Information storage capacity of discrete spin systems

Beni Yoshida

Center for Theoretical Physics, Massachusetts Institute of Technology, Cambridge, Massachusetts 02139, USA

(Dated: January 24, 2012)

What is the limit of information storage capacity of discrete spin systems? To answer this question, we study classical error-correcting codes which can be physically realized as the energy ground space of gapped local Hamiltonians. For discrete spin systems on a $D$-dimensional lattice governed by local frustration-free Hamiltonians, the following bound is known to hold:

$$kd^{1/D} \leq O(n)$$

where $k$ is the number of encodable logical bits, $d$ is the code distance, and $n$ is the total number of spins in the system. Yet, previously found codes were far below this bound and it remained open whether there exists an error-correcting code which saturates the bound or not. Here, we give a construction of local spin systems which saturate the bound asymptotically with $k \sim O(L^{D-1})$ and $d \sim O(L^{D-\epsilon})$ for an arbitrary small $\epsilon > 0$ where $L$ is the linear length of the system. Our model borrows an idea from a fractal geometry arising in Sierpinski triangle.

I. INTRODUCTION AND SUMMARY OF RESULTS

Understanding the limits imposed on information storage capacity of physical systems is a problem of fundamental and practical importance which bridges physics and information science [1]. This problem has been addressed for continuum systems by Bekenstein [2]. He showed that it is impossible to store an infinite amount of information on a finite system and derived the well-celebrated bound on the number of logical bits that can be stored inside a finite region:

$$S \leq \frac{2\pi k_B L E}{hc}$$

where $S$ is the amount of information stored, $L$ is the linear length of the region, and $E$ is the total energy.

The most beautiful outcome concerning the Bekenstein bound is that black holes saturate this theoretical limit [3]. This is essentially due to the observation that an object with a large amount of information (entropy) tends to have high energy, and will eventually turn into a black hole once its energy exceeds a critical value. This surprising connection between information theory and black hole physics is at the heart of the thermodynamic treatment of black holes and the holographic principle [4].

FIG. 1: (a) Storage of information in discrete spin systems via local interactions. (b) A theoretical upper bound on information storage capacity for $D = 2$. The graph is shown in a logarithmic scale. The dotted line represents a family of repetitive codes.
Recently, a similar question on information storage capacity for discrete spin systems on a lattice has been addressed. Consider discrete spin systems defined on a $D$-dimensional lattice which is governed by a local frustration-free commuting Hamiltonian where $D$ is the spatial dimension. Then, the following bound is known to hold [5]:

$$kd^{1/D} \leq O(n)$$  \hspace{1cm} (2)

where $k$ is the number of encoded logical bits, $d$ is the code distance, and $n$ is the total number of spins when the energy ground space of the Hamiltonian is viewed as the codeword space of an error-correcting code. (Note that the code distance $d$ is a quantitative measure of the robustness of encoded information against errors). Such spin systems, called local codes, cover a large class of physically realizable error-correcting codes, and may be viewed as low-density parity-check (LDPC) codes [6] constructed on a space with metric.

Then, one may be naturally led to an analogous question on information storage capacity of discrete spin systems, concerning local codes which saturate the bound in Eq. (2). This is a problem of practical importance since such a local code would be the best error-correcting code that is physically realizable with local Hamiltonians. This problem is also of fundamental importance since such a local code may be viewed as an analog of a black hole for discrete spin systems, and may be useful for further establishing the connection between continuum and discrete descriptions of space-time and quantum gravity [7, 8]. Such a local code may be a candidate of novel quantum phases arising in gapped correlated spin systems which are beyond descriptions of known effective theories.

However, finding a local code which saturates the bound turned out to be a challenging problem. In particular, previously found local codes were far below the bound as seen in Fig. 1(b). To gain some insights on the problem, let us look at a prototypical example of local codes on a two-dimensional lattice ($D = 2$). A repetitive code encodes 0 and 1 into repetitions of zeros and ones; $000\cdots$ and $111\cdots$, and can be physically realized as a local code through ferromagnetic interactions. Since a repetitive code encodes a single logical bit and the Hamming distance (i.e. the number of different spin values) between two codewords is $n$, it has $k = 1$ and $d = n$. On the other hand, for $D = 2$, the bound is $k\sqrt{d} \leq O(n)$, and thus, the repetitive code is far below the theoretical limit. One may modify a repetitive code by splitting the entire lattice into smaller subparts and using them as individual repetitive codes. However, such a construction gives a family of local codes with $kd = n$ as shown with a dotted line in Fig. 1(b), which is still below the bound. Therefore, in order to approach the theoretical limit, one needs to employ more complicated spin configurations instead of repetitive codes.

In this paper, we present a construction of local codes, called fractal codes, which saturate the bound asymptotically:

**Theorem 1.** There exists a local code which can approach the theoretical limit arbitrarily close:

$$k \sim O(L^{D-1}), \quad O(L^{D-\epsilon}) < d \leq O(L^D)$$

for $D \geq 2$ where $\epsilon$ is an arbitrary small positive number, $L$ is the linear length of the lattice and $n = L^D$.

Therefore, our construction gives the best physically realizable error-correcting code that is currently known. Note that $k$ is area-like and $d$ is asymptotically volume-like, and the area law naturally arises here.

Our construction borrows an idea from a well-known example of fractal geometries. Sierpinski triangle has self-similar properties where the same patterns appear repeatedly at different length scales as shown in Fig. 2(a). This peculiar geometric nature of the triangle is reflected in its non-integer dimensionality where the number of filled elements $L^{\log 3/\log 2}$ grows as if the spatial dimension is $\log 3/\log 2 \sim 1.585$ with respect to the linear length of the lattice $L$. While it had been long thought that Sierpinski triangle is a mathematical object, it became apparent that it can be physically realized as a ground state of interacting spin systems via three-body terms [9]. Fig 2(a) shows a physical realization of Sierpinski triangle on a square lattice where interaction terms are minimized when local constraints $a + b = c \mod 2$ on three neighboring spins are satisfied. It has been pointed out that such a fractal system may be
useful as an error-correcting code with an efficient decoder \cite{10}. Recently, coding properties of this fractal code have been predicted as follows \cite{5}:

$$k \sim O(L), \quad d \sim O(L^{\frac{\log 3}{\log 2}})$$

(4)

based on numerical simulations where the fractal dimension of the code distance is equal to the fractal dimension of Sierpinski triangle. This prediction is also shown to be true for infinite lattices. Therefore, this fractal code may be significantly better than previously found local codes such as repetitive codes.

FIG. 2: (a) Sierpinski triangle and its physical realization on a square lattice \((p = 2)\). Filled elements are mapped to 1s while unfilled elements are mapped to 0s. Interaction terms are three-body. (b) A generalization of Sierpinski triangle \((p = 3)\). Black elements are mapped to 1s, grey elements are mapped to 2s, and unfilled elements are mapped to 0s.

Despite a remarkable idea of constructing a local code based on Sierpinski triangle, previous works have two serious drawbacks. First, this fractal code is still far below the theoretical limit as seen in Fig. 1(b). Second, in order to prove the prediction of \(d \sim O(L^{\frac{\log 3}{\log 2}})\), one needs to analyze Hamming distances between all the \(2^L\) ground states and find the minimal Hamming distance, which is a formidable challenge both from analytical and computational perspectives.

We start by presenting the resolution of the first challenge. Our construction of fractal codes utilizes a generalization of Sierpinski triangle with higher-dimensional spins. To begin with, let us discuss fractal properties of Sierpinski triangle with three-dimensional spins where possible spin values are 0, 1, 2 as shown in Fig. 2(b). The number of non-zero spins in this generalized Sierpinski triangle is \(L^{\frac{\log 6}{\log 3}}\), and its fractal dimension is \(\frac{\log 6}{\log 3}\) \(\approx\) 1.631, which is larger than \(\frac{\log 3}{\log 2} \approx 1.585\). Then, one may naturally expect that this generalization gives a fractal code with \(k \sim O(L)\) and \(d \sim O(L^{\frac{\log 6}{\log 3}})\) where \(k\) is the number of encodable three-dimensional logical spins.

The key observation here is that the fractal dimension of Sierpinski triangle grows as the inner dimension of spins increases. In particular, at the limit where \(p\) goes to infinity, we notice

$$D_p^{(2)} = \frac{\log \left( \frac{(p+1)}{2} \right)}{\log p} \to 2 \quad \text{for} \quad p \to \infty.$$ 

(5)

Therefore, by taking sufficiently large \(p\), one can construct a fractal code with \(k^{(p)} \sim O(L)\) and \(d \geq O(L^{2-\epsilon})\) for an arbitrary small \(\epsilon > 0\) where \(k^{(p)}\) is the number of encodable \(p\)-dimensional spins. This family of fractal codes based
on generalized Sierpinski triangle will saturate the bound in Eq. (2) asymptotically. While our construction of fractal codes uses \( p \)-dimensional spins with \( p > 2 \), one can simulate these fractal codes through two-dimensional spins.

Then, what about the bound on higher-dimensional systems with \( D > 2 \)? Fortunately, there exist higher-dimensional generalizations of Sierpinski triangle constructed on a \( D \)-dimensional hypercubic lattice (see [11] for example). For \( D \)-dimensional Sierpinski triangle with \( p \)-dimensional spins, its fractal dimension is given by

\[
D_p^{(D)} = \log \left( \frac{p(p+1)\cdots(p+D-1)}{D!} \right) / \log(p) \tag{6}
\]

which approaches to \( D \) as \( p \) goes to infinity: \( D_p^{(D)} \to D \) for \( p \to \infty \). A fractal code based on \( D \)-dimensional Sierpinski triangle has \( k(p) \sim O(L^{D-1}) \) and \( d \sim O(L^{D_p^{(D)}}) \), and one can construct fractal codes which saturate the bound asymptotically in any spatial dimension.

