Entropy and mixing for $\mathbb{Z}^d$ SFTs

Ronnie Pavlov

University of Denver
www.math.du.edu/∼rpavlov

1st School on Dynamical Systems and Computation (DySyCo)
CMM, Santiago, Chile

Topological entropy

- Recall that for a $\mathbb{Z}^d$ SFT X, the **topological entropy** of X is

Topological entropy

- Recall that for a $\mathbb{Z}^d$ SFT X, the **topological entropy** of X is $h(X) := \lim_{n\to\infty} \frac{\log |L(X) \cap A^{\{1,\ldots,n\}^d}|}{n^d}$

Topological entropy

- Recall that for a $\mathbb{Z}^d$ SFT X, the **topological entropy** of X is
$$h(X) := \lim_{n\to\infty} \frac{\log |L(X) \cap A^{\{1,\dots,n\}^d}|}{n^d}$$
- When $d = 1$, there is a very simple algorithm for computing $h(X)$

Topological entropy

- Recall that for a $\mathbb{Z}^d$ SFT X, the **topological entropy** of X is
  $h(X) := \lim_{n \to \infty} \frac{\log |L(X) \cap A^{\{1,\ldots,n\}^d}|}{n^d}$
- When $d = 1$, there is a very simple algorithm for computing $h(X)$
- $h(X) = \log \lambda$ for λ the largest eigenvalue of an easy-to-define matrix

Topological entropy

- Recall that for a $\mathbb{Z}^d$ SFT X, the **topological entropy** of X is
$$h(X) := \lim_{n\to\infty} \frac{\log |L(X) \cap A^{\{1,\dots,n\}^d}|}{n^d}$$
- When $d = 1$, there is a very simple algorithm for computing $h(X)$
- $h(X) = \log \lambda$ for λ the largest eigenvalue of an easy-to-define matrix
- Bad news: for $d > 1$, there is in general no known closed form for entropy

Topological entropy

- Recall that for a $\mathbb{Z}^d$ SFT X, the **topological entropy** of X is

  $$h(X) := \lim_{n\to\infty} \frac{\log |L(X) \cap A^{\{1,\dots,n\}^d}|}{n^d}$$

- When $d = 1$, there is a very simple algorithm for computing $h(X)$
- $h(X) = \log \lambda$ for λ the largest eigenvalue of an easy-to-define matrix
- Bad news: for $d > 1$, there is in general no known closed form for entropy
- Only in a few specific cases are there clever arguments to find a closed form

Topological entropy

- Recall that for a $\mathbb{Z}^d$ SFT X, the **topological entropy** of X is
  $$h(X) := \lim_{n\to\infty} \frac{\log |L(X) \cap A^{\{1,\dots,n\}^d}|}{n^d}$$
- When $d = 1$, there is a very simple algorithm for computing $h(X)$
- $h(X) = \log \lambda$ for λ the largest eigenvalue of an easy-to-define matrix
- Bad news: for $d > 1$, there is in general no known closed form for entropy
- Only in a few specific cases are there clever arguments to find a closed form
  - Examples: domino tiling, square ice, hard hexagon, group shifts

Topological entropy

- Recall that for a $\mathbb{Z}^d$ SFT X, the **topological entropy** of X is

$$h(X) := \lim_{n\to\infty} \frac{\log |L(X) \cap A^{\{1,\ldots,n\}^d}|}{n^d}$$

- When $d = 1$, there is a very simple algorithm for computing $h(X)$
- $h(X) = \log \lambda$ for λ the largest eigenvalue of an easy-to-define matrix
- Bad news: for $d > 1$, there is in general no known closed form for entropy
- Only in a few specific cases are there clever arguments to find a closed form
  - Examples: domino tiling, square ice, hard hexagon, group shifts
- Can we say anything at all about these numbers?

Topological entropy

- Difficulty: for $d > 1$, the question of whether or not a pattern is in $L(X)$ is algorithmically undecidable!

Topological entropy

- Difficulty: for $d > 1$, the question of whether or not a pattern is in $L(X)$ is algorithmically undecidable!
- How does one even count patterns in $L(X) \cap A^{\{1,\ldots,n\}^d}$?

