Encoding Algorithms for Two-Dimensional Constraints for Patterned Media

Hiroshi Kamabe
Department of Information Science, Gifu University,
1-1, Yanagido, Gifu, 501-1193 Japan.
Email: Hiroshi.Kamabe@kmb.info.gifu-u.ac.jp

***Abstract*— We give a simple model for interferences occurring in 2-D patterned media for magnetic recording. We investigate some constraints induced from the model. We also give an encoding algorithm for the constraints which works well with minor exceptions.**

I. INTRODUCTION

In digital magnetic recording and digital data transmission systems, we need constrained codes for channels with input constraints because media or channels are assumed to have input constraints. We can also use constrained codes to increase the error correcting capability of one dimensional recording systems.

To achieve high recording density, 2-D recording devices have been investigated. Therefore many authors have studied encoding algorithms and capacities of two-dimensional input constraints[1], [2], [3], [4], [5].

Many researchers investigated 2-D constraints as input constraints for optical storage media. Recently 2-D patterned media have been received much attention as a possible candidate for next generation perpendicular recording media. Although these two kinds of media have common properties as recording channels for digital data, we should model them separately to obtain better performance.

In this draft we explain a simple model for 2-D perpendicular magnetic recording media and give bounds of capacities of constraints induced from the model. We give an encoding rule and show experimental results.

II. MODEL OF PHYSICAL CONSTRAINT

A. Assumptions

We consider a recording medium which consists of a lot of dots in the square lattice. The magnetic field created by a dot is illustrated in Fi.g 1. These dots are isolated each other, that is, there is some space between adjacent dots. Therefore, there is no jitter noise caused by the transition area of magnetization. Each dot is magnetized perpendicularly.

We show the inter-symbol-interference(actually, inter-dot-interference) caused by the magnetic field emitted from two adjacent dots in Fig. 2. Two dots in the figure are magnetized the same perpendicular direction. If the space between these dots is not enough or the magnetic field is very strong, the magnetic field of one of them has a negative effect on another dot. The effect decreases the level of read back signals and might flip the direction of magnetization in a very long term. Hokkyo et al. reported this interference degrades the quality of the read back signal and the reliability of a particular kind of 2-dimensional media [6].

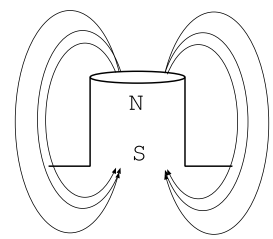

Fig. 1. Magnetic field of dot

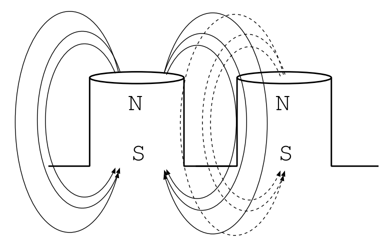

Fig. 2. Interference of Magnetic field

We represent the effect by a single dot to neighboring dots as follows: $\widetilde{h}(i,j,s)$ is the degree of an effect by the magnetic field from a dot having the magnetization direction s at (a,b) position to another dot at $(a+i,b+j)$ position. This definition implies our implicit assumption that these effects are uniform, that is, position independent. Coefficient $\widetilde{h}(0,0,s)$ means the level of magnetization of a single dot. Although there is a reverse magnetic field effect by itself, we assume that the effect is already included in $\widetilde{h}(0,0,s)$.

There are two magnetization directions, up and down. We represent these directions by symbols '$+$' and '$-$', respectively. Since we can expect that the magnetization process for recording is symmetric on the direction of electric current which drives a recording head, we assume that $\widetilde{h}(0,0,\text{`}+\text{'}) = -\widetilde{h}(0,0,\text{`}-\text{'})$.

Although $\widetilde{h}(i,j,s)$ is a real valued number in actual systems, we can assume that $\widetilde{h}(i,j,s)$ is an integer-valued number after appropriate quantizations and scaling for our purpose.

Since the magnetic field is symmetric on the square lattice, we can have the following assumption:

$$\widetilde{h}(i,j,s) = \widetilde{h}(-i,j,s) = \widetilde{h}(i,-j,s) = \widetilde{h}(-i,-j,s).$$

We will quantize every retrieved signal and the magnetic field tends to 0 as the distance between two dots gets large. Hence we assume that there is a constant L such that $\widetilde{h}(i,j,s) = 0$ for $i \geq L$ or $j \geq L$.