While our primary interest is in error-correcting codes which can be physically realized, it is worth considering the behaviors of coding properties as the spatial dimension \( D \) increases. In particular, by taking sufficiently large \( D \) and \( p \), one can construct a local code with

\[
k \sim O(n^{1-\epsilon}), \quad d \sim O(n^{1-\epsilon'}) \tag{7}
\]

where \( \epsilon \) and \( \epsilon' \) are arbitrary small positive numbers since \( \epsilon' \sim 1/D \). While the information storage rate \( R = k/n \) decreases as the system size \( n \) becomes large, the rate \( R \) decreases as slow as an arbitrary polynomial since \( R = O(n^{-\epsilon'}) \). It is interesting to observe that a good error-correcting code can be obtained by considering the limit where the spatial dimension of fractal codes becomes infinite.

Discussion above is valid only if the assumption that the fractal dimension of the code distance is equal to the fractal dimension of Sierpinski triangle is true:

**Theorem 2** (Fractal dimension of code distance). In fractal codes, the fractal dimension of the code distance \( d \) is equal to the fractal dimension of Sierpinski triangle:

\[
k \sim O(L^{D-1}) \quad d \sim O(L^{D_p^{(D)}}) \tag{8}
\]

where \( D_p^{(D)} \) is the fractal dimension of \( D \)-dimensional Sierpinski triangle constructed with \( p \)-dimensional spins, and \( k \) is the number of encodable logical \( p \)-dimensional spins.

The rest of the paper is dedicated to the proof of this theorem. In section II we discuss two-dimensional cases. In section III we discuss three-dimensional and higher-dimensional cases. In section IV we list possible future problems. The paper is written in a self-consistent way, and most of non-trivial mathematical proofs are presented in appendix, so the main discussion is accessible to readers both in coding theory and physics community. The main technical result of this paper is lemma 3 concerning the weights of raw vectors of Sierpinski triangle, which may be of interest by its own.

## II. TWO-DIMENSIONAL FRACTAL CODE

In this section, we introduce local codes, which we call fractal codes, on a two-dimensional lattice and show that there exist fractal codes which asymptotically saturate the bound in Eq. (2) for \( D = 2 \).

### A. Definition of fractal codes

We begin with the definition of fractal codes. Consider a two-dimensional square lattice with \( n = L \times L \) spins where spins are \( p \)-dimensional and spin values are \( 0, \cdots, p - 1 \). We assume that \( p \) is a prime number, and \( L = p^m \).
with arbitrary positive integer $m$. Each spin is labeled by “time” $t$ and “position” $r$ where $t, r = 0, \cdots, L - 1$. We set periodic boundary conditions along the time axis, and set open boundary conditions along the position axis (see Fig. 3). Note that $t$ and $r$ run from 0 to $L - 1$, instead of running from 1 to $L$.

FIG. 3: The construction of fractal codes. The example above shows the case with $p = 2$ and $L = 8$ ($m = 3$). Periodic boundary conditions are set along the time axis. Admissible spin configurations of a fractal code can be realized as ground states of a three-body Hamiltonian.

The admissible spin configurations of the system obeys the following local constraint:

$$x(t + 1)_r = x(t)_{r - 1} + x(t)_r \quad (\text{mod } p) \quad 0 \leq t \leq L - 2$$

(9)

where $x(t)_r = 0, \cdots, p - 1$ represents the spin value at $(t, r)$. Notice that such spin configurations can be physically realized as ground states of the following three-body local Hamiltonian:

$$H_{\text{fractal}} = \sum_{t,r} \Pi(t)_r, \quad \Pi(t)_r = x(t + 1)_r - x(t)_{r - 1} - x(t)_r \quad (\text{mod } p)$$

(10)

with a finite energy gap. There are $p^L$ admissible spin configurations which can be specified by the “initial condition” $(x(0)_0, \cdots, x(0)_{L - 1})$ for $t = 0$ on the first raw of a square lattice (see Fig. 3). The constraint given in Eq. (9) is very simple, involving only three neighboring spins at once, and is translationally symmetric except at the boundary. Yet, the arising spin configurations have highly non-trivial patterns as seen in Fig. 3. Our main idea is to utilize such non-trivial structures of spin configurations for reliable encoding of information. Here, the entire system can be viewed as a “computational machine” which computes a vector $x(t) = (x(t)_0, \cdots, x(t)_{L - 1})$ at time $t$ for a given initial condition $x(0) = (x(0)_0, \cdots, x(0)_{L - 1})$ after the time evolution according to the rule in Eq. (9). In this light, our code may be considered as a physical realization of the time evolution of one-dimensional cellular automaton [11].

Now, we construct the fractal codes based on admissible spin configurations obeying Eq. (9). Here, we further limit ourselves to spin configurations which satisfy the following initial condition:

$$x(0)_r = 0 \quad \text{for} \quad r > p^{m - 1}.$$  

(11)

This constraint may be physically realized by setting additional terms on the boundary of the lattice:

$$H_{\text{boundary}} = \sum_{r > p^{m - 1}} x(0)_r.$$  

(12)
We denote a space of spin configurations specified by Eq. (9) and Eq. (11) as $C_p^{(2)}$, and call it the \textit{codeword space} of a fractal code. Coding properties of fractal codes are summarized in the following theorem.

**Theorem 3** (Two-dimensional fractal code). For the codeword space $C_p^{(2)}$ specified by Eq. (9) and Eq. (11), let $k$ be the number of encodable $p$-dimensional spins and $d$ be the code distance of the code (i.e. the minimal Hamming distance among all the possible spin configurations). Then, we have

$$k = p^{m-1} + 1, \quad \left(\frac{p-1}{p+1}\right) \cdot \left(\frac{p(p+1)}{2}\right)^m \leq d \leq \left(\frac{p(p+1)}{2}\right)^m. \quad (13)$$

Thus, as $L = p^m$ becomes large, we have

$$k \sim O(L), \quad d \sim O(LD_p^{(2)}) \quad \text{where} \quad D_p^{(2)} = \log \left(\frac{p(p+1)}{2}\right)/\log(p). \quad (14)$$

Here, we notice that $D_p^{(2)}$ increases as $p$ increases. In particular, since $D_p^{(2)} \to 2$ for $p \to \infty$, we can construct a local code which asymptotically saturates the bound $k\sqrt{d} \leq O(n)$ in Eq. (2). The rest of this section is dedicated to the proof of theorem 3.

The reason why we limit our considerations to spin configurations obeying Eq. (11) comes from a technical difficulty which is not particularly interesting. For $p = 2$ and an initial condition $(x(0)_0, \cdots, x(0)_{L-1}) = (1, \cdots, 1)$, the resulting spin configurations are $(x(t)_0, \cdots, x(t)_{L-1}) = (0, \cdots, 0)$ for $t > 0$ which leads to $d = L$. To avoid this difficulty, we need Eq. (11). This is closely related to the irreversibility of the time evolution of cellular automaton for $p = 2$.

\textbf{B. Principal vector and fractal dimension}

\textbf{Principal vector:} We begin by analyzing a spin configuration specified by the following initial condition (see Fig. 4):

$$x(0)_0 = 1, \quad x(0)_r = 0 \quad (r \neq 0). \quad (15)$$

We denote spin values of the $t$-th row in such a spin configuration as $B(t)$ where

$$B(t) = (x(t)_0, \cdots, x(t)_{L-1})$$

and call them \textit{principal vectors}. For example, with $m = 2$ and $p = 2$, we have the following principal vectors:

$$B(0) = (1, 0, 0, 0), \quad B(1) = (1, 1, 0, 0), \quad B(2) = (1, 0, 1, 0), \quad B(3) = (1, 1, 1, 1).$$

We represent the spin configuration associated with the initial condition in Eq. (15) as an $L \times L$ matrix, and denote it by $\mathbf{B}$. For example, with $m = 2$ and $p = 2$, we have

$$\mathbf{B} = \begin{bmatrix} 1, & 0, & 0, & 0 \\ 1, & 1, & 0, & 0 \\ 1, & 0, & 1, & 0 \\ 1, & 1, & 1, & 1 \end{bmatrix}.$$ We show several other examples of $\mathbf{B}$ and $B(t)$ in Fig 4. We call $\mathbf{B}$ the \textit{Pascal matrix} modulo $p$ due to its resemblance to Pascal’s triangle which is also known as Sierpinski triangle.

\textbf{Fractal dimension:} Entries in principal vectors and the Pascal matrix can be derived easily from a simple formula. The lemma below summarizes how to compute the $r$-th entry of a principal vector $B(t)$, denoted as $B(t)_r$, for arbitrary $t$ and $r$. 


FIG. 4: Examples of principal vectors $B(t)$ and the Pascal matrices $B$. (a) $p = 2$ and $m = 3$. (b) $p = 3$ and $m = 2$. (c) $p = 5$. (d) $p = 7$ and $m = 1$.

