Pseudorandom Cayley Graphs with Sub-random Kolmogorov Complexity: Properties and Applications

...

Written by Svetlana Pack
Advised by Professor Alex Iosevich
University of Rochester

May 10 2024

This thesis was written to fulfill the upper-level writing requirement for a Honors degree in Mathematics at the University of Rochester.

1 Acknowledgements

I would first and foremost like to thank my advisor, Professor Alex Iosevich, for his deeply caring mentorship over the years. His taking a chance on me when I was a freshman is the reason I want to be a research mathematician now, and his unfaltering support and trust in me even when I didn't feel sure of myself allowed me to grow in ways I never expected. I hope we get to collaborate on many more projects in the future.

I would also like to thank my fellow student researchers from the TRIPODS 2023 Summer Research Geometric Complexity Project, Anuraag Kumar, Joshua Iosevich, and Alhussein Khalil, for helping lay the groundwork for my thesis research through the empirical "proof of concept" simulations. I would likewise like to thank the project leaders, Professor Iosevich and Professor Azita Mayeli, for their guidance. Additional thanks to Anuraag Kumar for some lovely and quite insightful conversations about Cayley graphs.

I would like to thank Professor Kaave Hosseini for introducing me to the utterly fascinating world of spectral graph theory and pseudorandomness. I would also like to thank my thesis committee members, Professors Jon Pakianathan and Juan Rivera-Letelier. Many of the classes I have taken at the University of Rochester that I count as most influential on my current research interests were with them. In particular, the contents of this thesis are in large part an amalgamation of ideas these three professors exposed me to. I am really grateful for the opportunity to have learned with such brilliant instructors.

I would like to thank my father for the advice on technical writing and proofreading of early versions of this manuscript, as well as for many wonderful conversations about math.

I would like to thank my partner, family, and friends for the unending well of moral support and patience in listening to me pontificate about graphs, flail around in excitement over graphs, and pace around the kitchen at 1AM thinking about graphs. It's really a privilege to be surrounded by people I feel I can be completely myself around in that way.

Finally, I would like to thank my cat Willow. Her periodically flopping on my scratch work reminded me to take breaks and thereby retain my sanity during the process of assembling the manuscript you are now reading.

2 Abstract

The concept of 'complexity' the amount of information you need to completely specify an object, is of considerable interest in a world where computational resources are at an ever-increasing premium. Several competing ideas have emerged in the last century to formalize this notion. Among them is Kolmogorov complexity or algorithmic entropy, the length of the shortest program describing an object. This can be used to describe the degree of 'randomness' exhibited by a single realization of said object.

Pseudorandom graphs, which have similar connective and spectral properties to random graphs of the same edge density, can be used to reduce the number of bits of randomness used in algorithms. One example of a family of pseudorandom graphs are Paley graphs, the Cayley graph on the integers modulo n under addition generated by quadratic residues modulo n. Cayley graphs have algebraic properties that make them relatively easy to construct, but can give rise to pseudorandom behavior such as that demonstrated by Paley graphs. This is especially remarkable given that Cayley graphs on the cyclic group such as Paley graphs are circulent, and thus exceedingly symmetric.

Although not computable, the Kolmogorov complexity of an object can be bounded above using lossless compression algorithms. Thus, proving the Kolmogorov complexity of an object has Kolmogorov complexity lower than that expected of a random string of the same length could be an indicator as to whether a pseudorandom object is helpful for reducing the randomness of an algorithm. We prove that additive Cayley graphs on the integers have sub-random Kolmogorov complexity using the Lempel-Ziv 1976 algorithm. We then briefly discuss the efficacy of Cayley graphs in reducing the randomness used by Echo State Recurrent Neural Networks, which typically rely on a random or random-like graph architecture to effectively forecast chaotic time series data.

3 Introduction

One of the most contentious philosophical kerfuffles of the last century is that over the idea of 'randomness'; what does it mean for an object to be truly random? Is such a thing even possible? Likewise, what does it mean for something to be 'complex'?

These questions are not entirely navel-gazey philosophical dalliance. Characterizing the complexity of objects such as networks is of great interest for a myriad of applications, from simulating gene expression and interaction, to detecting epileptic seizures in electroencephalogram data [1].

In 1948, Claude Shannon catalysed the information theory revolution by introducing "information entropy" as a measure of the number of bits needed to specify an element from a discrete probability distribution. This allowed for people to talk about the information capacity and complexity of probability distributions in a mathematically rigorous way. About a decade later, Kolmogorov (concurrently with but independently of Chaitin and Solomonoff) defined of 'algorithmic entropy', using Turing's theory of computation to adapt the concept of entropy to individual realizations of data instead of larger statistical ensembles.

Alternatively called 'Kolmogorov Complexity' or 'Chaitin-Kolmogorov-Solomonoff complexity', the algorithmic entropy of an object can be intuitively thought of as the minimum length of a computer program needed to completely specify that object. For instance, a sequence of n bits generated from a Bernoulli distribution with probability parameter $p = \frac{1}{2}$ would have $O(n)$ algorithmic entropy in that it would take n bits to uniquely specify the sequence. Meanwhile, a sequence of n bits all equal to one would (on a machine that already knows the value of 'n') would have $O(1)$ algorithmic entropy as the program 'repeat the value '1' n times', which does not scale with n. (Note: if the machine was not already provided with the parameter n then we would need to encode the value of n, taking $O(log(n))$ bits and thus significantly increasing the complexity of our program). This is quite consistent with the $p = \frac{1}{2}$ Bernoulli distribution underlying our first sequence having maximal Shannon entropy compared to all other parameter p Bernoulli distributions, while constant sequences have zero entropy.

Algorithmic entropy has both pronounced strengths and weaknesses as a measure of an object's information content. On one hand, a result termed the Invariance Theorem establishes that - unlike alternate measures of entropy- algorithmic entropy is invariant under object representation up to constant overhead[2]. On the other, algorithmic entropy is cumbersome to work with in that it is not explicitly computable [9]. However, algorithmic complexity is upper semicomputable, and we can use lossless compression algorithms on data to derive upper bounds for their algorithmic entropy.

4 Preliminaries

In this section we summarize the notation and basic graph theory that we will use to define algorithmic entropy and pseudorandomness. We also introduce Cayley graphs, one of the main focuses of this paper.

4.1 Big O and little o notation

Let f, g be function $f : \mathbb{R}_+ \to \mathbb{R}_+$. Then if there exists $x_o \in \mathbb{R}_+, M \in \mathbb{R}_+$ such that $f(x) \leq Mg(x) \quad \forall x_o \leq x$ then we write $f(x) = O(g(x))$.

If $\forall \epsilon > 0$ there exists $x_o > 0$ such that $f(x) \leq \epsilon g(x) \quad \forall x \geq x_o$ then we write $f(X) = o(g(x))$. Note this is equivalent to $\lim_{n \to \infty} \frac{f(x)}{g(x)} = 0$

4.2 Graph Notation

Let $G = (V, E)$ denote a graph on vertex set V with $E \subseteq V \times V$.

We will restrict our attention to simple graphs, ie. graphs with no more than one edge between vertices, and graphs without self-loops. Then, all graphs satisfying $|V| = n$ G will have an $n \times n$ adjacency matrix representation A_G below, with $[A_G]_{ii} = 0 \ \forall i \in [n]$:

$$[A_G]_{ij} = \begin{cases} 1 \text{ if } (v_i, v_j) \in E \\ 0 \text{ if not} \end{cases}$$

Let $d : V \to [|V|]$ denote the degree function of graph $G = (V, E)$ defined $d(v_i) = |\{(v_i, v_j) \in E : v_j \in V\}| \ \forall v_i \in V$. Then, $G = (V, E)$ with $|V| = n$ has degree list representation $D_G = \{d(v_1), ..., d(v_n)\}$. When $d(v_i) = d \forall i \in [|V|]$ we call a graph G d-regular.

We assume G is an undirected graph and thus A_G symmetric unless explicitly specified otherwise. Then, by spectral theorem A_G has an orthogonal eigenvector basis $\{v_1, ..., v_n\}$ corresponding to eigenvalues $\{\lambda_1, \lambda_2, ..., \lambda_n\}$, listed in descending order.

Remark 4.1 (Trivial Eigenvalue) *Let A_G be the adjacency matrix of d regular undirected graph, and $u = (1, ..., 1)$ denote the $n \times 1$ all 1s vector.*

Observe $[A_G u]_i = \sum_{j=1}^{n} A_{ij} \times 1 = d \times 1 = \forall i \in [n]$, so $A_G u = du$. So, u is an eigenvector of A_G, and because all entries of A_G are in $\{0, 1\}$ it is clear the corresponding eigenvalue d is the largest possible eigenvalue.

Then, because $v_1 = u$ and $\lambda_1 = d$ for all d regular undirected graphs, we term λ_1 the trivial eigenvalue.

4.3 Graph Properties

Here we define properties of graphs and families of graphs, sequences of graphs $\{G_n = (V_n, E_n)\}$ that unless otherwise specified will have $|V_n| = n \quad \forall n \in \mathbb{Z}_+$, that will be referenced throughout this paper:

Definition 4.1 ((n, d, λ) graph) *A graph $G = (V, E)$ is called (n, d, λ) if it has degree d with $|V| = n$ and has and the greatest absolute value of its nontrivial eigenvalues is λ, i.e., $\lambda = max_{i \in \{2,3,...,n\}} |\lambda_i| = max(\lambda_2, |\lambda_n|)$. We refer to λ as the spectral gap pf G.*

Definition 4.2 (Dense Family of Graphs) *We call a family of simple undirected graphs $\{G_n = (V_n, E_n)\}$ dense if there exists a constant $\rho \in (0, 1)$ such that $\forall n \in \mathbb{Z}_+$ we have that $|E_n| = (\rho + o(1))\binom{n}{2}$. We say that $\{G_n\}$ has constant order edge density ρ.*

Definition 4.3 (Sparse Family of Graphs) *We call a family of simple undirected graphs $\{G_n = (V_n, E_n)\}$ sparse if we have $\lim_{n \to +\infty} \frac{|E_n|}{\binom{n}{2}} = 0$.*

4.4 Cayley Graphs

Cayley graphs are families of graphs constructed from algebraic groups. This imbues Cayley Graphs with algebraic structure that, as we shall see, makes them a rich and easy to work with source of pseudorandom graphs.

Definition 4.4 (Cayley Graphs) *Let S be a generating set of a group on set H with group operation $: H \quad H \; ! \; H$. Then the Cayley graph $C(H,S)$ a graph with vertex set $V = H$ and edge set $E = f(u,v)jv = s \quad u, s \; 2 \; Sg$.*

A useful and well-known property of Cayley graphs on the additive subgroups of the finite fields $(\mathsf{Z}_n, +,)$ is the equivalence of the eigenspectrum of the adjacency matrix A_C of a Cayley graph $C(\mathsf{Z}_n, S)$ and the Fourier spectrum of the indicator function, as well as the eigenvectors of A and the characters in the Fourier basis of Z_p. This property will be of great use in proving the pseudorandomness of Paley graphs.