We introduce a function h with

$$h(i,j) = \widetilde{h}(i,j,\text{`}+\text{'}). \tag{1}$$

Then by interpreting symbols '$+$' and '$-$' as numbers $+1$ and -1, respectively, we can write

$$\widetilde{h}(i,j,s) = sh(i,j).$$

We can assume $h(0,0) > 0$ without loss of generality.

B. Constraints

Let d be a positive constant. We describe a constraint on the arrangement of bits (directions of magnetization of dots) on the square lattice as follows:

Constraint 1: For any position (a,b) we must have

$$\left| \sum_{i=-L}^{L} \sum_{j=-L}^{L} y(a+i, b+j) h(i,j) \right| \geq d \tag{2}$$

where $y(a',b')$ is a channel bit (or the magnetization direction) at (a',b') position.
This constraint requires that the sum of magnetization level and anti-magnetization effects at every position from neighboring dots should be bounded from above by the constant d.

Although this description of our constraint reflects physical constraints we have, it is not suitable for analyzing and counting allowable patterns. Hence we rewrite the constraint as follows.

Let $\mathcal{S}_h^d$ be a family of all subsets $S \subset \{(i,j) : -L \leq i,j \leq L\} \backslash \{(0,0)\}$ such that

$$\left| \sum_{(i,j)\in S} h(i,j) \right| > h(0,0) - d.$$

For each $S \in \mathcal{S}$ we define two patterns T_S^+ and T_S^- of size $(2L+1) \times (2L+1)$ as follows:

$$T_S^+(i,j) = \begin{cases} + & \text{if } (i,j) = (0,0); \\ + & \text{if } (i,j) \in S; \\ - & \text{otherwise,} \end{cases} \tag{3}$$

$$T_S^-(i,j) = -T_S^+(i,j), \tag{4}$$

where we index the center positions of T_S^+ and T_S^- as $(0,0)$. Let $\mathcal{T}_h^d$ be a set defined by

$$\mathcal{T}_h^d = \{T_S^+ : S \in \mathcal{S}\} \cup \{T_S^- : S \in \mathcal{S}\}.$$

Then our constraint is described as follows.

Constraint 2: No pattern in $\mathcal{T}_h^d$ appears in the square lattice.

In this draft, we index positions in the square lattice as follows

$$\begin{matrix} (0,0) & (0,1) & (0,2) & (0,3) & \cdots \\ (1,0) & (1,1) & (1,2) & (1,3) & \cdots \\ (2,0) & (2,1) & (2,2) & (2,3) & \cdots \\ (3,0) & (3,1) & (3,2) & (3,3) & \cdots \\ \vdots & \vdots & \vdots & \vdots & \ddots \end{matrix}$$

C. Capacity

Let Σ be a 2-D constraint and let $\mathcal{N}_\Sigma(n,m)$ be the number of patterns of size $n \times m$ satisfying the constraint. If

$$\lim_{\substack{n\to\infty \\ m\to\infty}} \frac{1}{mn} \log \mathcal{N}_\Sigma(n,m)$$

exists then the limit is called the capacity of Σ. By C_h^d we mean the capacity of a constraint specified by h and d.

D. Examples

We give some examples.

1) General example: Consider a function h_0 defined as follows

$$h_0(i,j) = \begin{cases} 100 & \text{if } (i,j) = (0,0); \\ -5 & \text{if } |i| + |j| = 1 \text{ and } ij = 0; \\ -2 & \text{if } |i| = |j| = 1; \\ -1 & \text{if } |i| + |j| = 2 \text{ and } ij = 0; \\ 0 & \text{otherwise.} \end{cases}$$

The following is a graphical representation of h_0.

$$\begin{matrix} 0 & 0 & -1 & 0 & 0 \\ 0 & -2 & -5 & -2 & 0 \\ -1 & -5 & 100 & -5 & -1 \\ 0 & -2 & -5 & -2 & 0 \\ 0 & 0 & -1 & 0 & 0 \end{matrix}$$

The center position of this configuration corresponds to $h_0(0,0)$.

