Topological Entropy Bounds for Hyperbolic Plateaus of the Hénon Map

Rafael M. Frongillo*

September 11, 2012

Abstract

Combining two existing rigorous computational methods, for verifying hyperbolicity (Arai [1]) and for computing topological entropy bounds (Day et al. [4]), we prove lower bounds on topological entropy for 43 hyperbolic plateaus of the Hénon map. We also examine the 16 area-preserving plateaus studied by Arai and compare our results with related work. Along the way, we augment the entropy algorithms of Day et al. with routines to optimize the algorithmic parameters and simplify the resulting semi-conjugate subshift.

1 Introduction

Dynamical systems theory has seen the emergence of many rigorous computational methods in recent years. Such tools often extend the realm of provable theorems well beyond what is possible with chalk and blackboard. This is particularly true of the recent automated tools to compute topological entropy bounds [17, 4] and to prove hyperbolicity [1, 13, 16].

Let us briefly recall the relevant characteristics of these methods. Both techniques for proving entropy bounds first construct a subshift of finite type (SFT), whose topological entropy is easily computed and is a lower bound of that of the original system. Newhouse et al. compute rigorous approximations of stable and unstable manifolds of periodic orbits, and then construct a SFT using pieces of these manifolds; in this regard, their technique could be considered a rigorous version of the trellis method. Day, Frongillo, and Treviño (DFT) [4] construct a discrete multivalued map from a discretization of the phase space, and then apply discrete Conley index theory to prove a semi-conjugacy to a particular SFT. In certain settings, such as the one we study here, the DFT method is *completely* automated, meaning that after a simple initialization, no further manual input is required. It is unclear whether the method of Newhouse et al. shares this automation property.

---

*University of California at Berkeley

Hruska [13] developed one of the first automated methods for rigorously verifying hyperbolicity, based on the computation of cone fields. In contrast, Arai [1] employs a more indirect technique using the notion of quasi-hyperbolicity. His approach allows for more efficient computations than Hruska, but does not guarantee that the nonwandering set is not just a finite collection of periodic orbits, or even that it is nonempty.

In [1], Arai identifies several hyperbolic regions of the Hénon map:

$$f_{a,b}(x,y) = (a - x^2 + by, x) \tag{1}$$

These regions are dubbed *hyperbolic plateaus* because topological entropy is constant across any such region. Arai's technique does not reveal anything about these topological entropy values, however, and it is therefore natural to combine his computations with an automated method for proving topological entropy bounds.

In this paper we use the DFT method [4] to compute lower bounds for topological entropy of the hyperbolic plateaus of Hénon computed by Arai in [1]. The constant entropy on each plateau enables us to extend a lower bound computation from a single setting of parameter values to an entire region of the parameter space. Additionally, the full automation of the DFT method enables us to study a total of 58 parameter values with essentially the same manual effort as studying one. See Section 4 for details on the DFT method, as well as new techniques for improving the robustness of the algorithms and for simplifying the resulting SFT.

Theorems 5.1 and 5.3 summarize our results. To the author's knowledge, all of these rigorous lower bounds are the largest known for their corresponding parameters. We selected the parameter regions so that the bounds obtained might give a global picture of the entropy of the Hénon map as a function of the parameters; see Figure 1 for such a picture.

We also study several of the area-preserving Hénon maps in Section 5.1, which have been well-studied in the Physics community [12, 22, 5]. Recently, some precise rigorous results emerged as well [2]. We find that our rigorous lower bounds match or are very close to estimates given in previous work, and match the rigorous results exactly when applicable.