Lemma 4.1 *Let $C(\mathsf{Z}_p{}^n, S)$ be a Cayley graph on the group $(\mathsf{Z}_p, +)$ for prime $p \; 2 \; \mathsf{Z}$ with generating set $S \quad \mathsf{Z}_p{}^n$ and adjacency matrix A, and let $\chi_S : \mathsf{Z}_n \; ! \; f0, 1g$ be the indicator function of S. Let $f f\lambda_1, \lambda_2, ..., \lambda_n g$ be the eigenspectrum of the adjacency matrix of G (with multiplicity, not necessarily in order of size), and let $v_1, v_2, ..., v_n$ be the associated eigenvectors. Let $f f\omega(x)^k g_{k \, 2 \, f0;:::;p \quad 1gg}$ (where $\omega(x) = e^{i\frac{2\pi x}{p}}$) be the Fourier basis of Z_p and $f\widehat{\chi_S}(k) : k \; 2 \; f0, ..., p \quad 1gg$ are the associated Fourier characters. Then, $\lambda_k = \widehat{\chi_S}(k)$ and the associated eigenvector $v_k = \omega^k : \mathsf{Z}_p \; ! \; \mathsf{R} \; 8k \; 2 \; \mathsf{Z}_p$*

PROOF OF 4.1:
We want to show that for any $k \; 2 \; \mathsf{Z}_p$ we have $\lambda_{k+1}\omega^k = A\omega^k$ Observe that

$$[A\omega^k]_l = \sum_{j=0}^{p \quad 1} A_{lj}\omega(j)^k = \sum_{j=0}^{p \quad 1} \chi_S(l \quad j)(\omega(j)^k)$$

χ_S has Fourier expansion $\chi_S(j) = \frac{1}{j\mathsf{Z}_p j} \sum_{m=0}^{p \quad 1} \widehat{\chi_S}(m)(\omega(j))^m = \frac{1}{p} \sum_{m=0}^{p \quad 1} \widehat{\chi_S}(m)(\omega(j))^m$. Thus,

$$\sum_{j=0}^{p \quad 1} \chi_S(l \quad j)(\omega(j))^k = \sum_{j=0}^{p \quad 1} \frac{1}{p}(\sum_{m=0}^{p \quad 1} \widehat{\chi_S}(m)(\omega(l \quad j))^m)(\omega(j))^k = \sum_{j=0}^{p \quad 1} \frac{1}{p}(\sum_{m=0}^{p \quad 1} \widehat{\chi_S}(m)((\omega(l \quad j))^m(\omega(j))^k))$$

$$= \frac{1}{p}\sum_{j=0}^{p \quad 1}(\sum_{m=0}^{p \quad 1} \widehat{\chi_S}(m)((e^{\frac{i2\pi(l \quad j)m}{p}})(e^{\frac{i2\pi(j)k}{p}})) = \frac{1}{p}\sum_{j=0}^{p \quad 1}(\sum_{m=0}^{p \quad 1} \widehat{\chi_S}(m)(e^{\frac{i2\pi(l+j(k \quad m))}{p}})$$

$$= \frac{1}{p}\sum_{m=0}^{p \quad 1} e^{\frac{i2\pi lm}{p}}(\widehat{\chi_S}(m)\sum_{j=0}^{p \quad 1} e^{\frac{i2\pi(j(k \quad m))}{p}})$$

By orthogonality of the Fourier basis $\sum_{j=0}^{p \quad 1} e^{\frac{i2\pi(j(k \quad m))}{p}} = 0$ when $(k \quad m) \; \cancel{6}= \; 0$, so

$$= \frac{1}{p}\sum_{m=0}^{p \quad 1} e^{\frac{i2\pi lm}{p}}(\widehat{\chi_S}(m)\sum_{j=0}^{p \quad 1} e^{\frac{i2\pi(j(k \quad m))}{p}}) = \frac{1}{p}e^{\frac{i2\pi lk}{p}}(\widehat{\chi_S}(k))\sum_{j=0}^{p \quad 1} 1 = \widehat{\chi_S}(k)e^{\frac{i2\pi lk}{p}} = \widehat{\chi_S}(k)(\omega(l))^k = \widehat{\chi_S}(k)[\omega^k]_l$$

Thus, $[A\omega^k]_l = \widehat{\chi_S}(k)[\omega^k]_l \; 8k \; 2 \; \mathsf{Z}_p$. QED
One widely used example of a Cayley graph on the additive subgroup of the finite field $(\mathsf{Z}_n, +,)$ is the cycle graph.

Definition 4.5 (Cycle graph) *A n vertex cycle graph is the Cayley graph on the additive group $(\mathsf{Z}_n, +)$ with generating set $f \quad 1g$. The cycle graph on $(\mathsf{Z}_{13}, +)$ is depicted in gure 1.*

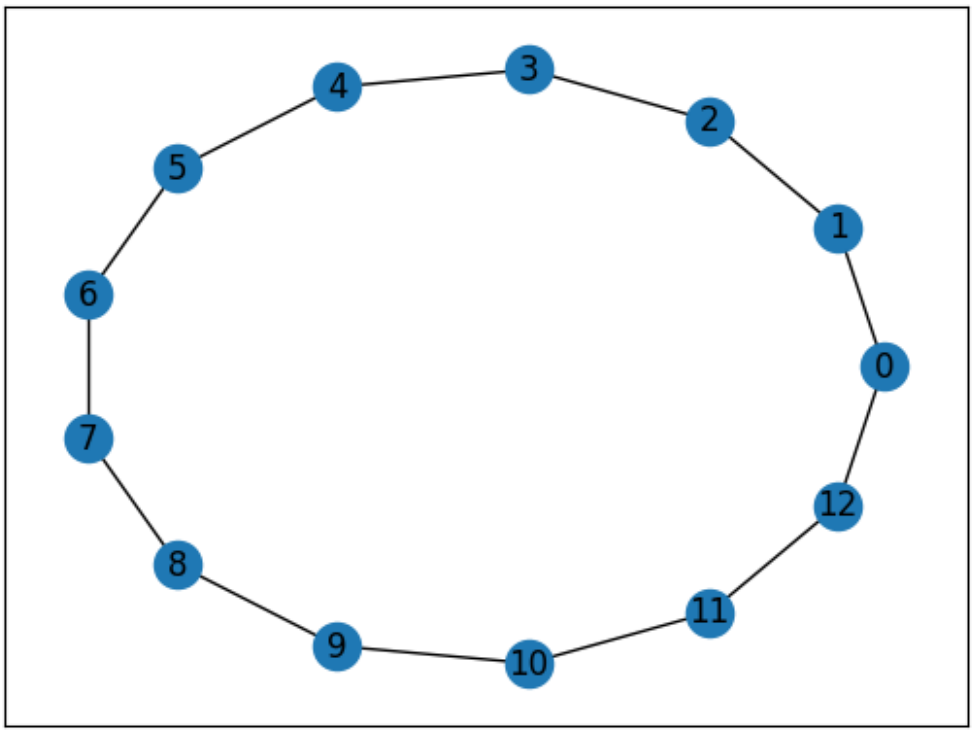

Figure 1: The Cycle graph $C((\mathsf{Z}_{13},+),U=\{1\})$

Using Lemma 4.1 we can calculate the eigenspectrum of a Cayley graph on $(\mathsf{Z}_n,+)$ as $\lambda_k = e^{\frac{i2\pi k}{n}} + e^{\frac{i2\pi k}{n}} = 2cos(\frac{2\pi k}{n}) \quad \forall k \in [n]$. We note the trivial eigenvalue $\lambda_1 = 2cos(\frac{2\pi 0}{n}) = 2(1) = |\{1\}|$, consistent with cycle graphs having two generators and thus being 2-regular.

Remark 4.2 (Cycle Graphs) *We claim Cycle graphs are the sparsest possible connected Cayley graphs on $(\mathsf{Z}_p,+)$ up to isomorphism when p prime, with $d=2$.*

It is trivial to observe that cycle graphs are indeed connected $\forall n \in \mathsf{Z}_+$ as $\{1\}$ generates $(\mathsf{Z}_+,+)$.

We note that the bipartite ladder graph is the Cayley graph on $(\mathsf{Z}_{2n},+)$ for $n \in \mathsf{Z}_+$ with generating set $\{\frac{2n}{2}\} = \{n\} = \{2n-n, n\} = \{n\}$ is the sparsest possible undirected Cayley graph, as d = 1. But the bipartite ladder graph is not connected for $n>1$ as $\{n\}$ has order 2 in $(\mathsf{Z}_{2n},+)$ as $n+n=2n \equiv 0$ and so n does not by itself generate $(\mathsf{Z}_{2n},+)$. Thus the resulting Cayley graph is not connected.

Then, we show that cycle graphs are the sparsest possible undirected Cayley graph on $(\mathsf{Z}_p,+)$ up to isomorphism by showing all undirected $d=2$ Cayley graphs on Z_p are isomorphic to the cycle graph when p prime.

For a Cayley graph on $(\mathsf{Z}_p,+)$ to be undirected and degree 2 we must have the generating set take form $\{a\}$ for some $a \in \mathsf{Z}_p$, as undirected Cayley graphs must have generating sets closed under inverses to ensure edges are symmetric.

Cayley graphs on $(\mathsf{Z}_p,+)$ with p prime and generating set $\{a\}$ for $a \in \mathsf{Z}_p \setminus \{0\}$ are identical to a cycle graph up to isomorphism. This is because when p is prime, $a \not\equiv ka$ in $(\mathsf{Z}_p,+)$ for any $k \in \mathsf{Z}_p \setminus \{0,1\}$, so $\{a\}$ generates $(\mathsf{Z}_p,+) \quad \forall a \in \mathsf{Z}_p \setminus \{0\}$.

Then, if C is the cycle graph of on $(\mathsf{Z}_p,+)$ and G is the Cayley graph of on $(\mathsf{Z}_p,+)$ with p prime and generating set $\{a\}$, we observe that C is isomorphic to G under the vertex relabelling $l:(\mathsf{Z}_p,+) \to (\mathsf{Z}_p,+)$ defined $l(i)=ai$.

Thus we can say that on graphs of prime vertex size, cycle graphs are the sparsest connected Cayley graphs up to isomorphism.

5 Background

5.1 Algorithmic Entropy

The algorithmic entropy of an object, the length of the shortest program describing the object, is formally defined thusly;

Definition 5.1 (Algorithmic Entropy) *Let s be a string of nite length de ned on a nite alphabet, T be a universal Turing machine, and P be any program for which T returns s upon halting. The algorithmic entropy of s is de ned*

$$K_T(s) = min\{|P| : T(P) = s\}$$

The Invariance Theorem is a well known result establishing that the Kolmogorov complexity of a string does not depend on the representation chosen, so we can fix any arbitrary Universal Turing Machine T for our purposes.

5.1.1 Lempel Ziv Complexity

Kolmogorov complexity is not computable, but is upper-semicomputable [7]. One may obtain upper bounds for the Kolmogorov complexity by applying tools such as a lossless compression algorithm to the string under consideration and calculating the size of the resulting compression [9]. One such algorithm we will make use of is the $LZ76$ algorithm to calculate the Lempel Ziv complexity of a string. Intuitively we can think of Lempel-Ziv complexity as measuring the number of unique non- repeating substrings in a string, formally described in the definition below [10]:

Definition 5.2 *Let $s = \{s_i\}_{i \in [n]}$ with $s_i \in \{0,1\} \quad \forall i \in [n]$ be a binary string of length $n \in \mathbb{Z}_+$. Let $s_j{}^k$ denote the substring $\{s_i\}_{i=j}^{k}$ of s starting at index j and ending at index k, for some $1 \quad j \quad k \quad n$*

Use the LZ76 algorithm to recursively partitioning s into some number $p \quad n$ of disjoint substrings, called 'blocks' $\{B_j\}_{j \in [p]}$, each representing the shortest substring that is not contained anywhere in the substring preceding it. The LZ76 Algorithm is formally described in Algorithm 1.