If d is very small, e.g., $d = 30$, then we can expect that many 2 dimensional configurations satisfy our constraint for this h. But if d is very large, e.g., $d = 90$, we expect that there might be no configuration which satisfies our constraint.

2) Simplest example: We consider a function h_1 which can be represented graphically as follows.

$$\begin{matrix} 0 & -2 & 0 \\ -2 & 20 & -2 \\ 0 & -2 & 0 \end{matrix}$$

If $d < 13$ then h_1 imposes no constraint.

There are two interesting cases for this example.

2.1) case E1 $d \geq 17$: This constraint prohibits the following patterns

$$T^d_{h_1} = \left\{ \begin{array}{ccc} * & + & * \\ + & + & + \\ * & + & * \end{array}, \begin{array}{ccc} * & + & * \\ + & + & + \\ * & - & * \end{array}, \begin{array}{ccc} * & + & * \\ + & + & - \\ * & + & * \end{array}, \right. \quad (5)$$

$$\begin{array}{ccc} * & - & * \\ + & + & + \\ * & + & * \end{array}, \begin{array}{ccc} * & + & * \\ - & + & + \\ * & + & * \end{array}, \quad (6)$$

$$\begin{array}{ccc} * & - & * \\ - & - & - \\ * & - & * \end{array}, \begin{array}{ccc} * & - & * \\ - & - & - \\ * & + & * \end{array}, \begin{array}{ccc} * & - & * \\ - & - & + \\ * & - & * \end{array}, \quad (7)$$

$$\left. \begin{array}{ccc} * & + & * \\ - & - & - \\ * & - & * \end{array}, \begin{array}{ccc} * & - & * \\ + & - & - \\ * & - & * \end{array}, \right\} \quad (8)$$

where '$*$' means the 'don't care' symbol, that is, it can be both '$+$' and '$-$'. This constraint can be stated as follows: for every dot two dots of four nearest dots have the same magnetization direction and other two dots have the inverse magnetization direction.

We can show that the capacity of this constraint is 0 (Proposition 2).

2.2) case E2 $13 \leq d < 17$: For this case we have

$$T^d_{h_1} = \left\{ \begin{array}{ccc} * & + & * \\ + & + & + \\ * & + & * \end{array}, \begin{array}{ccc} * & - & * \\ - & - & - \\ * & - & * \end{array}, \right\} \quad (9)$$

That is, this constraint requires that every dot should not be isolated and has been investigated in [7], [8].

3) Second simplest example: We define h_2 by

$$\begin{array}{ccc} -1 & -2 & -1 \\ -2 & 20 & -2 \\ -1 & -2 & -1 \end{array}. \quad (10)$$

This function induces several constraints.

III. Lower bound of capacity

We give lower bounds of capacities of constraints given in the previous section.

If d is relatively large, then first we must know whether the capacity of the given constraint is positive or not. Hence we give a simple test for the positivity of the capacity.

It is easy to see that if we have

$$\left| \sum_{i=-L}^{L} \sum_{\substack{j=-L \\ (i,j)\neq(0,0)}}^{L} h_\ell(i,j) \right| \leq h_\ell(0,0) - d,$$

then h_ℓ and d impose no constraint.

A. h_1

Consider a configuration y defined by

$$y(i,j) = \begin{cases} + & \text{if } j-i \equiv 0 \text{ or } 1 \bmod 3; \\ - & \text{if } j-i \equiv 2 \bmod 3; \end{cases} \quad (11)$$

A part of y is given as follows

$$\begin{array}{ccccccccccccc} + & + & - & + & + & - & + & + & - & + & + & - & \cdots \\ - & + & + & - & + & + & - & + & + & - & + & + & \cdots \\ + & - & + & + & - & + & + & - & + & + & - & + & \cdots \\ + & + & - & + & + & - & + & + & - & + & + & - & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \end{array}$$

Proposition 1: If for positions $(a,b) = (L,L), (L+1, L+1), (L+2, L+2)$, we have

$$\left| \sum_{i=-L}^{L} \sum_{j=-L}^{L} h(i,j) y(a+i, b+j) \right| \geq d, \quad (12)$$

then the capacity specified d and h is positive.