2 Background

We first review basic definitions related to symbolic dynamics, topological entropy, and hyperbolicity, in Sections 2.1 and 2.2. The DFT approach relies heavily on a combinatorial version of the discrete Conley index; we discuss the index in Section 2.3 and the combinatorial structures that relate it to our computational setting in 2.4.
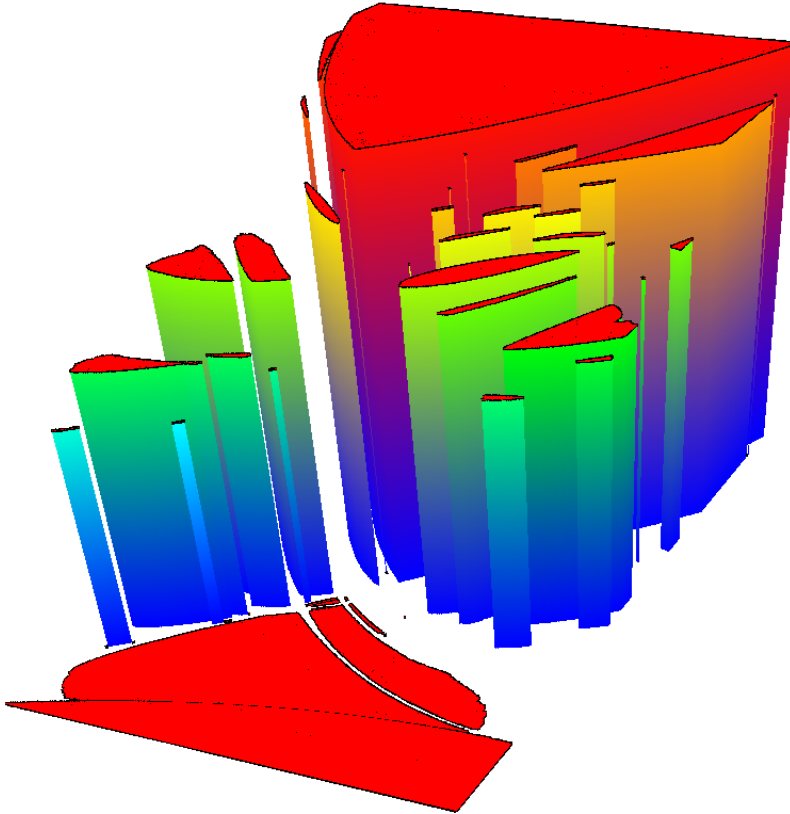
Figure 1: Rigorous lower bounds for topological entropy for the hyperbolic plateaus in Figure 4. The height of each plateau in the visualization is proportional to the entropy bound computed. See Theorem 5.1 or Figure 5 for the actual bounds.

2.1 Symbolic dynamics and topological entropy

Define a *symbol space* $X_n = \{0, \dots, n-1\}^{\mathbb{Z}}$ to be the set of all bi-infinite sequences on n symbols. It is well-known that X_n is a complete metric space. Let the *full n-shift* $\sigma : X_n \to X_n$ be the map acting on X_n by $(\sigma(x))_i = x_{i+1}$. Given a directed graph G on n nodes with $n \times n$ transition matrix A with $A_{i,j} \in \{0, 1\}$, we can define an induced symbol space $X_G \subset X_n$, where $x \in X_G$ if and only if for $x = (\dots, x_i, x_{i+1}, \dots)$, $A_{x_i, x_{i+1}} = 1$ for all i. That is, X_G consists of all sequences in X_n with transitions of σ allowed by the edges of G. Equipped with the corresponding shift map $\sigma_G : X_G \to X_G$, we call (X_G, σ_G) a *subshift of finite type*.

We use topological entropy to measure the relative complexity of different dynamical systems. If the topological entropy of a dynamical system f, denoted $h(f)$, is positive, we say that f is *chaotic*.

3

Definition 2.1 (Topological entropy [15])**.** *Let $f : X \to X$ be a continuous map with respect to a metric d. We say that a set $W \subseteq X$ is (n, ε)-separated under f if for any distinct $x, y \in W$ we have $d\big(f^j(x),\ f^j(y)\big) > \varepsilon$ for some $0 \le j < n$. The* topological entropy *of f is*

$$h(f) := \lim_{\varepsilon \to \infty} \limsup_{n \to \infty} \frac{log(s_f(n, \varepsilon))}{n}, \tag{2}$$

where $s_f(n, \varepsilon)$ denotes the maximum cardinality of an (n, ε)-separated set under f.