We then de ne the Lempel-Ziv complexity $LZ(s)$ of s as the number $LZ(s) := p = |\{B_j\}_{j \in [p]}|$ [10].

5.1.2 LZ76 Compression Algorithm Example Calculation

To illustrate the LZ76 algorithm, Algorithm 1, we perform an example computation on, say, the indicator function $\chi_U : \mathbb{Z}_{13} \ ! \ \{0,1\}$ of the set of nonzero quadratic residues of $(\mathbb{Z}_{13}, +)$, $U = \{u^2 mod 13 : u \in \mathbb{Z}_{13}$. This example is chosen strategically; U is the generating set of the Paley graph G on $(\mathbb{Z}_{13}, +)$, one of the archetypal examples of a dense pseudorandom graph. Accordingly, χ_U is the first row of the adjacency matrix of G, as this row corresponds to the indicator function of elements g of $\mathbb{Z}_{13}$ such that $0 + g \in U$. This kind of computation will turn out to be pivotal for our later proof that Cayley graphs general have at most loglinear Kolmogorov Complexity.

$(\mathbb{Z}_{13}, +)$ contains six nonzero quadratic residues, $U = \{1^2 \text{mod} 13 = 1,\ 2^2 \text{mod} 13 = 4,\ 3^2 \text{mod} 13 = 9,\ 4^2 \text{mod} 13 = 3,\ 5^2 \text{mod} 13 = 12,\ 6^2 \text{mod} 13 = 10\}$. Note that 13 is prime and $13 \quad 1 \text{mod} 4$, so U is closed under additive inverses as $1 \quad 12 \text{mod} 13 \in U$ and the set of nonzero quadratic residues form an equivalence relation on the multiplicative group $(\mathbb{Z}_{13} \quad \{0\}, \ )$.

Writing χ_U as an n dimensional vector indexed by $\mathbb{Z}_{13}$ we see that

$$\begin{bmatrix} Indices: & 0 & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 & 11 & 12 \\ \chi_U: & 0 & 1 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 1 \end{bmatrix}$$

Then the first block $B_1 = x_1^1$ is the length 1 string containing the fist bit

$\chi_U = \begin{bmatrix} 0| & 1 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 1 \end{bmatrix}$

B_2 is the shortest substring of $s = \chi_U$, starting at the index right after the previous block B_1 ended (so starting at 2), that does not occur as a substring of s_1^1. This turns out to be $B_2 = s_2^2 = \{1\}$. Thus,

$\chi_U = \begin{bmatrix} 0|1| & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 1 \end{bmatrix}$

Likewise B_3 is the shortest substring of $s = \chi_U$, starting at the index right after the previous block B_2 ended (so starting at 3), that does not occur as a substring of the substring of s from 1 to the ending index

Algorithm 1 LZ76 Compression

Require: binary string of length $n \in \mathbb{Z}_+$, $s = \{s_i\}_{i \in [n]}$ with $s_i \in \{0,1\} \quad \forall i \in [n]$

Define the starting block of our encoding $B_1 \quad x_1^1$.

Suppose after k steps we have encoded the first n_k bits of s as $B_1, B_2, ..., B_k = x_1^1, x_2^{n_2}, ..., x_{n_k \ 1+1}^{n_k}$

...

Set the following parameters:

uniqueSubstring $\quad FALSE$

$n_{k+1} \quad n_k + 1$

...

Let 'FindSubstring(pattern, string)' be a method implementing convolution-based string pattern matching that returns TRUE when pattern is a substring of string, FALSE otherwise.

while not(uniqueSubstring) **do**

if FindSubstring($x^{n_{k+1}}{}_{n_k+1}, x^{n_k}{}_1$) returns $FALSE$ **then**

$B_{k+1} \quad x^{n_{k+1}}{}_{n_k+1}$

uniqueSubstring $\quad TRUE$

else

$n_{k+1} \quad n_{k+1} + 1$

end if

end while

Repeat until $n_k \quad n$

—

return $\{B_j\}_{j \in k}$

of the previous block s_1^2. This turns out to be $B_3 = s_3^4 = \{011\}$. Thus,

$\chi_U = \begin{bmatrix} 0 & 1 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 1 \end{bmatrix}$

Proceeding like this up until the end of s yields the following partition into blocks:

$\chi_U = \begin{bmatrix} 0 & 1 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 1 \end{bmatrix}$

$\{B_1 = \{0\}, B_2 = \{1\}, B_3 = \{011\}, B_4 = \{00\}, B_5 = \{001\}, B_6 = \{101\}$.

We conclude that χ_U has a Lempel -Ziv complexity of six.

5.2 Pseudorandomness and Randomness

5.2.1 Pseudorandom Graphs

'Pseudorandomness' can refer to a myriad of graph theoretic properties that random graphs provably have with high probability. One of the first groups to describe pseudo- or quasi-random graphs were Chung, Graham, and Wilson, who originally enumerated a list of properties random graphs have with high probability and proved they are equivalent for dense graphs in 1989 [11]. We will focus on two of pseudorandomness properties: discrepancy pseudorandomness, which concerns the connectivity of a graph, and spectral pseudorandomness, which concerns the eigenspectrum of the graph.

We will first develop the notion of pseudorandomness for families of graphs with dense connections from Chung, Graham, and Wilson's work, and then dense discrepancy and spectral pseudorandomness properties are equivalent for d regular graphs (although it is worth noting Chung et' al.'s proof works for any simple undirected graph)[11][8].

We will then adapt the discrepancy and spectral pseudorandomness conditions to sparse graphs using later work by Conlon, Fox, and Zhao in 2014, and then summarize their proof that sparse discrepancy and spectral pseudorandomness properties are equivalent for sparse Cayley graphs [12][8].

Definition 5.3 (Dense Discrepancy Pseudorandomness) *Let $\{G_n = (V_n, E_n)\}$ be a family of dense graphs with constant order edge density ρ. Then $\{G_n\}$ satisfies the discrepancy pseudorandomness property if for vertex subsets $S, T \quad V_n$ we have that $|e(S,T) \quad \rho|S||T|| = o(n^2)$*

The discrepancy pseudorandomness condition is motivated by the behavior of binomial random graphs. In a a binomial random graph $G = (V, E)$ with probability parameter $\rho \in (0, 1)$ we have that $\rho|S||T|$ is the expected number of edges between disjoint subsets S and T of the vertex set V. So intuitively, a graph with edge density ρ having a relatively small discrepancy $|e(S,T) - \rho|S||T|| \leq \alpha|S||T|$ for all subsets S, T of V, including disjoint subsets, behaves similarly to a binomial random graph with probability ρ.

Definition 5.4 (Dense Spectral Pseudorandomness) *Let $\{G_n = (V_n, E_n)\}$ be a family of dense graphs. Then $\{G_n\}$ satisfies the spectral pseudorandomness property if the eigenvalues of the adjacency matrix of G_n, listed $\{\lambda_i\}_{i \in [n]}$ in descending order, satisfy $\lambda_1 = \rho n + o(n)$ and $\lambda = max_{i \in \{2,\ldots,n\}}\{|\lambda_i|\} = max(\lambda_2, |\lambda_n|) = o(n)$*

Theorem 5.1 (Equivalence of Discrepancy and Spectral Pseudorandomness for Dense Graphs) *Let $\{G_n = (V_n, E_n)\}$ be a family of dense d-regular graphs with constant order edge density ρ. Then $\{G_n\}$ satisfies the discrepancy pseudorandomness condition if and only if it satisfies the spectral pseudorandomness condition. First proved in [11]*

To prove this theorem, we first need to prove the Expander-Mixing Lemma, an important result relating the discrepancy of a d regular graph and spectral gap (i.e., the maximal absolute value λ of the nontrivial eigenvalues of the adjacency matrix) [4]. The essence of the lemma is that if spectral gap of a graph is small, then the graph will have small discrepancy and so be quite well connected relative to the density of its edges, similar to what you would expect from a binomial random graph of the same edge density.

Lemma 5.2 (Expander Mixing Lemma) *For any $(n, d, \lambda$ graph $G = (V, E)$, if $S \subseteq V$ and $T \subseteq V$ such that $S \setminus T = \emptyset$ and $e(S, T)$ denotes the set of edges with one vertex in S and one in T, then*

$$|e(S,T) - \frac{d}{n}|S||T|| \leq \lambda\sqrt{|S||T|}$$

PROOF OF 5.2: Let $(n, d, \lambda$ graph $G = (V, E)$ have adjacency matrix A. Let us write $V = [n]$ by enumerating the n vertices of V. G is a simple, undirected graph so A is a symmetric matrix, thus by spectral theorem A has an orthonormal eigenvector basis $\{v_i\}_{i \in [n]}$ associated to it's eigenvalues $\{\lambda_i\}_{i \in [n]}$ (written in decreasing order with multiplicity). As observed in remark 4.1, G degree d implies that $\lambda_1 = d$ and the associated eigenvector $v_i = \frac{1}{\sqrt{n}}(1, ..., 1)$, the normalized all ones $nx1$ vector.

Let $S, T \subseteq V = [n]$ and let $\chi_S : [n] \to \{0, 1\}$ and $\chi_T : [n] \to \{0, 1\}$ be the indicator functions of S and T respectively. Note that we can write χ_S as a length n vector in $\{0, 1\}^n$ where $chi_{S\,i} = 1 \iff i \in S \subseteq [n]$ (likewise for chi_T).

Now, we compute $e(S.T)$ from A and the indicator functions of S and T. We have that $A\chi_{T\,i} = \sum_{j \in [n]} A_{ij}\chi_{T\,j}$, so $\chi_S{}^{\top} A\chi_T = \sum_{i \in [n]}\sum_{j \in [n]} A_{ij}\chi_{S\,i}\chi_{T\,j}$. Observer that $\forall i, j \in [n]$ we have that $A_{ij}\chi_{S\,i}\chi_{T\,j} = 1 \iff (i, j) \in E, i \in S$, and $j \in T$, and $A_{ij}\chi_{S\,i}\chi_{T\,j} = 0$ otherwise. Then,

$$\chi_S{}^{\top} A\chi_T = \sum_{i \in [n]}\sum_{j \in [n]} A_{ij}\chi_{S\,i}\chi_{T\,j} = e(S,T)$$

.

Our orthonormal eigenbasis $\{v_i\}_{i \in [n]}$ of A spans $\mathbb{R}^n$, so we can write χ_S and χ_T in terms of $\{v_i\}_{i \in [n]}$: $\chi_S = \sum_{i \in [n]} a_i v_i$ and $\chi_T = \sum_{i \in [n]} b_i v_i$ where $a_i := \chi_S^{\top} v_i$ and $b_i := \chi_T^{\top} v_i$.

Thus, $A\chi_T = A(\sum_{i \in [n]} b_i v_i) = \sum_{i \in [n]} b_i(Av_i) = \sum_{i \in [n]} b_i(\lambda_i v_i) = d(b_1)(v_1) + \sum_{i \in \{2,\ldots,n\}} b_i(\lambda_i v_i)$.