Topological entropy

- Difficulty: for $d > 1$, the question of whether or not a pattern is in $L(X)$ is algorithmically undecidable!
- How does one even count patterns in $L(X) \cap A^{\{1,\ldots,n\}^d}$?
- A pattern is **locally admissible** if it contains no patterns from forbidden list $\mathcal{F}$ (patterns from $L(X)$ are called **globally admissible**)

Topological entropy

- Difficulty: for $d > 1$, the question of whether or not a pattern is in $L(X)$ is algorithmically undecidable!
- How does one even count patterns in $L(X) \cap A^{\{1,\ldots,n\}^d}$?
- A pattern is **locally admissible** if it contains no patterns from forbidden list $\mathcal{F}$ (patterns from $L(X)$ are called **globally admissible**)
- The question of whether or not a pattern is locally admissible is decidable

Topological entropy

- Difficulty: for $d > 1$, the question of whether or not a pattern is in $L(X)$ is algorithmically undecidable!
- How does one even count patterns in $L(X) \cap A^{\{1,\ldots,n\}^d}$?
- A pattern is **locally admissible** if it contains no patterns from forbidden list $\mathcal{F}$ (patterns from $L(X)$ are called **globally admissible**)
- The question of whether or not a pattern is locally admissible is decidable
- **Theorem:** (Friedland) Entropy can also be computed by counting LOCALLY admissible patterns rather than globally admissible ones! (PROVE)

Topological entropy

- In other words, $h(X)$ is also the limit of the sequence $(\log \#$ locally admissible patterns with shape $\{1, \ldots, n\}^d|)/n^d$

Topological entropy

- In other words, $h(X)$ is also the limit of the sequence $(\log \#$ locally admissible patterns with shape $\{1, \dots, n\}^d|)/n^d$
- A simple subadditivity argument shows that this sequence approaches $h(X)$ from above (PROVE for $d = 1$)

Topological entropy

- In other words, $h(X)$ is also the limit of the sequence (log # locally admissible patterns with shape $\{1, \ldots, n\}^d|)/n^d$
- A simple subadditivity argument shows that this sequence approaches $h(X)$ from above (PROVE for $d = 1$)
- For any $\mathbb{Z}^d$ SFT X, there exists a computer program which generates a sequence of approximations $\frac{p_n}{q_n} \in \mathbb{Q}$ approaching $h(X)$ from above

Topological entropy

- In other words, $h(X)$ is also the limit of the sequence $(\log \#$ locally admissible patterns with shape $\{1, \dots, n\}^d|)/n^d$
- A simple subadditivity argument shows that this sequence approaches $h(X)$ from above (PROVE for $d = 1$)
- For any $\mathbb{Z}^d$ SFT X, there exists a computer program which generates a sequence of approximations $\frac{p_n}{q_n} \in \mathbb{Q}$ approaching $h(X)$ from above
- Such numbers are called **right recursively enumerable**

Topological entropy

- In other words, $h(X)$ is also the limit of the sequence $(\log \#$ locally admissible patterns with shape $\{1, \dots, n\}^d|)/n^d$
- A simple subadditivity argument shows that this sequence approaches $h(X)$ from above (PROVE for $d = 1$)
- For any $\mathbb{Z}^d$ SFT X, there exists a computer program which generates a sequence of approximations $\frac{p_n}{q_n} \in \mathbb{Q}$ approaching $h(X)$ from above
- Such numbers are called **right recursively enumerable**
- Amazingly, the converse is also true; any right recursively enumerable number is also the entropy of a $\mathbb{Z}^d$ SFT!

Topological entropy

- In other words, $h(X)$ is also the limit of the sequence $(\log \#$ locally admissible patterns with shape $\{1,\ldots,n\}^d|)/n^d$
- A simple subadditivity argument shows that this sequence approaches $h(X)$ from above (PROVE for $d = 1$)
- For any $\mathbb{Z}^d$ SFT X, there exists a computer program which generates a sequence of approximations $\frac{p_n}{q_n} \in \mathbb{Q}$ approaching $h(X)$ from above
- Such numbers are called **right recursively enumerable**
- Amazingly, the converse is also true; any right recursively enumerable number is also the entropy of a $\mathbb{Z}^d$ SFT!
    - You'll learn more about this in Emmanuel's talks

