>Level-0 (0)</th>
<th>Level-1 (1)</th>
<th>Level-2 (2)</th>
<th>Level-3 (3)</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>3</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>4</td>
</tr>
</tbody>
</table>

Previous works: In [2], the author considered the problem of masking fully stuck cells together with error correction. The error-free case with partially stuck cells has been considered in [3] in which improvements on the redundancy necessary for masking compared to [2] are achieved.

Our Contribution: In our code constructions in [4], we consider the problem of combined error correction and masking of partially stuck cells, and we reduce the redundancy necessary for masking, similar to the results in [3] and even reduce further compared to [3, Construction 5].

Focus: We propose techniques of using part of the error-correcting capability to mask more partially stuck cells. This observation was also made in [1,Theorem 1].

Definitions

1

Let

(Σ, q)-PSMC: For

∈ Σ, n ∈ Σ and size

is a coding scheme consisting of a message set

of size

, an encoder

and a decoder

. If

= (n (0.1) ) [w (s) ≤ s], we say

q-ary

PSMC.

Proposition 1 [4, Proposition 3]

If there is an

, then for any

j ≤ t ≤ r, there is an

PSMC.

Theorem 1 (Partial Masking PSMC) [4, Theorem 6]

Let

and assume that there exists an

PSMC. For any

, there exists an

PSMC

where

Proof of Theorem 1

Let the encoder

and the decoder

for

be Algorithm 1 and Algorithm 2, respectively. By definition, c′ ≥ c. Moreover, if

, then

. As a result, d(c′) ≤ J.

In Algorithm 2, the decoder

of

is directly used for decoding

. As

, surely

. Moreover, we can write

. As shown above, wt(c′ − e) ≤ j, and so

. As a consequence, τ(y) ≡ w.

Algorithm 2 - Decoder

Input: Retrieved

where

and

Output: Message vector

Lemma 1 [4, Lemma 1]

Given an

code as defined in [4, Construction 3], then for any

such that

, there is a

PSMC of length

and size

.

Lemma 2 [4, Lemma 2]

Given an

code containing a word of weight

, let

and let

. There is a

PSMC of length

and size

.

Lemma 3 (Generalization of Lemma 2) [4, Lemma 3]

Comparisons Gilbert–Varshamov-type Bounds

For [4, Construction 3], we have derived Gilbert–Varshamov-type bound in [4, Theorem 11]. Then, we have compared the direct application of [4, Theorem 11] with the exchange of a one error correction ability with a single masking capability of a partially stuck cell by Theorem 1 that further compared to Lemma 1 and Lemma 2 as shown in Figure 1.

Figure 1. The achievable rates

for different

and

in [4, Theorem 11], Theorem 1, Lemma 1 and Lemma 2.

Conclusion

Treating some of the partially stuck cells as erroneous cells can decrease the required redundancy for some parameters using Theorem 6, Lemma 1 and Lemma 2 improve upon Theorem 6 (i.e. see the perfectly matched dotted-orange and dashed-red plots for

= 19).

References