By the orthonormality of $\{v_i\}_{i \in [n]}$ we have that $v_i^{\top} v_j = 1 \forall i = j$ and $v_i^{\top} v_j = 0 \forall i \neq j$. Therefore,

$$e(S,T) = \chi_S{}^{\top} A\chi_T = \sum_{i \in [n]}\sum_{j \in [n]} \lambda_j a_i b_j (v_i^{\top} v_j) = \sum_{i \in [n]} \lambda_i a_i b_i = \sum_{i \in [n]} \lambda_i(\chi_S^{\top} v_i)(\chi_T^{\top} v_i)$$

Observing that $\lambda_1(\chi_S^{\top} v_1)(\chi_T^{\top} v_1) = d(\sum i \in [n] \frac{1}{\sqrt{(n)}} \chi_S(i))(\sum_{j \in [n]} \frac{1}{\sqrt{(n)}} \chi_T(j)) = \frac{d}{n}|S||T|$. Therfore we have that $e(S,T) = \frac{d}{n}|S||T| + \sum_{i \in \{2,\ldots,n\}} {}_i(\ {}_S^{\top} v_i)(\ {}_T^{\top} v_i)$ and by Cauchy Schwarz we obtain,

$$|e(S,T) - \frac{d}{n}|S||T|| = |\sum_{i \in \{2,\ldots,n\}} \lambda_i(\chi_S^{\top} v_i)(\chi_T^{\top} v_i)| \leq \lambda\left[\sum_{i \in [n]} (\chi_S^{\top} v_i)^2\right]^{\frac{1}{2}}\left[\sum_{i \in [n]} (\chi_T^{\top} v_i)^2\right]^{\frac{1}{2}}$$

Let B be a matrix with column vectors equal to the vectors in the orthonormal eigenbasis $\{v_i\}_{i \in [n]}$. Then we have that $\chi_S^T B_i = \chi_S^T v_i \quad \forall i \in [n]$. Thus, we have that $\sum_{i \in [n]} (\chi_S^T v_i)^2 = \sum_{i \in [n]} \chi_S^T B_i^2 = (\chi_S^T B)(\chi_S^T B)^T = \chi_S^T B B^T \chi_S$. But B is an orthonormal matrix so $BB^T = I$, thus $\sum_{i \in [n]} (\chi_S^T v_i)^2 = \chi_S^T \chi_S = \sum_{i \in [n]} (\chi_{Si})^2$. Further, by definition of the characteristic function $\sum_{i \in [n]} (\chi_{Si})^2 = \sum_{i \in S \subseteq [n]} 1^2 = \sum_{i \in S \subseteq [n]} 1^2 = |S|$. By identical argument, $\sum_{i \in [n]} (\chi_T^T v_i)^2 = \sum_{i \in [n]} (\chi_{Ti})^2 = |T|$. Then we obtain

$$|e(S,T) - \frac{d}{n}|S||T|| \leq \lambda \left[\sum_{i \in [n]} (\chi_S^T v_i)^2 \right]^{\frac{1}{2}} \left[\sum_{i \in [n]} (\chi_T^T v_i)^2 \right]^{\frac{1}{2}} = \lambda \sqrt{|S||T|}$$

QED

With the expander mixing lemma in hand, we can now prove the dense pseudorandom equivalence theorem for d regular graphs.

PROOF OF 5.1:

First observe that in a family $\{G_n\}_{n \in Z^+}$ of d- regular graphs have edge density $\rho = \frac{d}{n}$. This is because $|E_n| = \frac{dn}{2}$ (d edges multiplied by n vertices, dividing by 2 so the undirected edges aren't double-counted) and so $|E_n| = \frac{nd}{2} = (\binom{n}{2})(\frac{d}{n} + \frac{d}{n(n-1)})$. $\lim_{n \to \infty} \frac{d}{n(n-1)} = 0$ so $\frac{d}{n(n-1)} = o(1)$ and we have $|E_n| = \binom{n}{2}(\frac{d}{n} + o(1))$.

Thus, $\lambda_1 = d = \frac{dn}{n} = \rho n + 0$ so to show a d-regular family of graphs satisfies spectral pseudorandomness it is sufficient to prove that the spectral gap of $\{G_n\}_{n \in Z^+}$ grows sub-linearly, i.e., $\lambda = o(n)$.

Dense spectral pseudorandomness $\Rightarrow$ dense discrepancy pseudorandomness: Suppose $\lambda = o(n)$. Then by the Expander- Mixing Lemma we have that $\forall S, T \subseteq V_n$ we have $|e(S,T) - \frac{d}{n}|S||T|| = |e(S,T) - \rho|S||T|| \leq \lambda \sqrt{|S||T|} \leq \lambda \sqrt{n \cdot n} = \lambda n = o(n^2)$.

QED

Dense discrepancy pseudorandomness $\Rightarrow$ dense spectral pseudorandomness: Proof omitted, not relevant to any work we have done so far.

Next, we define sparse pseudorandomness.

Definition 5.5 (Sparse ϵ- Discrepancy Pseudorandomness) *Let $\{G_n = (V_n, E_n)\}$ be a family of sparse d-regular graphs (i,e., $d = o(n^2)$) with edge density ρ and let $\epsilon > 0$. Then $\{G_n\}$ has the ϵ- discrepancy pseudorandomness property if for vertex subsets $S, T \subseteq V_n$ we have that $|e(S,T) - \rho|S||T|| \leq \epsilon dn$*

Definition 5.6 (Sparse ϵ- Spectral Pseudorandomness) *Let $\{G_n = (V_n, E_n)\}$ be a family of sparse d-regular graphs (i,e., $d = o(n^2)$) with edge density ρ and let $\epsilon > 0$. Then $\{G_n\}$ has the ϵ- spectral pseudorandomness property if the eigenvalues of the adjacency matrix of G_n, listed $\{\lambda_i\}_{i \in [n]}$ in descending order, satisfy $\lambda_1 = \rho n + o(n)$ and $\lambda = max_{i \in \{2,\ldots,n\}} \{|\lambda_i|\} = max(\lambda_2, |\lambda_n|) \leq \epsilon d$*

Theorem 5.3 *Let $\{C_n\}$ be a family of Cayley graphs on the group $(\Gamma_n, \star)$ of size $|\Gamma_n| = n$ and with group operation $\star : \Gamma_n \times \Gamma_n \to \Gamma_n$, with generating set $U \subseteq \Gamma_n$ of size $|U| = d$ that is closed under inverses. Then, C_n is undirected and d regular. Let $\{C_n\}$ be sparse, i.e., $\{C_n\}$ has edge density $lim_{n \to \infty} \frac{|E_n|}{\binom{n}{2}} = lim_{n \to \infty} \frac{d}{n} = 0$.*

Then, $\{C_n\}$ has the ϵ - - spectral pseudorandomness property $\Rightarrow$ $\{C_n\}$ has the ϵ- discrepancy pseudo randomness property. First Proved in [12].

(NOTE: Conlon et. al. prove the converse in [12] but we omit the proof as it is not needed for any of our later results)

PROOF OF 5.3:

ϵ- spectral pseudorandomness $\Rightarrow$ ϵ- discrepancy pseudorandomness: Similar to the previous proof, this result follows almost immediately from the Expander Mixing Lemma.

Suppose $\lambda \leq \epsilon d$. Then by the Expander- Mixing Lemma we have that $\forall S, T \subseteq V_n$ we have $|e(S,T) - \frac{d}{n}|S||T|| = |e(S,T) - \rho|S||T|| \leq \lambda \sqrt{|S||T|} \leq \lambda \sqrt{n \cdot n} = \lambda n \leq \epsilon dn$.

QED

5.2.2 Random Graphs

For the purposes of this research we will adopt the definition of a truly random graph G, employed in other literature on graph algorithmic entropy, as one that is not compressible, in the sense that there is no lossless encoding of the graph with fewer bits than its adjacency matrix representation (which has $\binom{n}{2}$ bits when the graph G is undirected and $n(n \quad 1)$ bits when G is directed).

Definition 5.7 (Random Graphs) *A graph G is random if* $K(G) = O(n^2)$

This definition is meaningful in the sense that it has been proven that the probability of uniformly randomly choosing a binary string s of length l out of the set of all such binary stings $f0,1g^l$ such that s has strictly less than $O(l)$ algorithmic entropy, i.e., $K(s) = o(l)$, converges to 0 in probability as $l \; ! \; 1$ [9]. We formally state and prove this in the following lemma:

Lemma 5.4 *For any* $l \; 2 \; Z_+$ *and any constant* $c \; 2 \; [l]$ *we have that there are* $2^l \quad 2^{l \; c}$ *binary strings of length* $l \; s \; 2 \; f0,1g^l$ *such that* $K(s) \quad l \quad c$ *[9].*

It follows that the probability of uniformly randomly choosing s *from* $f0,1g^l$ *such that* $K(s) < l \quad c$ *is at most* $\frac{1}{2^c} \quad 8c \; 2 \; [l]$.

PROOF OF 5.4:

Consider the set of binary strings of length $l, f0,1g^l$. Then for any constant $c \; 2 \; [l]$, there are exactly $jf0,1g^l \; {}^c j = 2^{l \; c}$ binary strings of length $l \quad c$, so there are only $2^{l \; c}$ ways to compress a string of length l by c bits. Then there must be at least $2^l \quad 2^{l \; c}$ binary strings of length l that do not compress by any more than c bits, i.e, $jfs \; 2 \; f0,1g^l : K(s) \quad l \quad cgj \quad 2^l \quad 2^{l \; c}$.

Then if we chose s form $f0,1g^l$ with uniform probability, $8c \; 2 \; [n]$ we have that $K(s) \quad l \quad c$ with probability at least $\frac{2^l \; {}^c(2^c \quad 1)}{2^l} = \frac{2^c \quad 1}{2^c}$, and $K(s) < l \quad c$ with probability at most $\frac{1}{2^c}$. QED

We can represent undirected graphs as binary strings length $\binom{n}{2}$), most of which have algorithmic entropy $\binom{O(n}{2) = O(n^2)}$ by Lemma 5.4. Thus graphs of subrandom Kolmogorov complexity are vanishingly rare for large n.

Our goal will be to demonstrate that the pseudorandom graphs under consideration nevertheless have strictly less than quadratic Algorithmic entropy, making them candidates for lower-complexity replacements for random graphs.

6 Results

6.1 Cayley Graphs

As we established in the preliminary section, Cayley graphs have highly desirable algebraic structure that imparts a variety of useful properties, such as an adjacency matrix with an eigenspectrum consisting of the graph's discrete Fourier components (Theorem with Theorem 4.1). In this section we use this property to prove a variety of both dense and sparse Cayley graphs display spectral pseudorandomness characteristics (and thus discrepancy pseudorandomness) [8].

Intuitively, we would expect Cayley graphs to have sub-random algorithmic complexity, as they are determined entirely by the group $(H, \;)$ and the generating set $U \quad H$ they are defined on, which should not take at most a linear in n number of bits to encode (possibly with algorithmic overhead to keep track of labels). We present a proof that this is indeed the case for Paley graphs on the additive group $(Z_n, +)$ for any $n \; 2 \; Z_+$. This gives us a rich repository of easy to construct and work with graphs with both proven pseudorandom properties and sub-random algorithmic entropy.