Proof: (Outline) We can see that y is an example of belt patterns for a 2-D $(1,2)$ RLL constraint[9], [10]. We can see that if the constraint satisfies (12) then the constraint satisfies (12) for any belt pattern of the 2-D $(1,2)$ RLL constraint. Since the capacity of the set of belt patterns is strictly positive, we can conclude that the capacity specified by h and d is strictly positive. ■

We give a proof of the following proposition in Appendix.

Proposition 2: The capacity of the constraint defined by h_1 and $d(\geq 17)$ has the capacity 0.

The capacity of the constraint defined by h_1 and $d(13 \leq < 17)$ has been investigated in [7], [8]. Halevy et al. reported an analytical estimate of the capacity of this constraint is 0.91276 and an empirical estimate 0.917 in [7]. They also gave an estimation $0.9238294367\cdots$ by using a method given by Weeks and Blahut [5]. Aviran et al. proposed a context dependent bit-stuffing algorithm and reported that an empirical estimate of the constraint by the algorithm is 0.9223 [8]. Our experimental results are givne in Table I.

We apply Proposition 1 to constraint induced by h_2. Let x_1 and x_2 be patterns

$$\begin{array}{ccc} + & + & - \\ - & + & + \\ + & - & + \end{array} \quad \text{and} \quad \begin{array}{ccc} - & + & + \\ + & - & + \\ + & + & - \end{array},$$

respectively, where we assume that the index of the center positions in these patterns are $(0,0)$.

We can have

$$\left| \sum_{i=-L}^{L} \sum_{j=-L}^{L} x_1(i,j) h_2(i,j) \right| = 18,$$

$$\left| \sum_{i=-L}^{L} \sum_{j=-L}^{L} x_2(i,j) h_2(i,j) \right| = 32.$$

Therefore from Proposition 1 we can conclude that the constraint defined by h_2 and d with $d \leq 18$ has the capacity strictly positive.

Let $C^d_{h_2}$ be the capacity of our constraint specified by h_2 and d. From the definition we see that

$$1 = C^8_{h_2} > C^9_{h_2} = C^{10}_{h_2} \geq \cdots C^{18}_{h_2} > C^{19}_{h_2} = C^{20}_{h_2}. \quad (13)$$

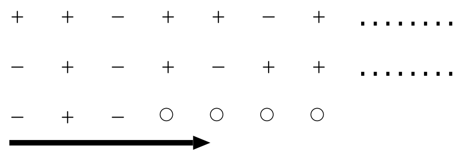

Fig. 3. Direction of Encoding

A constraint defined by h_2 and $d = 19$ (or 20) requires that for each position the sum of $h_2(i,j)$ around the position should be 0. We have $C^d_{h_2} = C^{d+1}_{h_2}$ for an odd integer d because $\left|\sum_i \sum_j h_2(i,j)y(a+i,b+j)\right|$ takes only even numbers.

Experimental results by computer are given in the next section.

IV. Encoding Algorithm

We obtained experimental results on capacities of constraints explained in the previous section by a computer program. The program is an implementation of an algorithm whose main part can be described as follows.

1) $O_+ \leftarrow w(i,j,\text{`+'})$,
2) $O_- \leftarrow w(i,j,\text{`}-\text{'})$,
3) if ($O_+ > 0$ and $O_- > 0$) then {
   a) if ($O_+ > O_-$) then $y(i,j) \leftarrow$ '−',
   b) if ($O_+ \leq O_-$) then $y(i,j) \leftarrow$ '+',
4) }
5) if ($O_+ > 0$ and $O_- \leq 0$) then $y(i,j) \leftarrow$ '−',
6) if ($O_+ \leq 0$ and $O_- > 0$) then $y(i,j) \leftarrow$ '+',
7) if ($O_+ \leq 0$ and $O_- \leq 0$) then $y(i,j) \leftarrow$ current data bit,

where $w(i,j,s)$ is the maximum 'overflow' when we put symbol s at (i,j) position, which is defined by

$$w(i,j,s) = \max_s \max_{(a',b')\in\mathcal{I}} \widetilde{w}(a',b',s)$$

where $\mathcal{I}$ is the set of following indecies

$$\begin{array}{ccc}(i-1,j-1) & (i-1,j) & (i-1,j+1)\\ (i,j-1) & (i,j) & \end{array}$$

and $\widetilde{w}(a',b',s) = d - \left|\sum_{i'=-L}^{L}\sum_{j'=-L}^{L} y(a'+i',b'+j')h(i',j')\right|$.