**Lemma 1** (Entries of the Pascal matrix). *Let us represent $t$ and $r$ in $p$-adic forms:*

$$r = (r_m, r_{m-1}, \cdots, r_1)_p, \quad r = \sum_{m' = 1}^{m} p^{m'-1} r_{m'}$$

$$t = (t_m, t_{m-1}, \cdots, t_1)_p, \quad t = \sum_{m' = 1}^{m} p^{m'-1} t_{m'}$$

where $r_j$ and $t_j$ are positive integers with $0 \leq r_j, t_j \leq p - 1$. Then, we have

$$B(t)_r = tC_r \quad (\text{mod } p) \quad (16)$$

where $tC_r = 0$ for $r > t$. In particular,

$$tC_r = \prod_{m' = 1}^{m} t_{m'} C_{r_{m'}} \quad (17)$$

and

$$tC_r \neq 0 \quad (\text{mod } p) \quad \text{iff} \quad t_{m'} \geq r_{m'} \quad \text{for all } m'. \quad (18)$$

We skip the proof of lemma [1] Some examples are shown in Fig. [5] As a direct consequence of the lemma above, we have the following corollary on the weight of the Pascal matrix $B$ (See Fig. [6]):
FIG. 5: Entries of the Pascal matrix. $t$ and $r$ are represented in $p$-adic forms. (a) $p = 2$ and $m = 3$. Shaded entries represent one, while unfilled entries represent zero. (b) $p = 3$ and $m = 2$. Shaded entries represent one, black entries represent two, while unfilled entries represent zero.

**Corollary 1** (Fractal dimension). For principal vectors $B(t)$, we denote its Hamming weights (the number non-zero entries) as $W(B(t))$. Similarly, we denote the Hamming weight of a the Pascal matrix $B$ as $W(B)$. Then, the number of spins with non-zero entries is

$$W(B) = \sum_{t=0}^{L-1} W(B(t)) = \left( \frac{p(p + 1)}{2} \right)^m.$$  

(19)

Thus, the fractal dimension of the Pascal matrix $B$ is

$$W(B) = L^{D_p^{(2)}} \quad \text{where} \quad D_p^{(2)} = \log \left( \frac{p(p + 1)}{2} \right) / \log p.$$  

(20)

Proof. When $p = 2$, the number of non-zero entries corresponds to the number of pairs of $t$ and $r$ such that

$$t_{m'} \geq r_{m'} \quad \text{for all } m'$$

from lemma 1. Then, we notice that only the following three pairs of $t_{m'}, r_{m'}$ are allowed:

$$(t_{m'}, r_{m'}) = (0, 0), (1, 0), (1, 1).$$

Therefore, in total, there are $3^m$ non-zero entries. For $p > 2$, there are $\frac{p(p+1)}{2}$ possible pairs of $t_{m'}$ and $r_{m'}$. Therefore, in total, there are $\left( \frac{p(p+1)}{2} \right)^m$ non-zero entries. \hfill $\square$

**C. Time evolution of principal vectors**

We have analyzed a spin configuration $B$ arising from an initial condition $B(0) = (1, 0, 0, 0, \cdots)$. In this subsection, we extend our analysis to spin configurations arising from arbitrary initial conditions.

Let us denote an arbitrary initial condition as $V(0)$ and its $t$-th raw as $V(t)$. Since principal vectors $B(t)$ are independent, one can decompose $V(0)$ as a linear combination of $B(t)$:

$$V(0) = \sum_{t=0}^{L-1} c(t)B(t) \quad (\mod p)$$  

(21)
where \( c(t) = 0, \ldots, p - 1 \). Then, our goal is to find an expression for \( V(t) \) in terms of principal vectors \( B(\tau) \) for arbitrary \( t \).

**Lemma 2** (Time evolution). We define \( B(L + a) = 2B(a) \) for \( a \geq 0 \). Then, when \( V(0) = \sum_\tau c(\tau)B(\tau) \), we have

\[
V(t) = \sum_\tau c(\tau)B(\tau + t) \quad (\text{mod } p).
\] (22)

When \( V(0) = B(\tau) \) \((\tau \neq 0)\), the time evolution is

\[
B(\tau) \rightarrow B(\tau + 1) \rightarrow \cdots \rightarrow B(L - 1) \rightarrow 0 \rightarrow \cdots \rightarrow 0
\]

for \( p = 2 \), while

\[
B(\tau) \rightarrow B(\tau + 1) \rightarrow \cdots \rightarrow B(L - 1) \rightarrow 2B(0) \rightarrow \cdots \rightarrow 2B(\tau - 1)
\]

for \( p > 2 \).

While we are primarily interested in fractal codes with periodic boundary conditions, one may generalize the analysis to fractal codes with open boundary conditions. In such cases, the time evolution is slightly modified since \( B(L - 1) \) always evolves into \( B(0) \) regardless of \( p: B(L - 1) \rightarrow B(0) \). A similar discussion holds for codes with open boundary conditions, and one can obtain a similar bound on the code distance.

**D. Inequality on principal vectors**

Finally, we prove theorem 3. Our proof relies on the following lemma on the weights of vectors represented in terms of principal vectors:

**Lemma 3** (Inequality on principal vectors). Consider the following linear combination of principal vectors:

\[
V(0) = \sum_{t=0}^{L-1} c(t)B(t)
\]

and denote the smallest positive integer \( t \) such that \( c(t) \neq 0 \) as \( t_{\text{min}} \). Then, for the Hamming weight of \( V(0) \), we have

\[
W(V(0)) \geq W(B(t_{\text{min}})).
\] (23)
This is one of the most important technical tools obtained in this paper. The proof is presented in appendix A. It is worth looking at an example. Let us consider the case where \( p = 3 \) and \( m = 2 \):

\[
B(2) = (1, 2, 1, 0, 0, 0, 0, 0), \quad B(5) = (1, 2, 1, 1, 2, 0, 0, 0), \quad B(7) = (1, 1, 0, 2, 0, 1, 1, 0)
\]

\[
B(2) + B(5) + B(7) = (0, 2, 2, 0, 1, 1, 1, 0).
\]

Then, we notice

\[
W(B(2) + B(5) + B(7)) \geq W(B(2))
\]

and the lemma holds.

The proof of theorem 3 also requires the following fact on the self-similar properties of principal vectors:

**Fact 1.** We have

\[
W(B(t + b \cdot p^{m-1})) = (b + 1)W(B(t))
\]

for \( 0 \leq b < p \) and \( 0 \leq t < p^{m-1} \).

We are now ready to prove theorem 3. We consider a spin configuration generated by an initial condition \( V(0) \):

\[
V(0) = \sum_{\tau=0}^{p^{m-1}-1} c(\tau)B(\tau)
\]

which obeys Eq. (11). We denote the smallest \( t \) such that \( c(t) \neq 0 \) as \( t_{min} \). We denote the \( t \)-th row of the spin configuration as \( V(t) \), and denote the entire spin configuration matrix as \( \mathbf{V} \). Then, due to lemma 3 we have

\[
W(V(t)) \geq W(B(t_{min} + t)) \quad \text{for} \quad 0 \leq t \leq p^m - p^{m-1} - 1,
\]

which leads to

\[
W(\mathbf{V}) = \sum_t W(V(t)) \geq \sum_{t=0}^{p^m-p^{m-1}-1} (B(t_{min} + t)).
\]

Then, by fact 1 we have

\[
\sum_{t=0}^{p^m-p^{m-1}-1} (B(t_{min} + t)) \geq \sum_{t=0}^{p^m-p^{m-1}-1} (B(t)) = \frac{p-1}{p+1} \cdot L^{D_p^{(2)}}.
\]

This completes the proof of theorem 3.

### III. THREE-DIMENSIONAL FRACTAL CODE

The construction of two-dimensional fractal codes can be generalized to higher-dimensional systems too. In this section, we introduce the three-dimensional version of fractal codes and prove theorem 1 for \( D = 3 \). Also, higher-dimensional cases with \( D > 3 \) are briefly discussed.

**A. Definition of fractal codes**

We consider a three-dimensional cubic lattice with \( n = L \times L \times L \) spins where spins are \( p \)-dimensional and \( L = p^m \). Each spin is labeled by “time” \( t \) and two “positions” \( r^{(1)} \) and \( r^{(2)} \) with \( t, r^{(1)}, r^{(2)} = 0, \cdots, L-1 \). We set
periodic boundary conditions on all the surfaces which are parallel to the time axis (see Fig. 7). The admissible spin configurations obey the following constraint:

\[ x(t + 1)_{r(1), r(2)} = x(t)_{r(1) - 1, r(2)} + x(t)_{r(1), r(2) - 1} + x(t)_{r(1), r(2)} \pmod{p} \quad 0 \leq t \leq L - 2 \quad (25) \]

where \( x(t)_{r(1), r(2)} = 0, \cdots, p - 1 \) represents the spin value at \( (t, r^{(1)}, r^{(2)}) \). Spin configurations may be uniquely specified by “initial conditions” \( x(0) \) with \( x(0)_{r(1), r(2)} = 0, \cdots, p - 1 \), which may be considered as \( L \times L \) matrices.

\[ \begin{align*}
\text{FIG. 7: The construction of three-dimensional fractal codes. The example above shows the case with } p = 2 \text{ and } m = 2. \\
\end{align*} \]

We further limit ourselves to spin configurations which satisfy the following initial condition (see Fig. 7(c)):

\[ x(0)_{r(1), r(2)} = 0 \quad \text{for} \quad r^{(1)} + r^{(2)} > p^{m-1}, \quad (26) \]

and denote a space of spin configurations specified by this condition as \( C_p^{(3)} \). Then, our main result is summarized in the following theorem:

**Theorem 4 (Three-dimensional fractal code).** For the codeword space \( C_p^{(3)} \), let \( k \) be the number of encodable \( p \)-dimensional spins and \( d \) be the code distance of the code. Then, we have

\[ k = \frac{(p^{m-1} + 1)(p^{m-1} + 2)}{2}, \quad \left( \frac{p - 1}{p + 2} \right)^m \leq d \leq \left( \frac{p(p + 1)(p + 2)}{6} \right)^m \quad (27) \]

where \( L = p^m \). Thus, we have

\[ k \sim O(L), \quad d \sim O(L^{D_p^{(3)}}) \quad \text{where} \quad D_p^{(3)} = \log \left( \frac{p(p + 1)(p + 2)}{6} \right) / \log(p). \quad (28) \]

When \( D = 3 \), the fractal dimension goes to three: \( D_p^{(3)} \to 3 \) for \( p \to \infty \). Therefore, the code saturates the bound \( kd^{1/3} \leq O(n) \) in Eq. (2) for \( D = 3 \) asymptotically. Below, we present the proof of theorem 4.