6.1.1 Paley Graphs

Consider the Cayley graph $C((Z_p, +), U)$ for p prime with generating set $U = fu \; 2 \; Z_p \; {}_{f0g} : u \quad a^2 mod(p)$ for some $a \; 2 \; Z$. Then, $(u, v) \; 2 \; E \quad (\;) \quad v = s + u$ where s is a nonzero quadratic residue in Z_p. We will restrict p such that $p \quad 1 mod 4$, so that u square in $Z_p \quad (\;) \quad u$ square in Z_p and thus, $v = s + u \; =) \quad u = \quad s + v$ and

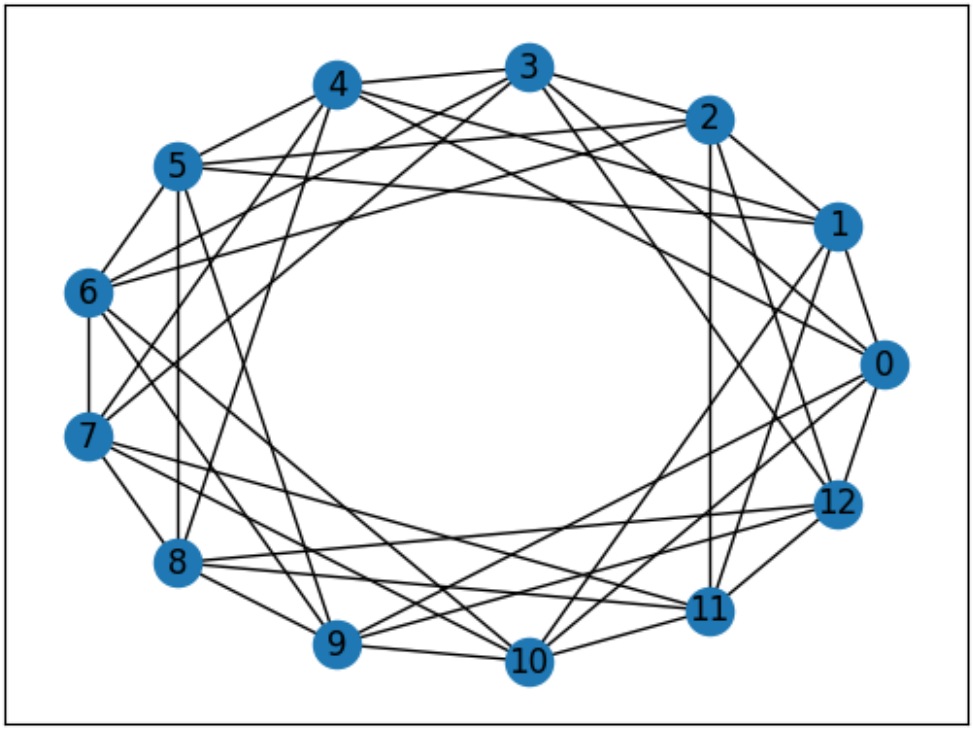

Figure 2: The Paley graph $C((\mathsf{Z}_{13},+),U = \{a^2 : a \in \mathsf{Z}_{13}\})$

s quadratic residue ( ) s quadratic residue give us that $(u,v) \in E$ ( ) $(v,u) \in E$ (ie. G undirected). Fig. 2 depicts the Paley graph $C((\mathsf{Z}_{13},+),S)$

Let $C(\mathsf{Z}_p,U)$ be a Paley graph with U be the set of nonzero quadratic residues as above. Being a nonzero quadratic residue is provably an equivalence relation on $\mathsf{Z}_p \setminus \{0\}$ with equivalence classes U and $\mathsf{Z}_p \setminus \{0\} \setminus U$, so $|U| = \frac{|\mathsf{Z}_p \setminus \{0\}|}{2} = \frac{p-1}{2}$. Thus, there are $\frac{p-1}{2}$ distinct squares in $\mathsf{Z}_p \setminus \{0\}$, so $C(\mathsf{Z}_p,U)$ is $\frac{p-1}{2}$- regular.

Despite their clear symmetry, we claim Paley graphs are both pseudorandom and have an adjacency matrix with high Shannon energy. The high Shannon entropy follows from the $\frac{p-1}{2} \approx \frac{p}{2}$, observing that the Shannon entropy of adjacency matrix is maximized in $\frac{|V|}{2}$-regular graphs. The pseudorandomness of Paley graphs is a well-known result that is outlined in the theorem below.

Theorem 6.1 *Let $C(\mathsf{Z}_p,U)$ be a Paley graph on the group $(\mathsf{Z}_p,+)$ for prime $p \equiv 1 \bmod 4$. Let $\{\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_n\}$ be the eigenspectrum of the adjacency matrix A_G of G. Then, $max(\lambda_2,|\lambda_n|) = O(\sqrt{p})$. Thus, $C(\mathsf{Z}_p,U)$ is pseudorandom 8.*

PROOF OF 6.1:

Per Lemma 4.1 we know that the Fourier characters $\{\omega^k\}$ are the eigenvectors corresponding to eigenvalues $\lambda_{k+1} = \widehat{\chi_S}(k)$ $\forall k \in \mathsf{Z}_p$, where $\chi_S : \mathsf{Z}_p \to \{0,1\}$ is the indicator function of $S = \{u \in \mathsf{Z}_p \setminus \{0\} : u \equiv a^2 \bmod p\}$. We can compute the Fourier components of χ_S using the Fourier transform $\widehat{\chi_S}(k) = \sum_{j=0}^{p-1} \chi_S(j)\omega(j)^k = \sum_{j=0}^{p-1} \chi_S(j)e^{\frac{i2\pi kj}{p}}$. Thus,

$$\lambda_{k+1} = \widehat{\chi_S}(k) = \sum_{j=0}^{p-1} \chi_S(j)e^{\frac{i2\pi kj}{p}} = \sum_{j \in S} 1 \cdot e^{\frac{i2\pi kj}{p}}$$

Observe that for any $l \in \{1,\ldots,\frac{p-1}{2}\}$ we have $(p-l)^2 \bmod p = (p^2 - 2pl + l^2) \bmod p \equiv l^2 \bmod p$, so the sequence $\{l^2 \bmod p\}_{l=0}^{p-1} = \{0\}, \{l^2 \bmod p\}_{l=1}^{\frac{p-1}{2}}, \{(p-l)^2 \bmod p\}_{l=\frac{p-1}{2}}^{1}$ iterates through $\{0\}$ and then S twice. Thus,

$$\sum_{j \in S} e^{\frac{i2\pi kj}{p}} = \frac{\sum_{l=1}^{p-1} e^{\frac{i2\pi kl^2}{p}}}{2} = \frac{(\sum_{l=0}^{p-1} e^{\frac{i2\pi kl^2}{p}}) - 1}{2}$$

.

Using Gauss sums we see that $p \equiv 1 mod(4) \Rightarrow \sum_{l=0}^{p-1} e^{\frac{i2\pi kl^2}{p}} = \sqrt{(p)}\ \forall k \in \mathbb{Z}_p - \{0\}$. Thus,

$$\lambda_{k+1} = \chi_S(k) = \frac{\sqrt{p}}{2} - 1 \quad \forall k \in \mathbb{Z}_p - \{0\}$$

Thus $\lambda = O(\sqrt{p})$
QED

Example: Bounding the Algorithmic Entropy of Paley Graphs

Theorem 6.2 (Algorithmic Entropy of Undirected Paley Graphs) *Let $\{P_n = C(\mathbb{Z}_p, U_n))$ be the family of Paley graphs on additive groups $(\mathbb{Z}_p, +)$ with p prime and $p \equiv 1 mod 4$, with $U_n := \{u^2 mod p : u \in \mathbb{Z}_p\}$. Then Paley graphs have linear Lempel-Ziv complexity, $LZ(C(\mathbb{Z}_p, U_n)) = O(n)$. This gives us loglinear algorithmic entropy, which is sub-random.*

PROOF OF 6.2
Let A be the adjacency matrix of $P_n = C(\mathbb{Z}_p, U_n)$ with entries $a_{ij} = 1 \Leftrightarrow i - j \in S$.
Observe that Paley graphs (and more generally Cayley graphs on the additive subgroup of a finite field $(\mathbb{Z}, +, \cdot)$ are circulent. That is, $(i,j) \in E \Leftrightarrow i = j+s$ for $s \in S \Leftrightarrow i+1 = (j+1)+s \Leftrightarrow (i+1, j+1) \in E$, so $a_{ij} = a_{(i+1)(j+1)}\ \forall i,j \in \mathbb{Z}_p$. Then, $\forall i \in \{0, ..., p-1\}$ the ith row vector $A_i = (a_{i0}, ..., a_{i(p-1)})$ of A satisfies

$$(a_{i0}, a_{i1} ..., a_{i;(p-2)}, a_{i(p-1)}) = (a_{(i+1)1} a_{(i+2)1}, ..., a_{(i+1)(p-1)}, a_{(i+1)0})$$

.

Therefore, A_{i+1} is a circular permutation/ one bit shift of $A_i\ \forall i \in \{0, ...p-2\}$. Thus, any row A_i of A for $i \in \{1, ..., p-1\}$ is reproducible from the first row of the adjacency matrix $A_0 = (a_{00}, ...a_{0p-1})$ via i circular shifts. Specifically, if we flatten the adjacency matrix A into a $1 \times \binom{n}{2}$ vector v_A by concatenating

$$A_{0[1:p-1]}, A_{1[2:p-1]}, ...A_{i[i+1;p-1]}, ..., A_{p-2[p-1]}$$

. The partitioning of the upper triangular part of the adjacency matrix of P_n into blocks/codewords is illustrated in figure 3.

Then, applying the LZ76 algorithm to resultant v_A with a dictionary and look ahead buffer of length $p-1$ each yields at most $O(p)$ code words, as $A_{i[i+1;p-1]}$ is a substring of $A_{i-1[i;p-1]}$ for all $i \in [p-1]$. A_{i-1} has length $p-i$ and so is contained entirely in its dictionary window by construction. Thus, each row vector A_i encoded after A_0 only adds a pointer to a substring of A_0 to the encoding (illustrated using the red and orange lines in fig. 3).

A_0 can be encoded in exactly p bits, plus an additional constant number of bits for each row after. Then, $LZ(C(\mathbb{Z}_p, S)) = O(p)$

Then, adding logarithmic overhead for storing the labels of the codewords themselves, this gives us $O(nlogn)$ algorithmic entropy- less than the $O(n^2)$ algorithmic entropy of the adjacency matrices of random graphs.

QED

Now we use the argument in Theorem 6.2 to prove that any undirected Cayley graph on the additive subgroup of the finite field $\{\mathbb{Z}_n, +, \cdot\}$ for any $n \in \mathbb{Z}_+$ has sub-random Kolmogorov complexity.