While topological entropy can be difficult to calculate in general, there is a simple formula for subshifts of finite type which is given in the following theorem. For a proof, see [15] or [20].

Theorem 2.2. *Let G be a directed graph with transition matrix A, and let (X_G, σ_G) be the corresponding subshift of finite type. Then the topological entropy of σ_G is $h(\sigma_G) = \log(sp(A))$, where $sp(A)$ denotes the spectral radius (maximum magnitude of an eigenvalue) of A.*

When studying a complex map f, it is sometimes useful to study a subsystem of f which can be precisely related to f via a semi-conjugacy.

Definition 2.3. *Let $f : X \to X$ and $g : Y \to Y$ be continuous maps. A* semi-conjugacy *from f to g is a continuous surjection $\phi : X \to Y$ with $\phi \circ f = g \circ \phi$. We say that f is* semi-conjugate *to g if there exists a semi-conjugacy from f to g. If additionally ϕ is a homeomorphism, then f and g are* conjugate.

Particularly relevant to our setting is the following result.

Theorem 2.4 ([20])**.** *Let f and g be continuous maps, and let ϕ be a semi-conjugacy from f to g. Then $h(f) \ge h(g)$.*

Note that if f and g are conjugate, Theorem 2.4 gives us $h(f) = h(g)$. In other words, topological entropy is invariant under conjugacy.

2.2 Hyperbolicity

To begin we define uniform hyperbolicity. Throughout the paper, we will refer to this property simply as hyperbolicity.

Definition 2.5 ([11])**.** *A map $f : X \to X$ is said to be (uniformly)* hyperbolic *if for every $x \in X$ the tangent space $T_x X$ for f is a direct sum of stable and unstable subspaces. More precisely, we have $T_x X = E^s(x) \oplus E^u(x)$, where $E^s(x)$ and $E^u(x)$ satisfy the following inequalities for some $C > 0$ and $0 < \lambda < 1$, and for all $n \in \mathbb{N}$:*

    1. $\|Df^n(v)\| \le C\lambda^n \|v\|$ for all $v \in E^s(x)$.

    2. $\|Df^{-n}(v)\| \le C\lambda^n \|v\|$ for all $v \in E^u(x)$.

4

This structure can be thought of as a generalization of the structure of the Smale horseshoe, namely that there are invariant directions, and there is uniform contraction and expansion in the stable and unstable directions, respectively. Some useful properties of hyperbolic systems are discussed below, but for more details see [20] and [11].

An important property of hyperbolic maps is that they are structurally stable [20], which implies that all maps in the same hyperbolic region are conjugate. Thus, by Theorem 2.4, the topological entropy is constant within such a region. For this reason, we will henceforth call these regions *hyperbolic plateaus*.

Hyperbolicity often makes it easier to identify interesting dynamics, but it is important to note that sometimes a system can be "vacuously" hyperbolic, in the sense that it is hyperbolic but there is no recurrent behavior. A helpful concept in this context is the *nonwandering set*.

Definition 2.6 ([20])**.** *The* nonwandering set *of a map f is the set of points x for which every neighborhood U of x has $f^n(U) \cap U \neq \emptyset$ for some $n \geq 1$.*

2.3 The discrete Conley index

Conley index theory is a topological tool which is a refinement of Morse theory for gradient-like flows. We introduce here a version of the Conley index which has been developed for discrete-time systems. Let $f : M \to M$ be a continuous map, where M is a smooth, orientable manifold.