**B. Principal matrix**

Let us begin by analyzing a spin configuration specified by the following initial condition:

\[ x(0)_{0,0} = 1, \quad x(0)_{r(1), r(2)} = 0 \quad \text{for} \quad (r^{(1)}, r^{(2)}) \neq (0, 0) \quad (29) \]
which may be represented in the following matrix form:

\[ B(0, 0) = \begin{bmatrix}
1, 0, \cdots, 0 \\
0, 0, \cdots, 0 \\
\vdots & \ddots & \vdots \\
0, 0, \cdots, 0
\end{bmatrix} \]  \hspace{1cm} (30)

We denote the spin configuration tensor, generated by the initial condition \( B(0, 0) \), as \( B \), and call it the Pascal tensor. We also denote a matrix representing a spin configuration on the \( t \)-th layer of the Pascal tensor \( B \) as \( B(t, 0) \), and call them principal matrices. Therefore, the time evolution of the initial condition \( B(0, 0) \) may be represented as follows:

\[ B(0, 0) \rightarrow B(0, 1) \rightarrow \cdots \rightarrow B(0, L - 1). \]  \hspace{1cm} (31)

When \( p = 2 \) and \( m = 2 \), principal matrices are

\[ B(0, 0) = \begin{bmatrix} 1, 0, 0, 0 \\ 0, 0, 0, 0 \\ 0, 0, 0, 0 \\ 0, 0, 0, 0 \end{bmatrix} \rightarrow B(0, 1) = \begin{bmatrix} 1, 1, 0, 0 \\ 1, 0, 0, 0 \\ 0, 0, 0, 0 \\ 0, 0, 0, 0 \end{bmatrix} \rightarrow B(0, 2) = \begin{bmatrix} 1, 0, 1, 0 \\ 0, 0, 0, 0 \\ 1, 0, 0, 0 \\ 0, 0, 0, 0 \end{bmatrix} \rightarrow B(0, 3) = \begin{bmatrix} 1, 1, 1, 1 \\ 1, 0, 1, 0 \\ 1, 1, 0, 0 \\ 1, 0, 0, 0 \end{bmatrix}. \]

Other examples are shown in Fig. 8.

\[ \begin{array}{|c|c|c|}
\hline
\text{B(0,0)} & \text{B(0,1)} & \text{B(0,2)} \\
\hline
1 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 2 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 2 & 2 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\hline
\text{B(0,3)} & \text{B(0,4)} \\
\hline
1 & 3 & 3 & 1 & 0 & 1 & 4 & 1 & 4 & 1 \\
3 & 1 & 3 & 0 & 4 & 2 & 2 & 4 & 0 \\
3 & 3 & 0 & 0 & 1 & 2 & 1 & 0 & 0 \\
1 & 0 & 0 & 0 & 4 & 4 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
\hline
\end{array} \]

FIG. 8: Examples of principal matrices \( B(0, t) \) for \( p = 5 \) and \( m = 1 \).

Now, we derive a formula to find spin configurations of principal matrices \( B(0, t) \). For \( 0 \leq t, r^{(1)}, r^{(2)} \leq L - 1 \), we define

\[ tC_{r^{(1)},r^{(2)}} = \frac{t!}{r^{(1)}!r^{(2)}!(t - r^{(1)} - r^{(2)})!} = tC_{r^{(1)}} \cdot t^{r^{(1)}}C_{r^{(2)}}. \]  \hspace{1cm} (32)

The lemma below summarizes how to compute the entry at \( (r^{(1)}, r^{(2)}) \) in \( B(0, t) \), denoted as \( B(0, t)_{r^{(1)},r^{(2)}} \), for arbitrary \( t, r^{(1)} \) and \( r^{(2)} \):
Lemma 4 (Entries of the Pascal tensor). Let us represent \( t, r^{(1)} \) and \( r^{(2)} \) in \( p \)-adic forms:

\[
\begin{align*}
  r^{(1)} &= (r_m^{(1)} r_{m-1}^{(1)} \cdots r_1^{(1)}), & r &= \sum_{m'=1} p^{m'-1} r_{m'} \\
  r^{(2)} &= (r_m^{(2)} r_{m-1}^{(2)} \cdots r_1^{(2)}), & r &= \sum_{m'=1} p^{m'-1} r_{m'} \\
  t &= (t_m t_{m-1} \cdots t_1), & t &= \sum_{m'=1} p^{m'-1} t_{m'}.
\end{align*}
\]

Then, we have

\[
B(0, t)_{r^{(1)}, r^{(2)}} = t C_{r^{(1)}, r^{(2)}} \pmod{p}.
\]

In particular,

\[
t C_{r^{(1)}, r^{(2)}} = \prod_{m'} t_{m'} C_{r^{(1)}_{m'}, r^{(2)}_{m'}}
\]

and

\[
t C_{r^{(1)}, r^{(2)}} \neq 0 \pmod{p} \quad \text{if and only if} \quad t_{m'} \geq r^{(1)}_{m'} + r^{(2)}_{m'} \quad \text{for all} \ m'.
\]

![FIG. 9: Entries of principal matrices for \( p = 2 \) and \( m = 2 \).](image)

An example is shown in Fig. 9. The proof is skipped. As a direct consequence of the lemma above, we have the following corollary on the weight of the Pascal tensor \( B \):

Corollary 2 (Fractal dimension). The number of non-zero entries in the Pascal tensor is

\[
W(B) = \sum_{t=0}^{L-1} W(B(0, t)) = \left( \frac{p(p+1)(p+2)}{6} \right)^m.
\]

Thus, the fractal dimension is

\[
W(B) = L^{D_p^{(3)}} \quad \text{where} \quad D_p^{(3)} = \log \left( \frac{p(p+1)(p+2)}{6} \right) / \log p.
\]

Proof. In order for \( B(0, t)_{r^{(1)}, r^{(2)}} \) to be non-zero, \((t_{m'}, r_{m'}^{(1)}, r_{m'}^{(2)}) \) must satisfy \( t_{m'} \geq r_{m'}^{(1)} + r_{m'}^{(2)} \) for all \( m' \). The number of such choices for each \( m' \) is

\[
\sum_{j=1}^p \frac{j(j+1)}{2} = \frac{p(p+1)(p+2)}{6}.
\]

\[\square\]
We continue our analysis on basic properties of principal matrices $B(0, t)$. We introduce the following shorthand notation of principal matrices $B(0, t)$:

\[
B(0, t) = \begin{bmatrix}
B(0, t)_0 \\
B(0, t)_1 \\
\vdots \\
B(0, t)_{L-1}
\end{bmatrix}
\]  

(38)

where $B(0, t)_j$ are $L$-component vectors with

\[
B(0, t)_j = (B(0, t)_0, j, B(0, t)_1, j, \cdots, B(0, t)_{L-1}, j).
\]  

(39)

For example, when $p = 2$ and $m = 2$, we have

\[
B(0, 3) = \begin{bmatrix}
1, 1, 1, 1 \\
1, 0, 1, 0 \\
1, 1, 0, 0 \\
1, 0, 0, 0
\end{bmatrix},
\]

and

\[
B(0, 3)_0 = (1, 1, 1, 1), \quad B(0, 3)_1 = (1, 0, 1, 0), \quad B(0, 3)_2 = (1, 1, 0, 0), \quad B(0, 3)_3 = (1, 0, 0, 0).
\]

Therefore, one may represent $B(0, 3)$ as follows:

\[
B(0, 3) = \begin{bmatrix}
B(3) \\
B(2) \\
B(1) \\
B(0)
\end{bmatrix}
\]

where $B(0), B(1), B(2)$ and $B(3)$ are principal vectors used in analyzing two-dimensional fractal codes. We also have

\[
B(0, 2) = \begin{bmatrix}
1, 0, 1, 0 \\
0, 0, 0, 0 \\
1, 0, 0, 0 \\
0, 0, 0, 0
\end{bmatrix} = \begin{bmatrix}
B(2) \\
0 \\
B(0)
\end{bmatrix},
\]

As examples above show, principal matrices $B(0, t)$ can be concisely represented in terms of principal vectors $B(t)$. Other examples are shown in Fig. 10 too. One may represent principal matrices $B(0, t)$ in terms of principal vectors $B(t)$ as follows:

Lemma 5 (Principal matrix). A principal matrix $B(0, t)$ can be represented as

\[
B(0, t) = \begin{bmatrix}
B(t)_0 \cdot B(t) \\
B(t)_1 \cdot B(t - 1) \\
\vdots \\
B(t)_{L-1} \cdot B(t - L + 1)
\end{bmatrix}
\]  

(40)

We skip the proof as it is straightforward. As an example, let us represent $B(0, 6)$ for $p = 2$ and $m = 3$ in this shorthand representation (see Fig. 10):

\[
B(6) = (1, 0, 1, 0, 1, 0, 0), \quad B(0, 6) = (B(6), 0, B(4), 0, B(2), 0, B(0), 0)^T
\]
where 0 represents vectors with zero entries. Similarly, we can represent \( B(0,7) \) for \( p = 3 \) and \( m = 2 \) as follows:

\[
B(7) = (1, 1, 0, 2, 2, 0, 1, 1, 0), \quad B(0, 7) = (B(7), B(6), 0, 2B(4), 2B(3), 0, B(1), B(0), 0)^T.
\]

It is worth representing all the principal matrices \( B(0, t) \) at once as in Fig. 10(b). Therefore, in a principal matrix \( B(0, t) \), principal vectors \( B(0), \ldots, B(t) \) are distributed with weights corresponding to a principal vector \( B(t) \).