Theorem 6.3 (Algorithmic Entropy of Undirected Cayley Graphs) *Let $\{C_n\}$ be a family of undirected Cayley graphs on the cyclic group $(\mathbb{Z}_n, +)$ with generating sets $U_n \subset \mathbb{Z}_n$. Then the adjacency matrices of $\{C_n\}$ have $O(n)$ Lempel- Ziv complexity. This gives us loglinear algorithmic entropy, which is sub-random.*

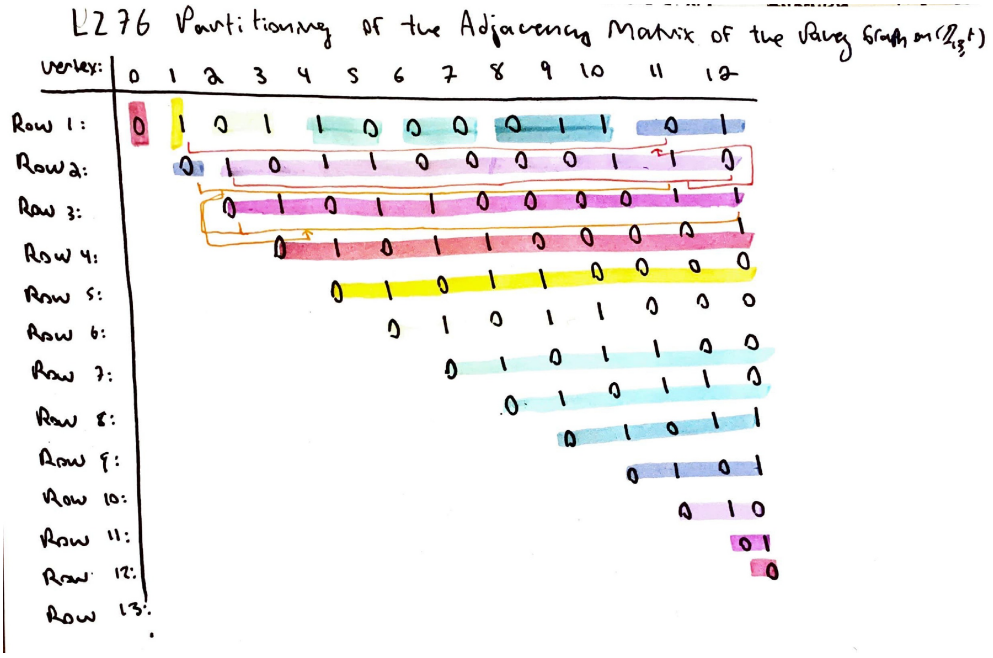

Figure 3: The LZ76 partitioning of the upper triangular adjacency matrix of the Paley graph $C((\mathsf{Z}_{13}, +), U = \{a^2 : a \in \mathsf{Z}_{13}\})$, with blocks highlighted in different colors and two example pointers to previous rows illustrated in red and orange lines.

PROOF OF 6.3:

Let A denote the adjacency matrix of C_n, and observe that C_n defined on the cyclic group $(\mathsf{Z}, +)$ $=)$ C_n circulent. For observe that each row of the adjacency matrix A_i corresponds to the vertex i $1 \in \mathsf{Z}_n$, and so we have $A_{ij} = 1$ $( )$ j $i \in U_n$. Thus, A_i is the indicator function χ_{S_i} of the set $S_i := \{s \in \mathsf{Z}_n : (i$ $1) + s \in U\}$. In particular, A_1 corresponds to 0, the identity element, and so $A_1 = \chi_U$ as $U = S_i := \{s \in \mathsf{Z}_n : 0 + s = s \in U\}$. Using LZ76 we can encode A_1 in at most n codewords, often less.

Because we're working with undirected graphs we see that the adjacency matrix of C_n is symmetric and we only need encode the upper triangular part, $\{A_1, A_{2[2:n]}, ...A_{i[i:n]}, ...A_{n\ 1}[(n$ $1) : n], A_{nn}\}$

We see that $\forall i \in [n$ $1]$ consecutive row vectors A_i and A_{i+1} are single bit circular permutations of each other, as $A_{i+1} = \chi_{\{fs2[n]:i+s2U\}} = \chi_{\{fs2[n]:(i\ 1)+(s+1)2U\}}$. All addition is taken modulo n so this gives us $A_{ij} = \chi_{S_i}(j$ $1) = \chi_{S_{i+1}}(j) = A_{(i+1)(j+1)}$ modulo n as desired.

Then, $\forall i \in [n$ $1]$, we have that $A_{(i+1)[(i+1):n]} = A_{i[i:n\ 1]}$, i.e, $A_{(i+1)[(i+1):n]}$ is a direct substring of $A_{i[i:n]}$, so applying the LZ76 algorithm encodes every row after the first row as its own codeword/block, resulting in only a linear number of codewords and thus linear Lempel Ziv complexity.

Then, adding logarithmic overhead for storing the labels of the codewords themselves, this gives us $O(nlogn)$ algorithmic entropy- less than the $O(n^2)$ algorithmic entropy of the adjacency matrices of random graphs.

QED

Empirical Lempel Ziv Complexity Validation Our theoretical calculation for the algorithmic entropy of undirected Cayley graphs on additive groups is validated by empirical computation of the Lempel-Ziv compression on both Paley graphs and cycle graphs using the Lempel-Siv Markov chain algorithm (LZMA). LZMA is an optimized version of the LZ77 algorithm, based on the LZ76 algorithm, that achieves "higher compression rate, faster decompression, and lower memory requirements" 13.

Recall from remark 4.2 that cycle are the sparsest possible undirected, connected Cayley graphs on $(\mathsf{Z}_p, +)$ for p prime up to isomorphism. Then comparing Paley graphs, which are dense, to cycle graphs captures the breadth of undirected Cayley graph connectivity.

Validation was performed using code produced in python for the TRIPODS 2023 Summer Research

Program using the numpy, lzma, and networkX packages, cited in the bibliography at [14]. Figures 4 and 5 depict the compression size vs original number of graph nodes curve for cycle graphs and Paley graphs respectively. For the sake of comparison, both graphs were evaluated for prime nodes congruent 1mod4.

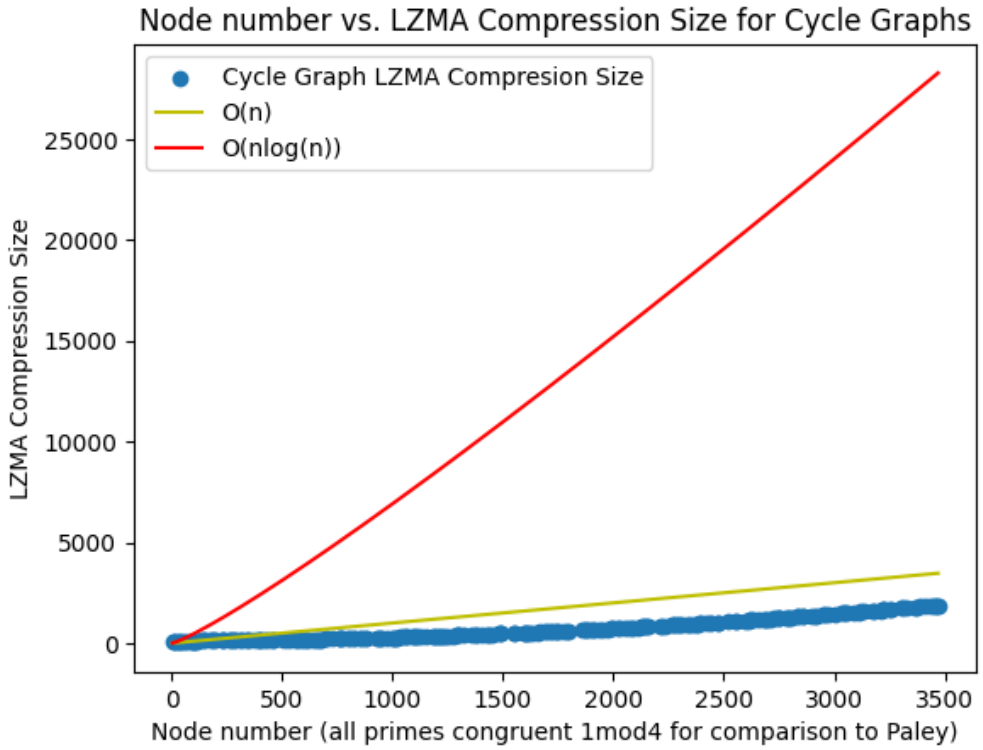

Figure 4: The LZMA compression size versus number of nodes in a cycle graph

Both curves are convex, consistent with our loglinear bound we proved for algorithmic entropy in Theorem 6.3. Note however the relatively flat (and indeed, sub-linear) curve of the cycle graph compression compared to the Paley graph compression.

6.2 Applications to Echo State Networks

In this section we discuss the possible applications of Cayley graphs as replacements for random reservoirs in Echo State networks, a variety of recurrent neural network.

Neural networks learn patterns in input data is that they embed the input in a high dimensional space, seek out local minima in the difference between the embedded data and the desired output data, and then adjust the parameters of the embedding accordingly. Feedforward neural networks, the most straightforward architecture of neural network, embed data by composing a linear transformation of the input data into a higher dimensional space with nonlinear convex 'activation functions' such as the sigmoid function and then adjusting the weights of the linear embedding via some manner of 'backpropagation' algorithm - propagating error backwards through the network to adjust neural network parameters in a way that minimizes the error [15].

Recurrent neural networks have emerged as a popular alternative network architecture that allows for the output of the neural network to feed back into the network as a new input. Thus, the network is not just a nonlinear function embedding the input into a higher dimensional space; it is a dynamical system, an iterated map depending just as much on its past values as it does its original inputs. This makes them ideal for forecasting time series data, such as dynamical systems or natural language. However, training the feedback or 'recurrent' weights of the network is much harder due to the the presence of time dependence, necessitating error be propagated through time in order to adjust the network weights [16]. This is quite

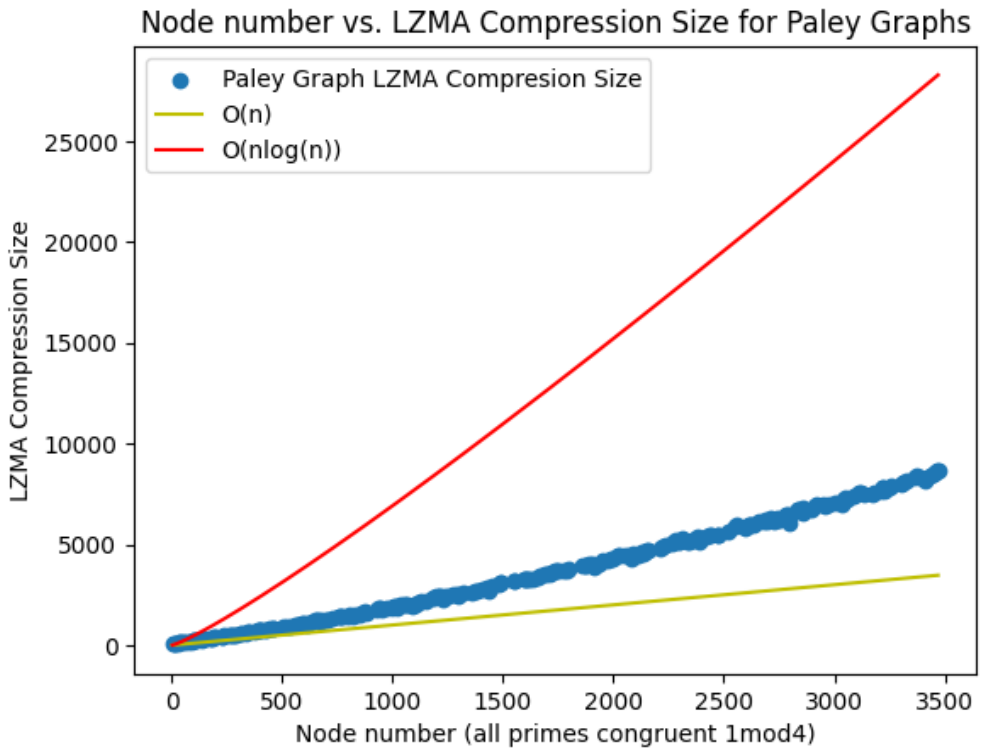

Figure 5: The LZMA compression size versus number of nodes in a Paley graph

computationally expensive. Echo state networks (ESNs) are a variety of recurrent neural network that circumvent the difficulties posed by propagation by fixing the recurrent weights. Thus, the only weights that require training are the linear weights reading computing the output of the network by linearly combining reservoir states. This can be done with normal backpropagation [17].