Topological entropy

- In other words, $h(X)$ is also the limit of the sequence (log # locally admissible patterns with shape $\{1, \ldots, n\}^d|)/n^d$
- A simple subadditivity argument shows that this sequence approaches $h(X)$ from above (PROVE for $d = 1$)
- For any $\mathbb{Z}^d$ SFT X, there exists a computer program which generates a sequence of approximations $\frac{p_n}{q_n} \in \mathbb{Q}$ approaching $h(X)$ from above
- Such numbers are called **right recursively enumerable**
- Amazingly, the converse is also true; any right recursively enumerable number is also the entropy of a $\mathbb{Z}^d$ SFT!
  - You'll learn more about this in Emmanuel's talks
- An example of utility of mixing conditions: uniform mixing conditions imply better computability properties

Topological entropy

- Consider $\mathcal{H}$, the $\mathbb{Z}^2$ hard square shift

Topological entropy

- Consider $\mathcal{H}$, the $\mathbb{Z}^2$ hard square shift
- There exists a computer program generating approximations $\frac{p_n}{q_n} \in \mathbb{Q}$ to $h(\mathcal{H})$ from above AND below (PROVE)

Topological entropy

- Consider $\mathcal{H}$, the $\mathbb{Z}^2$ hard square shift
- There exists a computer program generating approximations $\frac{p_n}{q_n} \in \mathbb{Q}$ to $h(\mathcal{H})$ from above AND below (PROVE)
- Such numbers are called **computable**

Topological entropy

- Consider $\mathcal{H}$, the $\mathbb{Z}^2$ hard square shift
- There exists a computer program generating approximations $\frac{p_n}{q_n} \in \mathbb{Q}$ to $h(\mathcal{H})$ from above AND below (PROVE)
- Such numbers are called **computable**
- **Theorem:** (Hochman-Meyerovitch) Strong irreducibility implies computability of $h(X)$ for any d

Topological entropy

- Consider $\mathcal{H}$, the $\mathbb{Z}^2$ hard square shift
- There exists a computer program generating approximations $\frac{p_n}{q_n} \in \mathbb{Q}$ to $h(\mathcal{H})$ from above AND below (PROVE)
- Such numbers are called **computable**
- **Theorem:** (Hochman-Meyerovitch) Strong irreducibility implies computability of $h(X)$ for any d
- **Theorem:** (Pavlov-Schraudner) Block gluing implies computability for $d = 2$, with bounds on the rate ($e^{O(n^2)}$ steps yields upper and lower approximations within tolerance $\frac{1}{n}$) (PROVE)

Topological entropy

- Consider $\mathcal{H}$, the $\mathbb{Z}^2$ hard square shift
- There exists a computer program generating approximations $\frac{p_n}{q_n} \in \mathbb{Q}$ to $h(\mathcal{H})$ from above AND below (PROVE)
- Such numbers are called **computable**
- **Theorem:** (Hochman-Meyerovitch) Strong irreducibility implies computability of $h(X)$ for any d
- **Theorem:** (Pavlov-Schraudner) Block gluing implies computability for $d = 2$, with bounds on the rate ($e^{O(n^2)}$ steps yields upper and lower approximations within tolerance $\frac{1}{n}$) (PROVE)
- Unknown if block gluing implies computability for $d > 2$

Realization of entropies

- $d = 1$: set of entropies of SFTs is known (algebraic criterion)

Realization of entropies

- $d = 1$: set of entropies of SFTs is known (algebraic criterion)
- $d = 1$: set of entropies of mixing SFTs is known (algebraic criterion)

Realization of entropies

- $d = 1$: set of entropies of SFTs is known (algebraic criterion)
- $d = 1$: set of entropies of mixing SFTs is known (algebraic criterion)
- $d > 1$: set of entropies of SFTs is known (computability criterion)

Realization of entropies

- $d = 1$: set of entropies of SFTs is known (algebraic criterion)
- $d = 1$: set of entropies of mixing SFTs is known (algebraic criterion)
- $d > 1$: set of entropies of SFTs is known (computability criterion)
- $d > 1$: set of entropies of (uniformly) mixing SFTs is NOT known

Realization of entropies

- $d > 1$: set of entropies of (uniformly) mixing SFTs is NOT known

Realization of entropies

- $d > 1$: set of entropies of (uniformly) mixing SFTs is NOT known
  - Such entropies must be computable (only for $d = 2$ if block gluing)