C. Generalized principal matrix

So far, we have analyzed the spin configuration generated by the following initial condition:

\[
B(0, 0) \equiv \begin{bmatrix} B(0) \\ 0 \\ 0 \\ 0 \end{bmatrix}. \quad (41)
\]

In this subsection, we generalize our analyses and consider spin configurations generated by other initial conditions:

\[
B(a, 0) \equiv \begin{bmatrix} B(a) \\ 0 \\ 0 \\ 0 \end{bmatrix} \quad (\text{for } a = 0, \ldots, L - 1). \quad (42)
\]

We denote the \( t \)-th layer of the spin configuration generated by \( B(a, 0) \) as \( B(a, t) \). For simplicity, we will call \( B(a, t) \) for both \( a = 0 \) and \( a \neq 0 \) principal matrices from now on.

One may represent \( B(a, t) \) explicitly as follows:
Lemma 6 (Generalized principal matrix). A principal matrix $B(a, t)$ can be represented as

$$B(a, t) = \begin{bmatrix} B(t)_0 \cdot B(t + a) \\ B(t)_1 \cdot B(t + a - 1) \\ \vdots \\ B(t)_{L-1} \cdot B(t + a - L + 1) \end{bmatrix}$$

where $B(\tau + L) = 2B(\tau)$.

Below, we show some examples. For $p = 2$ and $m = 2$, we have:

$$B(0, 0) = \begin{bmatrix} B(0) \\ B(0) \\ 0 \\ 0 \end{bmatrix}, \quad B(0, 1) = \begin{bmatrix} B(1) \\ B(2) \\ 0 \\ 0 \end{bmatrix}, \quad B(0, 2) = \begin{bmatrix} B(2) \\ 0 \\ B(0) \\ 0 \end{bmatrix}, \quad B(0, 3) = \begin{bmatrix} B(3) \\ B(2) \\ B(1) \\ B(0) \end{bmatrix}$$

$$B(1, 0) = \begin{bmatrix} B(1) \\ B(1) \\ 0 \\ 0 \end{bmatrix}, \quad B(1, 1) = \begin{bmatrix} B(2) \\ B(2) \\ 0 \\ 0 \end{bmatrix}, \quad B(1, 2) = \begin{bmatrix} B(3) \\ 0 \\ B(1) \\ 0 \end{bmatrix}, \quad B(1, 3) = \begin{bmatrix} B(3) \\ B(2) \\ B(1) \\ B(0) \end{bmatrix}$$

$$B(2, 0) = \begin{bmatrix} B(2) \\ B(2) \\ 0 \\ 0 \end{bmatrix}, \quad B(2, 1) = \begin{bmatrix} B(3) \\ B(3) \\ 0 \\ 0 \end{bmatrix}, \quad B(2, 2) = \begin{bmatrix} B(3) \\ 0 \\ B(2) \\ 0 \end{bmatrix}, \quad B(2, 3) = \begin{bmatrix} 0 \\ 0 \\ 0 \\ B(2) \end{bmatrix}$$

$$B(3, 0) = \begin{bmatrix} B(3) \\ B(3) \\ 0 \\ 0 \end{bmatrix}, \quad B(3, 1) = \begin{bmatrix} B(3) \\ B(3) \\ 0 \\ 0 \end{bmatrix}, \quad B(3, 2) = \begin{bmatrix} B(3) \\ B(3) \\ 0 \\ 0 \end{bmatrix}, \quad B(3, 3) = \begin{bmatrix} B(3) \\ B(3) \\ 0 \\ 0 \end{bmatrix}.$$

For $p = 3$ and $m = 1$, we have

$$B(0, 0) = \begin{bmatrix} B(0) \\ 0 \\ 0 \end{bmatrix}, \quad B(0, 1) = \begin{bmatrix} B(1) \\ B(0) \\ 0 \end{bmatrix}, \quad B(0, 2) = \begin{bmatrix} B(2) \\ 2B(1) \\ B(0) \end{bmatrix}$$

$$B(1, 0) = \begin{bmatrix} B(1) \\ 0 \\ 0 \end{bmatrix}, \quad B(1, 1) = \begin{bmatrix} B(2) \\ B(1) \\ 0 \end{bmatrix}, \quad B(1, 2) = \begin{bmatrix} 2B(0) \\ 2B(2) \\ B(1) \end{bmatrix}$$

$$B(2, 0) = \begin{bmatrix} B(2) \\ 0 \\ 0 \end{bmatrix}, \quad B(2, 1) = \begin{bmatrix} 2B(0) \\ B(2) \\ 0 \end{bmatrix}, \quad B(2, 2) = \begin{bmatrix} 2B(1) \\ B(2) \\ 0 \end{bmatrix}.$$

Now, let us analyze the time evolution of principal matrices. Recall that $B(a, 0)$ evolves as follows:

$$B(a, 0) \rightarrow B(a, 1) \rightarrow \cdots \rightarrow B(a, L - 1).$$

Here, we are interested in the time evolution of $B(a, \tau)$:

Lemma 7 (Periodic boundary conditions). We define $B(a, L + t) = 3B(a, t)$. When the initial condition is $V(0) = B(a, \tau)$, we have

$$V(t) = B(a, \tau + t).$$
D. Inequality on principal matrices

Finally, we prove theorem 4. We begin by noticing that there are $L^2$ principal matrices and they are all independent. Then, one can decompose an arbitrary initial condition $V(0)$ in terms of principal matrices $B(a, t)$:

$$V(0) = \sum_{a,t} c(a,t)B(a,t).$$  \hfill (46)

Since the initial condition must obey Eq. (26), we have

$$c(a,t) = 0, \quad \text{for } a + t > p^{m-1}. \hfill (47)$$

Next, we define the following sets:

$$R_0(V(0)) = \{(a,t) : c(a,t) \neq 0\}$$

$$R_1(V(0)) = \{(a,t) \in R_0 : a + t \leq a' + t' \text{ for all } (a',t') \in R_0\}$$

$$R_2(V(0)) = \{(a,t) \in R_1 : t \leq t' \text{ for all } (a',t') \in R_1\}. \hfill (48)$$

Note that $R_2 \subseteq R_1 \subseteq R_0$, and there is only one element in $R_2$. Examples of $R_0$, $R_1$ and $R_2$ are shown in Fig. 11.

Then, for the weight of the initial condition, we have the following inequality:

$$W(V(0)) \geq W(B(0,t')).$$  \hfill (50)

The proof is presented in appendix ???. As an example, let us consider the following linear decomposition:

$$V(0) = B(2,3) + B(5,3) + B(1,4) + B(0,8).$$
Then, we have
\[ R = \{(2,3), (5,3), (1,4), (0,8)\}, \quad R_1 = \{(2,3), (1,4)\}, \quad R_2 = \{(2,3)\} \]
and
\[ W(V(0)) \geq W(B(0,3)). \]

Now, we prove theorem 4. Consider the time evolution of an initial condition:
\[ V(\tau) = \sum_{a,t} c(c,t)B(a,t + \tau). \quad (51) \]

Let \( R_2(V(0)) = \{(a',t')\} \). Then, we have \( R_2(V(t)) = \{(a',t + t')\} \) for \( t \leq L - 1 - p^{m-1} \). Then, we have
\[ W(V) = \sum_{t=0}^{L-1} W(V(t)) \geq \sum_{t=0}^{L-1-p^{m-1}} W(V(t)) \geq \sum_{t=t'}^{L-1-p^{m-1}+t'} W(B(0,t)). \]

Due to fact \( \text{IV} \) one has
\[ \sum_{t=t'}^{L-1-p^{m-1}+t'} W(B(0,t)) \geq \sum_{t=0}^{L-p^{m-1}-1} W(B(0,t)) = \left( \frac{(p-1)p(p+1)}{6} \right) \cdot \left( \frac{p(p+1)(p+2)}{6} \right)^{m-1} = \frac{p-1}{p+2} \cdot L_p^{D(p)}. \]

This completes the proof of theorem \( \text{IV} \).

E. Higher-dimensional fractal code

Finally, we briefly discuss the \( D \)-dimensional fractal codes for \( D > 3 \). Most of proofs are skipped since they are complicated, but straightforward to obtain. We consider a \( D \)-dimensional hypercubic lattice with \( n = L \times \cdots \times L \) spins with \( L = p^m \). Each spin is labeled by “time” \( t \) and “positions” \( r^{(1)}, \ldots, r^{(D-1)} \), and we set periodic boundary conditions on all the \( D-1 \)-dimensional surfaces which are parallel to the time axis. The admissible spin configurations of the lattice obeys the following constraint:
\[ x(t+1)_r = x(t)_r + \sum_{j=1}^{D-1} x(t)_{r-e_j} \pmod{p} \quad 0 \leq t \leq L - 2 \quad (52) \]
where \( r = (r^{(1)}, \ldots, r^{(D-1)}) \), and \( e_j \) is a unit vector in the \( r^{(j)} \) direction. In addition, we limit ourselves to spin configurations which satisfy the following initial condition:
\[ x(0)_r = 0 \quad \text{for} \quad \sum_{j=1}^{D-1} r^{(j)} > p^{m-1}, \quad (53) \]
and denote a space of spin configurations specified by the condition above as \( C_p^{(D)} \). Then, we have the following theorem.
Theorem 5 (Higher-dimensional fractal code). For the codeword space \( C_p(D) \), let \( k \) be the number of encodable spins and \( d \) be the code distance of the code. Then, we have
\[
k = \frac{(p^{m-1} + 1) \cdots (p^{m-1} + D - 1)}{(D - 1)!}, \quad \left( \frac{p - 1}{p + D - 1} \right) \cdot L_p^{D(D)} \leq d \leq L_p^{D(D)}
\]
where \( L = p^m \). Thus, we have
\[
k \sim O(L^{D-1}), \quad d \sim O(L_p^{D(D)}) \quad \text{where} \quad D_p^{(D)} = \log \left( \frac{p(p + 1) \cdots (p + D - 1)}{D!} \right) / \log(p).
\]
as \( L \) becomes large.