Definition 2.7. *A compact set $K \subset M$ is an* isolating neighborhood *if*

$$Inv(K, f) \subset Int(K),$$

where $Inv(K, f)$ denotes the maximal invariant set of K and $Int(K)$ denotes the interior of K. A set S is an isolated invariant set *if $S = Inv(K, f)$ for some isolating neighborhood K.*

Definition 2.8 ([19])**.** *Let S be an isolated invariant set for f. Then $P = (P_1, P_0)$ is an* index pair *for S if*

1. *$\overline{P_1 \backslash P_0}$ is an isolating neighborhood for S.*

2. *The induced map*

$$f_P(x) = \begin{cases} f(x) & \text{if } x, \, f(x) \in P_1 \backslash P_0 \text{ ,} \\ [P_0] & \text{otherwise.} \end{cases}$$

defined on the pointed space $(P_1 \backslash P_0, [P_0])$ is continuous.

Definition 2.9 ([23])**.** *Let G, H be abelian groups and $\varphi : G \to G$, $\psi : H \to H$ homomorphisms. Then φ and ψ are* shift equivalent *if there exist homomorphisms $r : G \to H$ and $s : H \to G$ and some constant $k \in \mathbb{N}$ such that*

$$r \circ \varphi = \psi \circ r, \quad s \circ \psi = \varphi \circ s, \quad r \circ s = \psi^k, \quad and \quad s \circ r = \varphi^k.$$

Shift equivalence defines an equivalence relation; we denote by $[\varphi]_s$ the equivalence class of homomorphisms which are shift equivalent to φ.

Definition 2.10 ([7])**.** *Let $P = (P_1, P_0)$ be an index pair for an isolated invariant set $S = Inv(\overline{P_1 \backslash P_0}, f)$ and let $f_{P*} : H_*(P_1, P_0; \mathbb{Z}) \to H_*(P_1, P_0; \mathbb{Z})$ be the map induced by f_P on the relative homology groups $H_*(P_1, P_0; \mathbb{Z})$. The* Conley index *of S is the shift equivalence class $[f_{P*}]_s$ of f_{P*}.*

Intuitively, two maps f_{P*} and $g_{\bar{P}*}$ are shift equivalent if and only if they have the same assymptotic behavior. Note that since an index pair for an isolated invariant set is not unique, the Conley index of an isolated invariant set does not depend on the choice of index pair.

The key property of the Conley index is that it says something about the dynamics of f. In particular, if the index is nontrivial, so are the dynamics. This is the so-called Ważewski property:

Theorem 2.11. *If $[f_{P*}]_s \neq [0]_s$, then $S \neq \varnothing$.*

Since similar assymptotic behavior relates two different maps in the same shift equivalence class, it is sufficient then to have a map f_{P*} not be nilpotent in order to have a non-empty isolated invariant set. In practice, non-nilpotency can be verified by taking iterates of a representative of $[f_{P*}]_s$ until non-nilpotent behavior is detected.

Corollary 2.12. *Let $K \subset M$ be the finite union of disjoint, compact sets $K_1, \ldots, K_m$ and let $S = Inv(K, f)$. Let $S' = Inv(K_1, f_{K_m} \circ \cdots \circ f_{K_1}) \subset S$ where f_{K_i} denotes the restriction of f to K_i. If*

$$[(f_{K_m} \circ \cdots \circ f_{K_1})_*]_s \neq [0]_s,$$

then S' is nonempty. Moreover, there is a point in S whose trajectory visits the sets K_i in such order.

This corollary is used heavily to in our computational proofs of symbolic dynamics. What makes the implementation possible is an efficiently-computable sufficient condition for non-nilpotency. For more details on the implementation, see [4].

2.4 Combinatorial structures

The concepts from discrete Conley index theory from the previous section have analogs in a combinatorial setting which is much more natural from a computational perspective.