Most ESNs use a reservoir with random connectivity, such as a sparsely connected Erdős–Rényi graph, in keeping with the widely held belief that randomness is one of the most effective ways to ensure the reservoir was sufficiently "rich" in neural correlations that a [16]. However, recent literature has started experimenting with other cycle architectures, with one exciting recent preprint by Li et.al. demonstrating that reservoirs comprised of a cycle graph (or in the authors' words, "full cycle permutation") and are "universal approximators of any unrestricted linear reservoir system" [19]. This raises the question of whether undirected Cayley graphs on finite fields have sufficient expressive power be suitable replacements for random reservoirs in ESNs, which is what we are currently investigating. This section elaborates on the technical details of constructing and evaluating the capacity of ESNs, and reports some of our preliminary findings with regard to Cayley graphs.

6.2.1 Echo State Networks

Architecture and training An echo state network (ESN) linearly maps an input time series with weight matrix W_in into a fixed 'reservoir', a graph making the body of the neural network with weighted adjacency matrix W. All vertices or 'neurons' of the reservoir have an associated state $X = fx_i(t)\ 2\ \mathsf{R}g_{i2[n]}$ at each time step $t\ 2\ \mathsf{Z}$, the evolution of which over time depends on both $u(t)$ and its past values $x(t \quad 1)$. Thus, the reservoir X is a dynamical system, and cycles in the weighted adjacency matrix of X comprise the recurrent connections of the neural network.

Thus, consider an ESN with k input units, n reservoir neurons, and l output units.

Let $fu(t)g_{t2\mathsf{Z}} = fu_i(t)g_{i2[k];t2\mathsf{Z}}$ is the input time series of k dimensional input values, $x(t)_{t2\mathsf{Z}} = x_i(t)_{i2[n];t2\mathsf{Z}}$ be the time series of n dimensional reservoir neuron states, and $y(t)_{t2\mathsf{Z}} = y_i(t)_{i2[l];t2\mathsf{Z}}$ be the time series

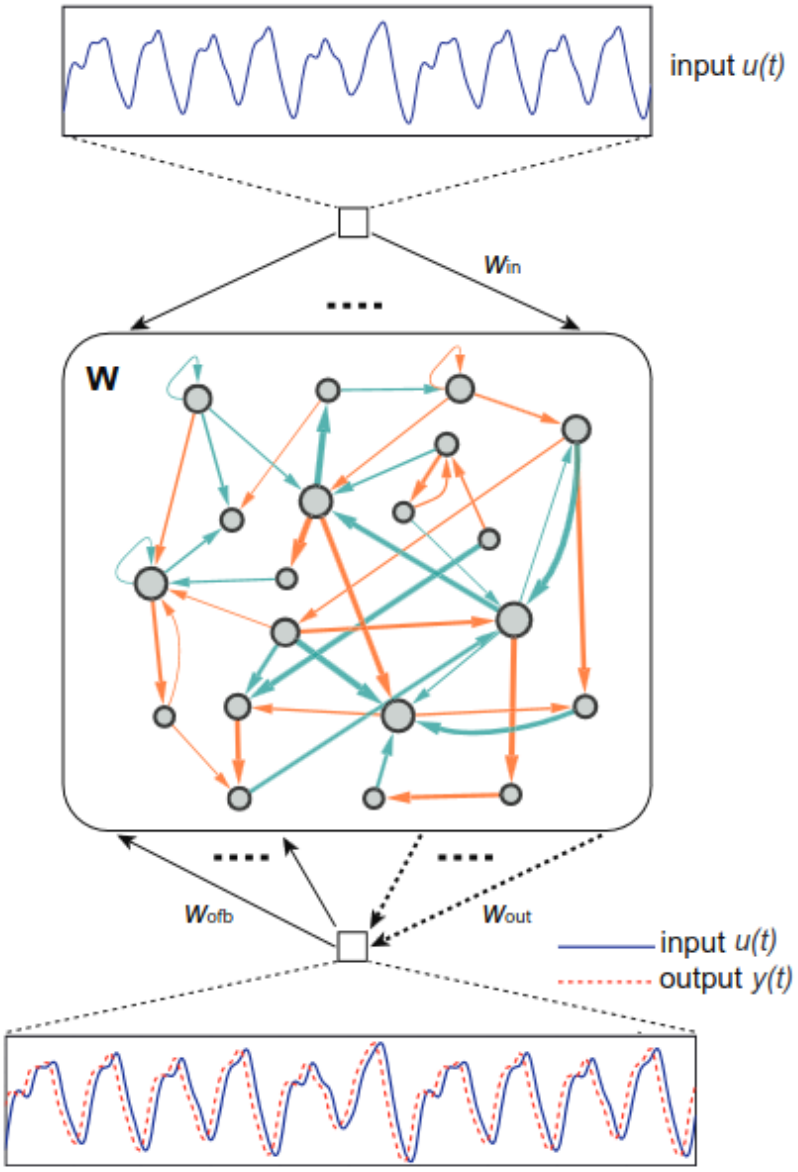

Figure 6: A diagram of the architecture of an echo state network depicting edges between reservoir nodes in orange and cyan representing positive and negative weights. Reprinted from "Tailoring Echo State Networks for Optimal Learning" by P. Aceituno, G. Yan, and Y. Liu, 2020, iScience, Volume(23), page number 2.

of l dimensional outputs. Finally, let f : R ! R be a convex nonlinear activation function (to apply f elementwise to a matrix M we simply write $f(M)$ by convention).

Then the states of the reservoir are updated as follows on time step t [16]:

$$\begin{cases} x(t) = f(Wx(t \quad 1) + W_{in}u(t)) \\ y(t) = W_{out}(x(t)) \end{cases}$$

(Note that W_{in} mapping $fu_i(t)g_{i2[k]}$ into the reservoir at time step t is $k \quad n$ matrix, W mapping

$\{x_i(t-1)\}_{i\in[n]}$ to $\{x_i(t)\}_{i\in[n]}$ at time step t is an $n \times n$ matrix, and W_{out} mapping $\{x_i(t)\}_{i\in[n]}$ to $\{y_i(t)\}_{i\in[l]}$ is a $n \times l$ matrix.)

Properties As mentioned, the recurrent part of the network is a reservoir comprising fixed weights W with fixed input weights W_{in}; only the linear output weights W_{out} are trained, using classic gradient descent and backpropogation algorithms [17].

The choice of reservoir graph is of vital importance for the performance of the network. The one strict condition on the reservoir weighted adjacency matrix W is that the eigenvalue of greatest modulus λ_1 of the weighted adjacency matrix W satisfy $|\lambda_1| < 1$ [16]. This can be obtained by normalizing the network weights by $|\lambda_1|$. In particular, if we want W to reflect the network topology of an undirected d-regular graph with adjacency matrix A then $W = \frac{1}{d}A$ as $\lambda_1 = d$.

For some intuition as to why such normalization is necessary, consider f equal to the identity function and W_{in} equal to the zero vector (so no inputs are driving the dynamics of the reservoir. Then if you diagonalize the matrix (which you can do for a weighted undirected graph, by symmetry of the adjacency matrix by the spectral theorem), you obtain that $x(t) = Wx(t-1) = B\Lambda B^{-1}$ where Λ is the diagonal matrix of eigenvalues ($\Lambda_{ii} = \lambda_i \ \forall i \in [n]$, zero elsewhere) and B is the associated invertible change of basis matrix. Then,

$$x(t) = W^n x(t-n) = (B\Lambda B^{-1})^n x(t-n) = B\Lambda^n B^{-1}$$

.

Observe $\Lambda^n{}_{ii} = \lambda_i^n \ \forall i \in [n]$, zero elsewhere. Thus if there is a diagonal entry in Λ with $|\lambda_i| > 1$ we have that $x(t-n)$ gets amplified exponentially quickly as n increases, so small perturbations in state quickly overtake the network dynamics, i.e. the reservoir dynamical system behaves chaotically. This kind of sensitivity to initial conditions is undesirable in an echo state network. Normalizing by the eigenvalue of largest modulus thus ensures our reservoir represents a stable dynamical system.

6.2.2 Memory Capacity

One of the key properties of interest in studying reservoir networks is memory capacity, representing the network's ability to approximate a target function using a linear estimator with weights W_out on the set of reservoir states [18]. Dambre et.al. describe how to compute memory capacity thusly:

Suppose we have an ESN $X = \{x_i(t)\}_{i\in[n]} : t \in \mathbb{Z}\}$ parameterized by W_in, W, W_out. Let $\{u(t)\}_{t\in[-T;T]}$ be an independent and identically distributed input data vector of length $2T$, so that all dynamical behavior is due to correlations between nodes in the reservoir and not within the input data. Run the ESN from time $-T$ to 0 to eliminate transient behavior from the initialization of the reservoir nodes. We then consider the evolution of our ESN from time $t = 1$ to $t = T$.

Let $u^h(t) = \{u(t-h+1), ...u(t-1), u(t)\}$ denote u time lagged by h units from t and let $U^h = \{u^h(t) : t \in \mathbb{Z}\}$ denote the set of such time lagged vectors. Consider an arbitrary target function $z : U^h \to \mathbb{R}$ with $z(t) = z(u^h(t))$, which we seek to approximate using the linear estimator $\widehat{z(t)} = W_{out}(x(t))$.

Define the mean squared error between z and $\hat{z}$ for time series of length T thusly:

$$MSE_T(\hat{z}, z) = \frac{1}{T}\sum_{t=1}^{T}(\widehat{z(t)} - z(t))^2$$

Then, letting $< z^2 >_T = \frac{1}{T}\sum_{t\in[T]}(z(t))^2$ be the average squared value of z over the time frame $[T]$, define the capacity for time series length T, to be

$$C_T(X, \hat{z}, z) = \frac{< z^2 >_T - MSE_T(\hat{z}, z)}{< z^2 >_T}$$

Dambre et.al. prove that $C_T(X, \hat{z}, z) \in [0, 1]$ with C closer to one indicating that the ESN X is better at approximating z and closer to 0 indicating X is worse at approximating z. Recent work by Aceituno et. al. suggest that the memory capacity of echo state networks is highly correlated with the average moduli of the eigenspectrum of the adjacency matrix of the reservoir. This stands to reason given that echo state networks represent functions as linear combinations of their reservoir states, so the more linearly independent reservoir

vertices are the greater variety of functions it can represent. Aceituno et. al. prove increased correlations between neurons in a reservoir decreases the average modulus of the associated weighted adjacency matrix [17].

Thus, one of the first steps to verifying whether Cayley graphs have sufficiently "rich" structure to to approximate a wide array of functions is to examine the behavior of the average modulus of its eigenvalues.

Memory Capacity of Cayley Graphs on Finite Fields We use the Fourier characterization of the eigenspectrum of Cayley graphs (see Lemma 4.1) to prove an upper bound on the average eigenvalue modulus of fC_ng for each n.