The fractal dimension \( D_p^{(D)} \) goes to \( D; D_p^{(D)} \to D \) for \( p \to \infty \). Therefore, the code saturates the bound \( kd^{1/D} \leq O(n) \) in Eq. (2) asymptotically for arbitrary \( D \).

Here, we only give a sketch of the proof. We begin by defining \((D - 1)\)-dimensional principal tensors. Recall that we have defined two-dimensional principal matrices from one-dimensional principal vectors by using one-dimensional principal vectors as initial conditions. Here, \((D - 1)\)-dimensional principal tensors can be constructed recursively from \((D - 2)\)-dimensional principal tensors. In particular, a \((D - 1)\)-dimensional principal tensor \( B(r, t) \) with \((D - 2)\)-dimensional vector \( r \) is defined as the time evolution of \( B(r, 0) \):
\[
B(r, 0) = \begin{bmatrix}
B(r) \\
0 \\
\vdots \\
0
\end{bmatrix}
\]
where \( B(r) \) is a \((D - 2)\)-dimensional principal tensor.

With these independent \((D - 1)\)-dimensional principal tensors \( B(r, t) \), one can decompose an arbitrary \((D - 1)\)-dimensional tensor uniquely. Now, we need to obtain an inequality to bound the weight of \((D - 1)\)-dimensional tensors. For the decomposition
\[
V(0) = \sum_{r, t} c(r, t) B(r, t) \quad \text{where} \quad c(r, t) = 0 \quad \text{for all} \quad t + \sum_{j=1}^{D-2} r_j \geq L,
\]
we define the following sets with \( R_D-1 \subseteq \cdots \subseteq R_1 \subseteq R_0 \):
\[
R_0 = \{(r_1, \cdots, r_{D-1}) : c(r_1, \cdots, r_{D-1}) \neq 0\}
\]
\[
R_a = \{(r_1, \cdots, r_{D-1}) \in R_{a-1} : \sum_{j=a}^{D-1} r_j \leq \sum_{j=a}^{D-1} r_j' \quad \text{for all} \quad (r_1', \cdots, r_{D-1}') \in R_{a-1}\}.
\]

Then, assuming that \((r_1', \cdots, r_{D-1}') \in R_{D-1} \), we have
\[
W(V(0)) \geq W(B(0, r_{D-1}'))
\]
This bound can be proven by using the machineries which are very similar to the proof techniques used for lemma 8.

As a result of this bound, one can easily obtain a lower bound for the weight of arbitrary spin configurations arising in \( D \)-dimensional fractal codes.

IV. OPEN QUESTIONS

In this paper, we have constructed local codes, called fractal codes, which asymptotically saturate the theoretical upper limit on the information storage capacity of discrete spin systems. These fractal codes are essentially the best
physically realizable error-correcting codes, and may be actually manufactured in real physical systems in the future when necessary technologies become available.

An immediate future problem concerns whether there exists a local code which exactly saturate the bound or not. Below, we present other possible future problems as well as some comments on implications of our results.

1. Time evolutions of arbitrary cellular automaton, based on linear update rules, can be physically realized as ground states of local Hamiltonians. It would be interesting to find other classes of local codes with fractal properties and analyze their coding properties.

2. Currently, a connection between the bound in Eq. (2) for discrete spin systems and the Bekenstein bound for continuum systems has not been established yet. Since two bounds address similar questions from different aspects, it may be possible to develop a theoretical framework to discuss both bounds on the same footing. Note that fractal codes can be represented as spin network states [7] in their generalized interpretation.

3. While our discussion in this paper is limited to classical error-correcting codes, finding a quantum analog of fractal codes may be an interesting future problem. Note that the bound on the information storage capacity for local quantum codes was given in [5]. Perhaps, a construction similar to our fractal codes may give local quantum codes which asymptotically saturate the bound.

4. Experimental realization of fractal codes is an important future problem. The biggest obstacle in physically realizing fractal codes is the fact that it involves three-body interactions which may not exist in natural physical systems. Yet, it may be possible to simulate three-body terms through two-body terms perturbatively [12] or non-perturbatively [13].

5. Finding local codes supported by frustration-free local Hamiltonians is fundamentally akin to searching for novel quantum phases since frustration-free Hamiltonians can be used as candidates of quantum phases with which one can classify physical phases [14]. Therefore, a problem of finding physically realizable codes, which is of practical importance in coding theory community, is also of fundamental importance in condensed matter physics community.

6. Recently, the author has argued that all the frustration-free systems with continuous scale symmetries can be described effectively by topological quantum field theory via a complete classification of two and three-dimensional stabilizer codes with continuous scale symmetries [15]. However, it seems that frustration-free systems with discrete scale symmetries, such as fractal codes discussed in this paper, are beyond descriptions of topological quantum field theory. It would be interesting to search for effective theories for systems with discrete scale symmetries.

Acknowledgments

I thank Eddie Farhi and Peter Shor for support. I thank Sergey Bravyi for sharing his insights on decoding problems with local Hamiltonians. This work is supported in part by DOE Grant No. DE-FG02-05ER41360 and by Nakajima Foundation.

Appendix A: Proofs of some lemmas

In this appendix, we give proofs of some lemmas used in the main discussion.
### 1. Self-similarity

First, it is worth observing basic properties of the Pascal matrix $B$. As seen in Fig. 4, the Pascal matrices $B$ have fractal properties with self-similar structures. In particular, as shown in Fig. 12(a), similar patterns of spin values appear repeatedly at various length scales. Such self-similar structures are summarized as follows:

**Fact 2 (Self-similarity).** *We denote the Pascal matrix for $L = p^m$ as $B^{(m)}$. Then, we have*

$$
B^{(m)} = \begin{bmatrix}
    B^{(m-1)}, & B^{(m-1)}, & \cdots, & B^{(m-1)}, \\
    B^{(m-1)}, & B^{(m-1)}, & \cdots, & B^{(m-1)}, \\
    \vdots & \vdots & \ddots & \vdots \\
    B^{(m-1)}, & B^{(m-1)}, & \cdots, & B^{(m-1)}
\end{bmatrix}.
$$

(A1)

![Fig. 12: (a) An example of a self-similar property for $p = 3$. $B^{(1)}$ appears repeatedly as submatrices of $B^{(2)}$. (b) Self-similar properties at different length scales.](image)

Therefore, $B^{(m-1)}$ appear repeatedly as submatrices of $B^{(m)}$. It is worth looking at an example for $p = 2$. Then, we notice that

$$
B^{(1)} = \begin{bmatrix} 1, & 0 \\ 1, & 1 \end{bmatrix}, \quad B^{(m)} = \begin{bmatrix} B^{(m-1)}, & 0 \\ B^{(m-1)}, & B^{(m-1)} \end{bmatrix}
$$

(A2)

where 0 represents a $2^{m-1} \times 2^{m-1}$ matrix whose entries are all zeros. An example for $p = 3$ is shown in Fig. 12.

### 2. Proof of lemma 3

The proof of lemma 3 consists of several non-trivial steps.

**Inverse matrices:** In order to capture the properties of the Pascal matrix $B$, it is worth finding its inverse matrix. The inverse $B^{-1}$ is related to the original matrix $B$ through the following simple rule:
Lemma 9. We denote the inverse matrix of $B$ as $B^{-1}$:

$$B \cdot B^{-1} = I$$ (A3)

where $I$ is an identity matrix, and the matrix multiplication is computed modulo $p$.

(a) The inverse matrix is given by:

$$B^{-1}(t)_r = B(L - 1 - r)_{L-1-t}$$ (A4)

where $B^{-1}(t)_r$ represents an entry of $B^{-1}$ at $(t,r)$.

(b) The entries of the inverse matrix is given by

$$B^{-1}(t)_r = L_{-1-r}C_{L-1-t} = (-1)^{t+r}tC_r = (-1)^{t+r}B(t)_r.$$ (A5)

![FIG. 13: Examples of the Pascal matrices and its inverse matrices. Only the shaded regions with $t+r = \text{odd}$ may have different entries in $B$ and $B^{-1}$. The inverse matrices can be obtained by reflecting the original matrices along the arrows shown above. (a) $p = 3$ and $m = 2$. (b) $p = 7$ and $m = 1$.](image)

Examples of inverse matrices are shown in Fig. 13. Note that the inverse matrix $B^{-1}$ can be obtained by taking a reflection of the original matrix $B$.

Proof. Since $B^{-1}(t)_r = B(L - 1 - r)_{L-1-t}$, we have

$$B^{-1}(t)_r = L_{-1-r}C_{L-1-t} = \frac{(p^m - t) \cdots (p^m - 1 - r)}{(t-r)!}$$

$$= \frac{(-t)(-t+1) \cdots (-1 - r)}{(t-r)!}$$

$$= \frac{(-1)^{t+r}(r+1) \cdots (t-1)t}{(t-r)!}$$

$$= \frac{(-1)^{t+r}t!}{r!(t-r)!}$$

$$= (-1)^{t+r}tC_r$$
for \( t \geq r \) where all the calculations are carried out modulo \( p \). It is straightforward to see that \( B \cdot B^{-1} = I \) with some calculations.