Definition 2.13. *A* multivalued map *$F : X \rightrightarrows X$ is a map from X to its power set, so that $F(x) \subset X$. If F is acyclic and we have $f(x) \in F(x)$ for some continuous, single-valued map f, then F is an* enclosure *of f.*

One reason multivalued maps and enclosures are used in our computations is that, if constructed properly, they enable rigorous results. Moreover, if F is an enclosure of f and (P_0, P_1) is an index pair for F (as we will define below), then it is an index pair for f. It follows that the information contained in the Conley index of F translates back to the dynamics of f.

We begin by setting up a grid $\mathcal{G}$ on M, which is a compact subset of the n dimentional manifold M composed of finitely many elements $\mathcal{B}_i$. Each element is a cubical complex, hence a compact set, and it is essentially an element of a finite partition of a compact subset of M. In practice, all elements of the grid are rectangles represented as products of intervals (viewed in some nice coordinate chart); that is, for $\mathcal{B}_i \in \mathcal{G}$, $\mathcal{B}_i = \prod_{k=1}^n [x_k^i, y_k^i]$. We refer to each element of $\mathcal{G}$ as a *box*. In our setting, for each k the interval widths $y_k^i - x_k^i$ are the same for all i, meaning the dimensions of all boxes are the same, but this is in no way necessary. For a collection of boxes $\mathcal{K} \subset \mathcal{G}$, we denote by $|\mathcal{K}|$ its *topological realization*, that is, its corresponding subset of M. From now on we will use caligraphy capital letters to denote collections of boxes in $\mathcal{G}$ and by regular capital letters we will denote their topological realization, e.g., $|\mathcal{B}_i| = B_i$.

In our setting, we create a grid by selecting one box $\mathcal{B}$ such that $|\mathcal{B}|$ encloses the entire area we wish to study. Then we subdivide $\mathcal{B}$ evenly d times in each coordinate direction in order to increase the resolution at which the dynamics are studied. The integer d will be refered to as the *depth*. Thus working at depth d gives us a maximum of 2^{dn} (where $n = \dim M$) boxes with which to work, each coordinate of size 2^{-d} relative to the original size of the box $\mathcal{B}$.

Definition 2.14. *A* combinatorial enclosure *of f is a multivalued map $\mathcal{F} : \mathcal{G} \rightrightarrows \mathcal{G}$ defined by*

$$\mathcal{F}(\mathcal{B}) = \{\mathcal{B}' \in \mathcal{G} : |\mathcal{B}'| \cap F(B) \neq \varnothing\},$$

where F is an enclosure of f.

We construct combinatorial enclosures as follows. Given any $\mathcal{B} \in \mathcal{G}$, we define $F(x)$, $x \in B = |\mathcal{B}|$, as the image of B using a rigorous enclosure for the map f. This rigorous enclosures is obtained by keeping track of the error terms in the computations of the image of a box and ensuring that the true image $f(B)$ is contained in $|\mathcal{F}(\mathcal{B})|$. In particular, we the interval arithmetic library Intlab [21] to provide this guarantee. Note that immediately $|\mathcal{F}|$ becomes an enclosure of f.

Definition 2.15. *A* combinatorial trajectory *of a combinatorial enclosure $\mathcal{F}$ through $\mathcal{B} \in \mathcal{G}$ is a bi-infinite sequence $\gamma_G = (\ldots, \mathcal{B}_{-1}, \mathcal{B}_0, \mathcal{B}_1, \ldots)$ with $\mathcal{B}_0 = \mathcal{B}$, $\mathcal{B}_n \in \mathcal{G}$, and $\mathcal{B}_{n+1} \in \mathcal{F}(\mathcal{B}_n)$ for all $n \in \mathbb{Z}$.*

The definitions which follow are by now standard in the computational Conley index literature. We will state definitions and refer the reader to [4] for the algorithms which construct the objects defined.