Realization of entropies

- $d > 1$: set of entropies of (uniformly) mixing SFTs is NOT known
  - Such entropies must be computable (only for $d = 2$ if block gluing)
  - **Theorem:** (Pavlov-Schraudner) All computable numbers with "sufficiently quickly computable" approximations appear as entropies of block gluing $\mathbb{Z}^3$ SFTs

Realization of entropies

- $d > 1$: set of entropies of (uniformly) mixing SFTs is NOT known
  - Such entropies must be computable (only for $d = 2$ if block gluing)
  - **Theorem:** (Pavlov-Schraudner) All computable numbers with "sufficiently quickly computable" approximations appear as entropies of block gluing $\mathbb{Z}^3$ SFTs
  - Proof relies on NON-entropy minimality of block gluing SFTs

Realization of entropies

- $d > 1$: set of entropies of (uniformly) mixing SFTs is NOT known
  - Such entropies must be computable (only for $d = 2$ if block gluing)
  - **Theorem:** (Pavlov-Schraudner) All computable numbers with "sufficiently quickly computable" approximations appear as entropies of block gluing $\mathbb{Z}^3$ SFTs
  - Proof relies on NON-entropy minimality of block gluing SFTs
  - Possible to create a non-block gluing SFT with desired entropy and "upgrade" it to a block gluing SFT without increasing entropy

Realization of entropies

- $d > 1$: set of entropies of (uniformly) mixing SFTs is NOT known
    - Such entropies must be computable (only for $d = 2$ if block gluing)
    - **Theorem:** (Pavlov-Schraudner) All computable numbers with "sufficiently quickly computable" approximations appear as entropies of block gluing $\mathbb{Z}^3$ SFTs
    - Proof relies on NON-entropy minimality of block gluing SFTs
    - Possible to create a non-block gluing SFT with desired entropy and "upgrade" it to a block gluing SFT without increasing entropy
- Impossible for strongly irreducible SFTs, since they are entropy minimal

Realization of entropies

- $d > 1$: set of entropies of (uniformly) mixing SFTs is NOT known
  - Such entropies must be computable (only for $d = 2$ if block gluing)
  - **Theorem:** (Pavlov-Schraudner) All computable numbers with "sufficiently quickly computable" approximations appear as entropies of block gluing $\mathbb{Z}^3$ SFTs
  - Proof relies on NON-entropy minimality of block gluing SFTs
  - Possible to create a non-block gluing SFT with desired entropy and "upgrade" it to a block gluing SFT without increasing entropy
- Impossible for strongly irreducible SFTs, since they are entropy minimal
- There is no method to create strongly irreducible SFT with specific entropy

Computability of $h(\mathcal{H})$

- Recall that topological mixing conditions yielded $h(\mathcal{H})$ computable, and $e^{O(n^2)}$ steps yielded approximation within $\frac{1}{n}$

Computability of $h(\mathcal{H})$

- Recall that topological mixing conditions yielded $h(\mathcal{H})$ computable, and $e^{O(n^2)}$ steps yielded approximation within $\frac{1}{n}$
- In fact much more can be said: $n^{O(1)}$ steps yield approximation within $\frac{1}{n}$

Computability of $h(\mathcal{H})$

- Recall that topological mixing conditions yielded $h(\mathcal{H})$ computable, and $e^{O(n^2)}$ steps yielded approximation within $\frac{1}{n}$
- In fact much more can be said: $n^{O(1)}$ steps yield approximation within $\frac{1}{n}$
- This stronger result not provable with topological techniques

Computability of $h(\mathcal{H})$

- Recall that topological mixing conditions yielded $h(\mathcal{H})$ computable, and $e^{O(n^2)}$ steps yielded approximation within $\frac{1}{n}$
- In fact much more can be said: $n^{O(1)}$ steps yield approximation within $\frac{1}{n}$
- This stronger result not provable with topological techniques
- Key: using measure theory!

Computability of $h(\mathcal{H})$

- Recall that topological mixing conditions yielded $h(\mathcal{H})$ computable, and $e^{O(n^2)}$ steps yielded approximation within $\frac{1}{n}$
- In fact much more can be said: $n^{O(1)}$ steps yield approximation within $\frac{1}{n}$
- This stronger result not provable with topological techniques
- Key: using measure theory!
  - We'll learn more about this next time