The inverse matrix can be also obtained by taking the powers of the original matrix:

**Lemma 10.** Consider spin configuration matrices generated by the following modified rule

\[
x(t + 1)_{r} = x(t)_{r-1} + cx(t)_{r} \quad (\text{mod } p) \quad 0 \leq t \leq L - 2
\]

where \( c \) is a positive integer with an initial condition

\[
x(0)_{0} = 1, \quad x(0)_{r} = 0 \quad (r \neq 0).
\]

We denote such spin configuration matrices as \( R^{(c)} \). Then, we have

\[
B^c = R^{(c)}.
\]

Therefore, by modifying the local rule for spin configurations, one can obtain powers of the Pascal matrix easily. An example is shown in Fig. 14. Here, we notice that \( R^{(p)} = I \) since the update rule is reduced to

\[
x(t + 1)_{r} = x(t)_{r-1} \quad (\text{mod } p) \quad 0 \leq t \leq L - 2.
\]

Also, we notice that \( R^{(p-1)} = B^{-1} \).

**Proof.** For simplicity of discussion, we only prove \( B^2 = R^{(2)} \). Since

\[
B(t)_{r} = B(t - 1)_{r} + B(t - 1)_{r-1},
\]

\[
\begin{array}{cccc}
\hline
    & B & B^2 & B^3 \\
1 0 0 0 & 1 0 0 0 & 1 0 0 0 & 1 0 0 0 \\
1 1 0 0 & 2 1 0 0 & 3 1 0 0 & 3 1 0 0 \\
1 2 1 0 & 4 4 1 0 & 6 6 1 0 & 6 6 1 0 \\
1 3 3 1 & 3 2 1 1 & 5 5 1 1 & 5 5 1 1 \\
1 4 4 2 & 1 2 4 3 & 1 3 4 2 & 1 3 4 2 \\
\hline
\end{array}
\]

\[
\begin{array}{cccc}
& B^4 & B^5 \\
1 0 0 0 & 1 0 0 0 & 1 0 0 0 & 1 0 0 0 \\
4 1 0 0 & 0 1 0 0 & 0 1 0 0 & 0 1 0 0 \\
1 3 3 1 & 0 0 1 0 & 0 0 1 0 & 0 0 1 0 \\
4 3 2 1 & 0 0 0 1 & 0 0 0 1 & 0 0 0 1 \\
1 1 1 1 & 0 0 0 1 & 0 0 0 1 & 0 0 0 1 \\
\hline
\end{array}
\]

**FIG. 14:** Examples of powers of the Pascal matrix \( B \) for \( p = 5 \).
we have
\[
B^2(t)_r = \sum_a B(t)_a B(a)_r
= \sum_a (B(t - 1)_a + B(t - 1)_{a-1}) B(a)_r
= B^2(t - 1)_r + \sum_a B(t - 1)_{a-1} B(a)_r
= B^2(t - 1)_r + \sum_a B(t - 1)_a B(a + 1)_r
= B^2(t - 1)_r + \sum_a (B(t - 1)_a B(a)_r + B(a)_r-1)
= B^2(t - 1)_r + B^2(t - 1)_r + B^2(t - 1)_r-1
\]

where \(B(t)_a = 0\) for \(t < 0\), and \(B(t)_a = B(t)_{a+L}\) if \(a < 0\). Then, we notice that entries of \(B^2\) obeys the rule in Eq. (25) for \(c = 2\). Therefore, \(B^2\) can be generated from the rule for \(c = 2\). In a similar way, one can prove \(B^c = R^{(c)}\).

**Submatrices of the Pascal matrix:** The Pascal matrix \(B\) is invertible since principal vectors \(B(t)\) are pairwise independent. Similar properties holds for submatrices of \(B\). We denote the Pascal matrix \(B\) for \(m = 1\) as \(B^{(1)}\):

\[
B^{(1)} = \begin{bmatrix}
0C_0, & 0C_1, & 0C_2, & \cdots & 0C_{p-1} \\
1C_0, & 1C_1, & 1C_2, & \cdots & 1C_{p-1} \\
\vdots & \vdots & \vdots & \ddots & \vdots \\
p-1C_0, & p-1C_1, & p-1C_2, & \cdots & p-1C_{p-1}
\end{bmatrix}
\]

which is a \(p \times p\) matrix. Then, for submatrices of \(B^{(1)}\), we have the following lemma:

**Lemma 11.** Consider the following submatrix of \(B^{(1)}\), denoted as \(A\):

\[
A = \begin{bmatrix}
x_0C_0, & x_0C_1, & x_0C_2, & \cdots & x_0C_a \\
x_1C_0, & x_1C_1, & x_1C_2, & \cdots & x_1C_a \\
\vdots & \vdots & \vdots & \ddots & \vdots \\
x_aC_0, & x_aC_1, & x_aC_2, & \cdots & x_aC_a
\end{bmatrix}
\]

where \(a < p\) and \(0 \leq x_0 < x_1 < \cdots < x_a < p\). Then, \(A\) always has an inverse matrix \(A^{-1}\). Similarly, consider the following submatrix of \(B^{(1)}\), denoted as \(A'\):

\[
A' = \begin{bmatrix}
p-a-1C_{y_0}, & p-a-1C_{y_1}, & p-a-1C_{y_2}, & \cdots & p-a-1C_{y_a} \\
p-y_0C_{y_0}, & p-y_0C_{y_1}, & p-y_0C_{y_2}, & \cdots & p-y_0C_{y_a} \\
p-y_1C_{y_0}, & p-y_1C_{y_1}, & p-y_1C_{y_2}, & \cdots & p-y_1C_{y_a} \\
p-y_aC_{y_0}, & p-y_aC_{y_1}, & p-y_aC_{y_2}, & \cdots & p-y_aC_{y_a}
\end{bmatrix}
\]

where \(0 \leq y_0 < y_1 < \cdots < y_a < p\). Then, \(A'\) always has an inverse matrix \((A')^{-1}\).

The construction of \(A\) goes as follows. First, we choose an \(p \times a\) submatrix from \(B^{(1)}\) on the left hand side of \(B^{(1)}\) \((a < p)\). Then, we pick up \(a\) rows to create an \(a \times a\) matrix \(A\). Similarly, to construct \(A'\), we choose an \(a \times p\) submatrix on the bottom of \(B^{(1)}\), and pick up \(a\) columns to create an \(a \times a\) matrix \(A'\). Examples of such constructions of submatrices are shown in Fig. 15.
It is worth looking at examples. Consider the case where $p = 5$. Then, we have

\[
B^{(1)} = \begin{bmatrix}
1, 0, 0, 0, 0 \\
1, 1, 0, 0, 0 \\
1, 2, 1, 0, 0 \\
1, 3, 3, 1, 0 \\
1, 4, 6, 4, 1 \\
1, 5, 3, 3, 5 \\
1, 6, 2, 6, 2, 6 \\
\end{bmatrix}.
\]  

(A13)

For $a = 2$ and $x_0 = 2$, $x_1 = 3$ and $x_2 = 4$, we have

\[
A = \begin{bmatrix}
1, 2, 1 \\
1, 3, 3 \\
1, 4, 1 \\
\end{bmatrix}.
\]  

(A14)

Since three vectors $(1, 2, 1)$, $(1, 3, 3)$ and $(1, 4, 1)$ are independent (mod 5), $A$ is invertible. For $a = 2$ and $y_0 = 1$, $y_1 = 2$ and $y_2 = 3$, we have

\[
A' = \begin{bmatrix}
2, 1, 0 \\
3, 3, 1 \\
4, 1, 4 \\
\end{bmatrix}
\]  

(A15)

which is also invertible.

**Proof.** For simplicity of discussion, we present a proof only for $A$. First, recall that the Vandermonde matrix $M$ has the following well-known property:

\[
M = \begin{bmatrix}
1, x_0, x_0^2, \cdots, x_0^a \\
1, x_1, x_1^2, \cdots, x_1^a \\
\vdots & \vdots & \ddots & \ddots \\
1, x_a, x_a^2, \cdots, x_a^a \\
\end{bmatrix}, \quad \det(M) = \prod_{0 \leq i < j \leq a} (x_j - x_i).
\]  

(A16)

Therefore, the following vectors are independent when $x_i \neq x_j$ for all $i$ and $j$:

\[
(1, 1, \cdots, 1), (x_0, x_1, \cdots, x_a) \cdots (x_0^a, x_1^a, \cdots, x_a^a).
\]  

(A17)

Now, let us consider the Vandermonde matrix modulo $p$. When $0 \leq x_0 < x_1 < \cdots < x_a < p$, the vectors above are independent modulo $p$ since the determinant of $M$ computed modulo $p$ is nonzero. Notice that our goal to prove that the following vectors are independent modulo $p$:

\[
(x_0 C_0, x_1 C_0, \cdots, x_a C_0), (x_0 C_1, x_1 C_1, \cdots, x_a C_1) \cdots (x_0 C_a, x_1 C_a, \cdots, x_a C_a).
\]  

(A18)
This is immediate since vectors in Eq. (A18) can be created by adding vectors in Eq. (A17). Therefore, \( A \) is invertible. A similar proof works for \( A' \) by using lemma 9.

**Proof of lemma 3** Finally, we prove lemma 3. Due to the self-similar structures of the Pascal matrix \( B \), it is sufficient to prove the lemma for \( m = 1 \). Consider a decomposition

\[
V(0) = \sum_t c(t)B(t)
\]

where \( t_{\text{min}} \) is the minimal integer such that \( c(t) \neq 0 \). Then, the goal is to prove

\[
W(V(0)) \geq W(B(t_{\text{min}})).
\]

We list all the integers \( r \) such that

\[
B(t_{\text{min}})_r \neq 0 \quad \text{and} \quad V(0)_r = 0
\]

and denote them as \( r_1, \ldots, r_a \). Then, the number of non-zero entries of \( V(0)_r \) for \( r \leq t_{\text{min}} \) is \( W(B(t_{\text{min}})) - a \). Next, we list all the integers \( r \) such that

\[
B(t_{\text{min}})_r = 0 \quad \text{and} \quad V(0)_r = 0
\]

and denote them as \( r'_1, \ldots, r'_b \). Then, the number of non-zero entries of \( V(0)_r \) for \( t_{\text{min}} < r \) is \( (p-1 - t_{\text{min}}) - b \). Then, we have

\[
W(V(0)) = W(B(t_{\text{min}})) - a + (p-1 - t_{\text{min}}) - b
\]

from a simple counting argument. Therefore, it suffices to prove that \( a + b \leq p - 1 - t_{\text{min}} \).