Definition 2.16. *The* combinatorial invariant set *in $\mathcal{N} \subset \mathcal{G}$ for a combinatorial enclosure $\mathcal{F}$ is*

$$Inv(\mathcal{N}, \mathcal{F}) = \{\mathcal{B} \in \mathcal{G} : \text{ there exits a trajectory } \gamma_G \subset \mathcal{N}\}.$$

Definition 2.17. *The* combinatorial neighborhood *or* one-box beighborhood *of* $\mathcal{B} \subset \mathcal{G}$ *is*

$$o(\mathcal{B}) = \{\mathcal{B}' \in \mathcal{G} : |\mathcal{B}'| \cap |\mathcal{B}| \neq \varnothing\}.$$

Definition 2.18. *If*

$$o(Inv(\mathcal{N}, \mathcal{F})) \subset \mathcal{N}$$

then $\mathcal{N} \subset \mathcal{G}$ *is a* combinatorial isolating neighborhood *for* $\mathcal{F}$.

Definition 2.19. *A pair* $\mathcal{P} = (\mathcal{P}_1, \mathcal{P}_0) \subset \mathcal{G}$ *is a* combinatorial index pair *for the combinatorial enclosure* $\mathcal{F}$ *if its topological realization* $P_i = |\mathcal{P}_i|$ *is an index pair for any map* f *for which* $\mathcal{F}$ *is an enclosure.*

We have now made all definitions necessary to define the discrete Conley index. Note however that computing the induced map on homology is a difficult task in and of itself. For a thorough treatment of computational homology, dealing with this and other applications, see [14]. For our computations, we use the *homcubes* package, part of the software package *CHomP* [10], which computes the necessary maps on homology to define the Conley index.

3   Simplifying Subshifts

Given a subshift of finite type (X_G, σ_G) for a graph G, it is often of interest to know whether there is a graph H on fewer vertices such that (X_G, σ_G) and (X_H, σ_H) are conjugate. To this end, we recall the notion of strong shift equivalence, which is stronger than shift equivalence from Definition 2.9.

Definition 3.1 (Strong shift equivalence). *Let A and B be matrices. An* elementary shift equivalence *between A and B is a pair (R, S) such that*

$$A = RS \ and \ B = SR. \tag{3}$$

In this case, we write $(R, S) : A \rightarrow B$. *If there is a sequence of such elementary shift equivalences* $(R_i, S_i) : A_{i-1} \rightarrow A_i$, $1 \leq i \leq k$, *we say that A_0 and A_k are* strongly shift equivalent.

This notion is useful because of the following result due to R. F. Williams relating shift equivalence to symbolic dynamics.

Theorem 3.2 ([24]). *For directed graphs G and H, the corresponding subshifts* (X_G, σ_G) *and* (X_H, σ_H) *are conjugate if and only if the transition matrices of G and H are strongly shift equivalent.*

Theorem 3.2 allows us to prove that two subshifts are conjugate by a series of simple matrix computations. Finding matrices that give a strong shift equivalence, however, can be a very difficult problem. Two methods of finding such equivalences are given in [15]: state splitting, where a single vertex is split into two, or state amalgamation, where two vertices are combined into one. In graph-theoretic terms, amalgamating two vertices is equivalent to contracting

them, or contracting the edge between them. In general obtaining the smallest element of a strong shift equivalence class may involve both splittings and amalgamations. We instead focus on the simpler problem of obtaining H only by amalgamating vertices in G. This also has the advantage of producing a matrix that is more useful for our needs in this paper (see Section 4).