Theorem 6.4 *Let fC_ng be a family of undirected Cayley graphs on the cyclic group $(Z_n, +)$ with generating set U_n Z_n and eigenvalues $f\lambda_k g_{i2Z_n} g$.*

Let $< j\lambda_n j >= \frac{1}{n}\sum_{k2Z_n} j\lambda_k j$. Then we have that $< j\lambda j > \quad jU_n j^{\frac{1}{2}}$

PROOF OF 6.4:

Recall by Lemma 4.1 that $8k\ 2\ Z_n$ we have that $\lambda_k = \widehat{\chi_{U_n}}(k)$ where $\chi_{U_n}^{\setminus}(k) = \sum_{x2f0;:::;n\ \ 1g} \chi_{U_n}(x) e^{\frac{i2\pi kx}{n}}$ is the kth Fourier character of the indicator function χ_{U_n} of the generating set U_n

The, by the Plancherel identity observe that

$$\sum_{k2Z_n} j\chi_{U_n}^{\setminus}(k)j^2 = jZ_n j \sum_{x2Z_n} j\chi_{U_n}(k)j^2 = n \sum_{x2Z_n} j\chi_{U_n}(k)j = njU_n j$$

Then by the Cauchy-Schwarz inequality observe

$$njU_n j = \sum_{k2Z_n} j\chi_{U_n}^{\setminus}(k)j^2 = \sum_{k2Z_n} j\lambda_k j^2 = \sum_{k2Z_n} j\lambda_k j^2 \sum_{k2Z_n} j\,\overline{p}\overline{1}\overline{=}\overline{n}j^2 \quad \left(\sum_{k2Z_n} \frac{j\lambda_k j}{\overline{p}\overline{=}\overline{n}}\right)^2$$

Thus, dividing by n and taking the square root of both sides we see

$$< j\lambda_n j >= \sum_{k2Z_n} \frac{j\lambda_k j}{n} \quad (jU_n j)^{\frac{1}{2}}$$

QED

Corollary 6.4.1 *Let $W_n = \frac{C_n}{jU_n j}$ be the normalized adjacency matrix of the Cayley graph $C_n = \big((Z_n, +), U_n\big)$ with maximal eigenvalue modulus 1. Let W_n have eigenspectrum $f\sigma_k g_{k2Z_n}$ and average eigenvalue modulus $\sum_{k2Z_n} \frac{jsigma_k j}{n}$.*

Then, if $jU_n j$ is an increasing function of n, we have that the limit of the average eigenvalue modulus as n approaches in nity is $\lim_{n\ !\ 1} < j\lambda_n j >= 0$.

PROOF:

Recall by the $jU_n j$-regularity of C_n we have that $\lambda_0 = max_{k2Z_n}\lambda_k max_{k2Z_n} j\lambda_k j = jU(n)j$, so W indeed correctly normalized. Futher, observe that if f

Let $f\lambda_k g_{k2Z}$ be the eigenspectrum of C_n and observe that$8k\ 2\ Z$ we have that $\sigma_k = \frac{k}{jU_n j}$.

Then by Theorem 6.4 we have that $\sum_{k2Z_n} \frac{j\ kj}{n} \quad jU_n j^{\frac{1}{2}}$, so

$$\sum_{k2Z_n} \frac{j\sigma_k j}{n} = \frac{1}{jU_n j} \sum_{k2Z_n} \frac{j\lambda_k j}{n} \quad jU_n j^{\overline{\frac{1}{2}}}$$

The result follows.

QED

Observing this we see that normalized Paley graphs are immediately ruled out as viable reservoir candidates on the basis of having vanishing average eigenvalue modulus, as is any dense family of Cayley graphs. Meanwhile, the most viable reservoir candidate among undirected families of Cayley graph is the family of

cycle graphs, previously established to be the undirected Cayley graph withe the smallest possible generating set. This is consistent with the favorable results obtained by Aceituno et. al. and Li et. al. for cycle graphs.

However, cycle graphs have eigenvalue set to $\{e^{\frac{i2\pi k}{n}} + e^{\frac{i2\pi k}{n}}\}_{k \in \mathbb{Z}_n} = \{2cos(\frac{2\pi k}{n})\}_{k \in \mathbb{Z}_n}$. Then observe that as $n \to +\infty$ we have $\lim_{n \to \infty} 2cos(\frac{2\pi}{n}) = 2cos(0) = 2 = |\lambda_0|$, so the second largest eigenvalue magnitude approaches the greatest eigenvalue modulus. Thus, cycle graphs are not remotely pseudorandom in a spectral sense, as they do not have sparse ϵ pseudorandomness for any $epsilon < 1$ after sufficiently large n.

Further, having small second greatest eigenvalue modulus in general isn't conducive to having large average eigenvalue modulus. This suggests that, counter-intuitively, the more pseudorandom a graph is, the less suitable a replacement for a random reservoir it is in an echo state network. In the context of Aceituno et. al., the conclusion that spectral pseudorandomness negatively affects the memory capacity of an echo state network is unavoidable

However, recall that although spectral pseudorandomness implies discrepancy pseudorandomness for all graphs, Conlon et. al. only proved the converse for Cayley graphs [12]. In fact, it is provably isn't true that sparse discrepancy pseudorandomness implies sparse spectral pseudorandomness in general [8]. This is intriguing because, as established above, sparse random graphs, the current standard choice of ESN reservoir, satisfy discrepancy pseudorandomness with high probability.

Therefore, if we broaden our search for pseudorandom reservoir candidates beyond Cayley graphs, we might be able to find suitable pseudorandom reservoir candidates with good memory capacity. A particular promising direction is to look into expander graphs, a variety of sparse graphs with very good 'edge expansion' (ie, discrepancy pseudorandomness) that can be deterministically constructed (and thus, perhaps, proven to have sub-random algorithmic entropy). Such graphs have already seen wide use in reducing the number of bits of randomness spent in random-walk algorithms, which further suggests their promise as substitutes for random reservoirs [5].

7 Current Work

Our ongoing work comprises two directions: first, we are working on generalizing Cayley graphs results to expander graphs from constructed from Cayley graphs using the Zig-Zag graph product. Such as such graphs have pseudorandom spectral properties, and are (unlike many of the Cayley graphs we looked at, such as Paley graphs) quite sparse. This makes them good reservoir candidates in Echo State Networks, and also facilitates the networks having low algorithmic entropy.

The second is to apply our pseudorandom reservoir candidates to echo state networks, which traditionally use binomial graph reservoirs to achieve a nonlinear embedding of an input time series in a higher dimensional space in order to forecast the time series [17]. We hope to use pseudorandom graphs to "save on bits of randomness" expended in the construction of reservoirs for echo state networks. As part of this endeavour, we are studying the memory capacity of echo state networks build using our pseudorandom graphs. We then plan to perform empirical tests on the ability of simple echo state networks with pseudorandom reservoirs to forecast the Mackey-Glass time series, single variable chaotic time series. Our end goal is to identify a pseudorandom graph with low algorithmic entropy that outperforms a binomial random graph of the same edge density in the chaotic time series forecasting task. Preliminary results indicate that undirected Cayley graphs on the cyclic group $(\mathbb{Z}_n, +)$ with nonconstant generating sets have vanishing memory capacity as $n \to 1$, ruling out Paley graphs and pseudorandom Cayley graphs on $(\mathbb{Z}_n, +)$ writ large as viable reservoir candidates. However, past research has shown that cycle graphs, the undirected Cayley graph on $(\mathbb{Z}_n, +)$ with the smallest possible generating set, make remarkably good reservoirs despite their lack of pseudorandom structure. The way expander graphs combine pseudorandomness with sparseness suggests they might make excellent reservoir candidates. We plan to investigate these next, in order to more thoroughly examining the relationship between the pseudorandomness and memory capacity of a graph.

References

[1] Zenil, Héctor, Narsis A. Kiani, and Jesper Tegnér. 2016. "Methods of Information Theory and Algorithmic Complexity for Network Biology." Seminars in Cell and Developmental Biology 51 (March): 32–43.

https://doi.org/10.1016/j.semcdb.2016.01.011.

[2] Mikołaj Morzy, Tomasz Kajdanowicz, and Przemysław Kazienko, "On Measuring the Complexity of Networks: Kolmogorov Complexity versus Entropy," Complexity 2017 (January 1, 2017): 1–12, https://doi.org/10.1155/2017/3250301.

[3] Zenil, Kiani, and Tegnér, 2017. "Low-Algorithmic-Complexity Entropy-Deceiving Graphs." Physical Review 96 (1). https://doi.org/10.1103/physreve.96.012308.

[4] Krivelevich, Michael, and Benny Sudakov. 2006. "Pseudo-Random Graphs." In Bolyai Society Mathematical Studies, 199–262. https://doi.org/10.1007/978-3-540-32439-3_10.

[5] Trevisan, Luca. 2017. Lecture Notes on Graph Partitioning, Expanders and Spectral Methods.

[6] Mowshowitz, Abbe, and Matthias Dehmer. 2012. "Entropy and the Complexity of Graphs Revisited." Entropy 14 (3): 559–70. https://doi.org/10.3390/e14030559.

[7] Li, Ming, and Paul Vitányi. 2008. An Introduction to Kolmogorov Complexity and Its Applications. Texts in Computer Science. https://doi.org/10.1007/978-3-030-11298-1.

[8] Zhao, Yufei. Graph Theory and Additive Combinatorics. Fall 2019. Massachusetts Institute of Technology: MIT OpenCouseWare, https://ocw.mit.edu/

[9] Trachtenberg, Ari. 2018. "Empirical Kolmogorov Complexity." IEEE Conference Publication — IEEE Xplore, February. https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=arnumber=8503120.

[10] Amigó, José M., Janusz Szczepański, Eligiusz Wajnryb, and Maria V. Sanchez-Vives. 2004. "Estimating the Entropy Rate of Spike Trains via Lempel-Ziv Complexity." Neural Computation 16 (4): 717–36. https://doi.org/10.1162/089976604322860677.

[11] Chung, Fan, Ronald L. Graham, and R. Wilson. 1989. "Quasi-random Graphs." Combinatorica 9 (4): 345–62. https://doi.org/10.1007/bf02125347.

[12] Conlon, David, Jacob Fox, and Yufei Zhao. 2014. "Extremal Results in Sparse Pseudorandom Graphs." Advances in Mathematics (New York. 1965) 256 (May): 206–90. https://doi.org/10.1016/j.aim.2013.12.004.

[13] Horita, Augusto Y., Ricardo Bonna, Denis S. Loubach, Ingo Sander, and Ingemar Söderquist. 2019. "Lempel-Ziv-Markov Chain Algorithm Modeling Using Models of Computation and ForSyDe." Linköping Electronic Conference Proceedings (Print), October. https://doi.org/10.3384/ecp19162017.

[14] Pack, Svetlana, Anuraag Kumar, Joshua Iosevich, Alhussein Khalil, Azita Mayeli, and Alex Iosevich. 2023. "Graph LZMA Complexity Empirical Validation." Software. https://colab.research.google.com/drive/1Zn68SkA_T_brFgZnd-zN7dtZhWA7bMiY?usp=sharing.

[15] Shalev-Shwartz, Shai, and Shai Ben-David. 2014. Understanding Machine Learning. https://doi.org/10.1017/cbo9781107298019.

[16] Jaeger, Herbert. 2001. "The 'echo state' approach to analysing and training recurrent neural networks – with an Erratum note." GMD Report 148.

[17] Aceituno, Pau Vilimelis, Gang Yan, and Yang-Yu Liu. 2020. "Tailoring Echo State Networks for Optimal Learning." iScience (Cambridge) 23 (9): 101440. https://doi.org/10.1016/j.isci.2020.101440.

[18] Dambre, Joni, David Verstraeten, Benjamin Schrauwen, and Serge Massar. 2012. "Information Processing Capacity of Dynamical Systems." Scientific Reports 2 (1). https://doi.org/10.1038/srep00514.

[19] Li, Boyu, Robert S Fong, and Peter Tiňo. 2023. "SIMPLE CYCLE RESERVOIRS ARE UNIVERSAL." arXiv, August. https://doi.org/10.48550/arXiv.2308.10793.