Our algorithm calculates these steps for every position (i,j) in a plane of size $N \times M$ where N is the number of rows and M the number of columns. For each i it goes from left to right, that is, along the horizontal direction. See Fig. 3. Then it moves to the next row. Hence it needs only a memory of size $3M\left\lceil \log \sum_i \sum_j |h(i,j)| \right\rceil$.

This algorithm tries to make 2-D configurations which satisfy a given constraint but it might violate the constraint when it encounters a 'deadlock pattern.' Such patterns induce executions of step 3)-a) in the algorithm.

We consider the constraint specified by h_1 and $d \geq 17$. We can see that the following pattern is a deadlock pattern

$$\begin{array}{ccc} & & - \\ & + & - & - \\ + & + & x \end{array} \tag{14}$$

where x means the position at which the algorithm is working. Since the capacity of this constraint is 0, this existence seems to be natural. However, we can show that if we initialize the left most row appropriately then we can execute our algorithm without any violation of the constraint. But this fact may be useless for data storage applications. Results of our experiments are given in Table I where numbers are obtained when $N = 5000$, $M = 100000$, and $\Pr\{x_i = \text{`+'}\} = \Pr\{x_i = \text{`}-\text{'}\} = 0.5$.

d	# of Deadlocks	# of Violation	Code rate
10	0	0	0.99999
11	0	0	0.99999
12	0	0	0.99999
13	0	0	0.89652
14	0	0	0.89652
15	0	0	0.77802
16	0	0	0.77802
17	27610	0	0.01392
18	27610	0	0.01392
19	181809	35336	0.00006

TABLE I

Results on h_1

We define $R^d_{h_1}$ to be the code rate given Table I. We see that $R^d_{h_1} = R^{d+1}_{h_2}$ in the table. This relation follows from the fact that the value of $w(i,j,s)$ is always an even number for h_1.

We can show that our algorithm encounter no deadlock pattern for the constraint specified by h_1 and $d(13 \leq d < 17)$ by investigating combinations of prohibited patterns. Therefore we can execute our algorithm without any violation for the constraint.

There are several cases for the constraint specified by h_2. It is not easy to analyze these cases.

Results of our experiments are given in table II with the same parameters of Table I. The result on $C^{19}_{h_2}$ seems to suggest us that $C^{19}_{h_2} = 0$.

d	# of Deadlocks	# of Violation	Code rate
10	0	0	0.98167
11	0	0	0.96680
12	0	0	0.93201
13	0	0	0.92113
14	0	0	0.85195
15	8	1	0.83722
16	1718480	0	0.72371
17	1438	7	0.62599
18	17178648	0	0.47905
19	43947	2470	0.00021

TABLE II

Results on h_2

V. Concluding Remark

We have introduced a model of a constraint which represents a physical constraint occurring in 2-D perpendicular recording media. We have investigated constraints induced from the model for several parameters of the model. One of them is the "no isolated bit" constraint. We have also given an encoding algorithm for those constraint which works well with few exceptions.

Hokkyo et al. have shown that interference caused by adjacent dots in 2-D perpendicular recording media degrades the quality of the media. Therefore we will apply the encoding rule to the media and estimate the coding gain of our encoding rule.

Acknowledgment

This work was partially supported by SRC (Storage Research Consortium).

Appendix: Proof of Proposition 2

Let S_e be a set of (i, j) positions with $(i + j) \equiv 0 \bmod 2$. We consider a 2-D pattern on S_e in which we can find the following pattern

$$\begin{array}{ccc} & + & \\ + & x_0 & - \\ & - & \end{array} \qquad (15)$$

We note that a symbol at a position indicated by x_0 can be both '$+$' and '$-$'. From our constraint we can see that we must have

$$\begin{array}{cccccccc}
\ddots & & + & & & & & \\
 & & + & & & & & \\
 & + & x_{-2} & - & & & & \\
 & & - & x_{-1} & + & & & \\
 & & & + & x_0 & - & & \\
 & & & & - & x_1 & + & \\
 & & & & & + & x_2 & - \\
 & & & & & & - & \\
 & & & & & & & \ddots
\end{array}$$

Hence the pattern of symbols around x_0 determines arrangements of symbols along the diagonal direction.