We next consider constrains on coefficients \( c(t) \):

\[
V(0)_r = \sum_{t=0}^{p-1} c(t)B(t)_r = 0 \quad \text{for} \quad r = r_1, \ldots, r_a, r'_1, \ldots, r'_b.
\]

Recall that \( t_{\text{min}} \) is the minimal \( t \) such that \( c(t) \neq 0 \) and \( c(t) = 0 \) for \( t < t_{\text{min}} \). Then, the above constraints can be concisely represented as follows:

\[
(A')^T \cdot \begin{bmatrix} c(t_{\text{min}}) \\ c(t_{\text{min}} + 1) \\ \vdots \\ c(p-1) \end{bmatrix} = 0
\]

where

\[
A' = \begin{bmatrix}
B(t_{\text{min}})_{r_1}, & \cdots, & B(t_{\text{min}})_{r_a}, & B(t_{\text{min}})_{r'_1}, & \cdots, & B(t_{\text{min}})_{r'_b} \\
B(t_{\text{min}} + 1)_{r_1}, & \cdots, & B(t_{\text{min}} + 1)_{r_a}, & B(t_{\text{min}} + 1)_{r'_1}, & \cdots, & B(t_{\text{min}} + 1)_{r'_b} \\
\vdots & \cdots & \vdots & \cdots & \cdots & \vdots \\
B(p-1)_{r_1}, & \cdots, & B(p-1)_{r_a}, & B(p-1)_{r'_1}, & \cdots, & B(p-1)_{r'_b}
\end{bmatrix}
\]

Here, \( A' \) is a \((p-t_{\text{min}}) \times (a+b)\) matrix. Notice that the rank of \( A' \) is \( p-t_{\text{min}} \) when \( a+b \geq p-t_{\text{min}} \) due to lemma 11. In such cases, we have \( c(t) = 0 \) for all \( t \) which leads to a contradiction. Therefore, \( a+b < p-t_{\text{min}} \). This leads to \( W(V(0)) \geq W(B_{\text{min}}) \), and completes the proof of lemma 3.
3. Proof of lemma 8

We begin by proving the following lemma.

Lemma 12. We have

\[
W(V(0)) \geq W \left( \sum_{(a, t) \in R_1(V(0))} c(a, t)B(a, t) \right). \tag{A19}
\]

Proof. We decompose \( V(0) \) as follows:

\[
V(0) = \sum_{a + t = 0} c(a, t)B(a, t) + \sum_{a + t = 1} c(a, t)B(a, t) + \sum_{a + t = 2} c(a, t)B(a, t) + \cdots. \tag{A20}
\]

In particular, we set

\[
V(0) = \sum_{\delta} V(0)^{(\delta)}, \quad V(0)^{(\delta)} = \sum_{a + t = \delta} c(a, t)B(a, t). \tag{A21}
\]

For simplicity of discussion, we consider the case where

\[
V(0) = V(0)^{(\delta)} + V(0)^{(\delta+1)}.
\]

\( V(0)^{(\delta)} \) can be represented as follows:

\[
V(0)^{(\delta+1)} = \begin{bmatrix}
    d(\delta) \cdot B(\delta) \\
    d(\delta - 1) \cdot B(\delta - 1) \\
    \vdots \\
    d(0) \cdot B(0) \\
    0 \\
    \vdots
\end{bmatrix}
\]

where \( d(0), \ldots, d(\delta) \) are some integers, and \( V(0)^{(\delta+1)} \) can be represented as follows:

\[
V(0)^{(\delta+1)} = \begin{bmatrix}
    d'(\delta + 1) \cdot B(\delta + 1) \\
    d'(\delta) \cdot B(\delta) \\
    \vdots \\
    d'(0) \cdot B(0) \\
    0 \\
    \vdots
\end{bmatrix}
\]

where \( d'(0), \ldots, d'(\delta + 1) \) are some integers. Then, we have

\[
V(0)^{(\delta)} + V(0)^{(\delta+1)} = \begin{bmatrix}
    d'(\delta + 1) \cdot B(\delta + 1) + d(\delta) \cdot B(\delta) \\
    d'(\delta) \cdot B(\delta) + d(\delta - 1) \cdot B(\delta - 1) \\
    \vdots \\
    d'(1) \cdot B(1) + d(0) \cdot B(0) \\
    d'(0) \cdot B(0) \\
    \vdots
\end{bmatrix}
\]
Then, due to lemma 3, we have

\[ W(V(0)) \geq W(V(0)^{(4)}). \]

The discussion above can be easily extended to general cases. This completes the proof.

Next, it is convenient to consider the transposes of principal matrices. Let us begin with an example. When \( V(0) = B(1, 1) \) for \( p = 2 \) and \( m = 2 \), we have

\[
B(1, 1) = \begin{bmatrix}
B(2) \\
B(1) \\
0 \\
0
\end{bmatrix} = \begin{bmatrix}
1, 0, 1, 0 \\
1, 1, 0, 0 \\
0, 0, 0, 0 \\
0, 0, 0, 0
\end{bmatrix}.
\]

Then, its transpose is

\[
B(1, 1)^T = \begin{bmatrix}
1, 1, 0, 0 \\
0, 1, 0, 0 \\
1, 0, 0, 0 \\
0, 0, 0, 0
\end{bmatrix} = \begin{bmatrix}
B(1) \\
B(0) + B(1) \\
B(0) \\
0
\end{bmatrix}.
\]

Here, we apply lemma 12 to \( B(1, 1)^T \):

\[
B(1, 1)^T = (B(1, 1)^T)^{(1)} + (B(1, 1)^T)^{(2)}
\]

where

\[
(B(1, 1)^T)^{(1)} = \begin{bmatrix}
B(1) \\
B(0) \\
0 \\
0
\end{bmatrix} = B(0, 1), \quad (B(1, 1)^T)^{(2)} = \begin{bmatrix}
0 \\
B(1) \\
B(0) \\
0
\end{bmatrix},
\]

and

\[
W(V(0)) \geq W((B(1, 1)^T)^{(1)}) = W(B(0, 1)).
\]

As the example above shows, by considering the transpose of principal matrices, one can further lower bound on the weight of principal matrices. For this purpose, the following lemma is particularly useful.

**Lemma 13.** Consider the decomposition of \( B(a, t)^T \) with \( a + t < L \):

\[
B(a, t)^T = \sum_{a', t'} c(a', t') B(a', t'). \tag{A22}
\]

Then, we have

\[
R_1(B(a, t)^T) = \{(0, t)\}. \tag{A23}
\]

The proof of the lemma 13 is immediate by noticing the following fact:

**Fact 3.** The principal matrices \( B(0, t) \) are symmetric under the transpose:

\[
B(0, t)^T = B(0, t). \tag{A24}
\]
The principal matrix $B(a,t)$ for $a \neq 0$ is given by

$$B(a,t) = \sum_x T_{x(t)}^{-1}(B(a)_x \cdot B(0,t)),$$

(A25)

and its transpose is given by

$$B(a,t)^T = \sum_y T_{y(t)}^{-1}(B(a)_y \cdot B(0,t)),$$

(A26)

We finally prove lemma.3 For the decomposition

$$V(0) = \sum_{a,t} c(a,t)B(a,t),$$

we define

$$V(0)^* = \sum_{(a,t) \in R_1(V(0))} c(a,t)B(a,t).$$

Then, we have

$$W(V(0)) \geq W(V(0)^*).$$

Here, we consider the transpose of $V(0)^*$. Then, for $(a',t') \in R_2(V(0))$, we have

$$R_1((V(0)^*)^T) = (0,t').$$

Therefore, we have

$$W(V(0)^*) = W((V(0)^*)^T) \geq W(B(0,t')).$$

This completes the proof.

[1] R. Landauer, Nature 335, 779 (1988)
[2] J. D. Bekenstein, Phys. Rev. D 23, 287 (1981)
[3] S. Hawking, Commun. Math. Phys. 43, 199 (1975)
[4] L. Susskind, J. Math. Phys. 36, 6377 (1995)
[5] S. Bravyi, D. Poulin, and B. Terhal, Phys. Rev. Lett. 104, 050503 (2010)
[6] R. Gallager, IRE Trans. Inform. Theory 8, 21 (1962)
[7] R. Penrose, in Quantum Theory and Beyond (Cambridge University Press, 1971)
[8] C. Rovelli and L. Smolin, Phys. Rev. D 52, 5743 (1995)
[9] M. E. J. Newman and C. Moore, Phys. Rev. E 60, 5068 (1999)
[10] D. R. Chowdhury, S. Basu, I. S. Gupta, and P. P. Chaudhuri, IEEE Transactions on Computers 43, 759 (1994)
[11] S. Wolfram, A new kind of science (Wolfram Media Inc. Champaign, 2002)
[12] S. P. Jordan and E. Farhi, Phys. Rev. A 77, 062329 (2008)
[13] S. A. Ocko and B. Yoshida, quant-ph/1107.2697 (2011)
[14] B. Yoshida, Ann. Phys. 326, 15 (2011)
[15] B. Yoshida, Ann. Phys. 326, 2566 (2011)