For this arrangement we have only the following two patterns for right upper positions.

$$\begin{array}{cccccccc}
 & & - & & & & & \\
 & + & & + & & & & \\
+ & x_{-2} & - & & - & & & \\
 & - & x_{-1} & + & & + & & \\
 & & + & x_0 & - & & - & \\
 & & & - & x_1 & + & & + \\
 & & & & + & x_2 & - & \\
 & & & & & - & &
\end{array}$$

and

$$\begin{array}{cccccccc}
 & & + & & & & & \\
 & + & & - & & & & \\
+ & x_{-2} & - & & + & & & \\
 & - & x_{-1} & + & & - & & \\
 & & + & x_0 & - & & + & \\
 & & & - & x_1 & + & & - \\
 & & & & + & x_2 & - & \\
 & & & & & - & &
\end{array}$$

From these observations we can see that there are at most 2^M patterns which contains the pattern (15) at the position indexed by x_0. There are $NM/2$ possible positions for x_0.

The remaining patterns have pattern

$$\begin{array}{ccc} & - & \\ + & & + \\ & - & \end{array} \quad \text{or} \quad \begin{array}{ccc} & + & \\ - & & - \\ & + & \end{array}$$

everywhere. We can have the same estimation for (i, j) positions with $(i + j) \equiv 1 \bmod 2$.

Therefore we have at most

$$(MN2^M + 2)^2$$

possible patterns of size $N \times M$ which satisfy the constraint. This implies that the capacity of this constraint is 0.

References

[1] R. Talyansky, T. Etzion, and R. Roth, "Efficient code constructions for certain two-dimensional constraints," *IEEE Trans. Inform. Theory*, vol. IT-45, pp. 794–799, 1999.

[2] A. Kato and K. Zeger, "On the Capacity of Two-Dimensional Run-Length Constrained Channels," *IEEE Trans. Inform. Theory*, vol. IT-45, pp. 1527–1540, 1999.

[3] T. Ichihashi and H. Kamabe, "The asymptotic property of the rate of two dimensional balanced code," *Technical Report of IEICE*, vol. IT99-235, pp. 43–47, July 1999.

[4] T. Etzion and K. G. Paterson, "Zero/Positive Capacities of Two-Dimensional Runlength-Constrained Arrays," *IEEE Trans. Inform. Theory*, pp. 3186–3199, Sep 2005.

[5] W. Weeks and R. E. Blahut, "The Capacity and Coding Gain of Certain Checkerboard Codes," *IEEE Trans. Inform. Theory*, vol. IT-44, pp. 1193–1203, 1998.

[6] J. Hokkyo, J. Kawaji, and J. Sayama, "Theoretical Approach of Read-Write Characteristics in Perpendicular Media with Nano-composite recording film," *Abstracts of the Inter. Symp. on Creation of Magnetic Recording Materials with Nano-Interfacial Technologies, Waseda Univ.*, p. 34, Dec 2005.

[7] S. Halevy, J. Chen, R. Roth, P. Siegel, and J. Wolf, "Improved Bit-Stuffing Bounds on Two-Dimensional Constraints," *IEEE Trans. Inform. Theory*, vol. IT-50, pp. 824–838, 2004.

[8] S. Aviran, P. Siegel, and J. Wolf, "Two-Dimensional Bit-Stuffing Schemes with Multiple Transformers," *Proceedings of HISC 2005*, pp. 1478–1482, Sep 2005.

[9] H. Kamabe, "Encoding Algorithms for 2 Dimensional Run-Length-Limited Constraints," *Proceedings of 2004 IEEE ISIT*, p. 193, 2004.

[10] H. Kamabe, "Capacity of Two Dimensional Run-Length Limited Constraint," *Proceedings of 3rd Asia-Europe Workshop on Information Theory*, 2003.