Let A be the binary $n \times n$ transition matrix for G. The following two conditions, adapted from [15], will allow us to amalgamate vertices i and j.

$$\textbf{Forward Condition:} \quad A\vec{e}_i = A\vec{e}_j \text{ and } (\vec{e}_i^\top A) \cdot (\vec{e}_j^\top A) = 0 \qquad (4)$$

$$\textbf{Backward Condition:} \quad \vec{e}_i^\top A = \vec{e}_j^\top A \text{ and } (A\vec{e}_i) \cdot (A\vec{e}_j) = 0 \qquad (5)$$

Here $\vec{e}_i$ denotes the column vector with a 1 in position i and zeros elsewhere. From a graph-theoretic or dynamical systems point of view, the forward condition says that i and j have the same image but disjoint preimages, and the backward condition says they have the same preimage but disjoint images. See Figure 2 for an example.
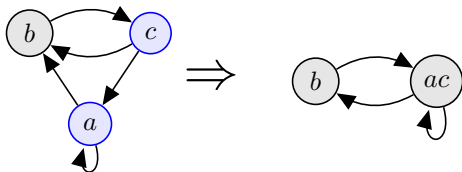

Figure 2: A forward amalgamation

Note that the backward condition for A is the same as the forward condition for $A^\top$. The following result allows us to reduce A to a smaller $n - 1$ by $n - 1$ matrix B if either of these conditions are satisfied for some pair of vertices. See e.g. [15, §2] for a proof.

Theorem 3.3. *If i and j satisfy the forward condition (4) or backward condition (5) for a transition matrix A, then there is an elementary shift equivalence from A to the matrix obtained by amalgamating i and j.*

By applying Theorem 3.3 repeatedly, as long as there exist i, j satisfying either contraction condition, one can reduce A to a much smaller representative of its strong shift equivalence class. The resulting matrix B at the end of this process corresponds to a subshift (X_H, σ_H) which is therefore conjugate to (X_G, σ_G).

For small enough matrices it is feasible to perform a simple brute-force search to find the smallest B which can be obtained from A via amalgamations, but we would like a more efficient algorithm for larger matrices. Unfortunately it is shown in [9] that it is NP-hard (computationally intractable) to find an ordering of amalgamations which yields the smallest representative. In light of this result, we use the procedure outlined in Algorithm 1, which is a essentially a randomized greedy algorithm performed many times. Taking the number of

---

Algorithm 1 `simplify_subshift`: Amalgamating a subshift of finite type

Input: subshift $T \in \{0,1\}^{n \times n}$, number of trials K
$T_{\min} \leftarrow T$
for k from 1 to K **do**
  $\pi \leftarrow$ `random_permutation`(n)
  $T_\pi \leftarrow T(\pi, \pi)$ {relabel vertices}
  repeat
    `amalgamated` $\leftarrow$ `false`
    for $(i,j) \in E(T_\pi)$ ordered lexographically **do**
      if conditions (4) or (5) hold for $A = T_\pi$ **then**
        $T_\pi \leftarrow$ `amalgamate`(T_π, i, j)
        `amalgamated` $\leftarrow$ `true`
        break for
      end if
    end for
  until not `amalgamated`   {no further amalgamations}
  if `size`$(T_\pi) <$ `size`$(T_{\min})$ **then**
    $T_{\min} \leftarrow T_\pi$
  end if
end for
Output: $T_{\min}$

---

trials k to be about n^2 will typically give a good approximation factor, meaning that if m amalgamations are possible, the algorithm will perform roughly $m/3$ or $m/2$ amalgamations. It remains an open question whether an algorithm exists which has a provable approximation guarantee.

4 Techniques

Given a continuous map $f : X \to X$, we will apply discrete Conley index theory to compute subshifts of finite type to which f is semi-conjugate. This is the approach of the DFT method, which we describe first in Section 4.1. We also add a handful of new techniques and algorithms which we present in Sections 4.2 and 4.3. Finally, we discuss implementation and efficiency details in Section 4.4.

4.1 The DFT method

To obtain lower bounds on topological entropy, we compute a semi-conjugate symbolic dynamical system using the approach described in [4], which is based on the discrete Conley index. The technique consists of three main steps:

1. Discretize a compact subset of the phase space by constructing a grid $\mathcal{G}$ of boxes, and compute a rigorous enclosure for the dynamics on these boxes as described in Section 2.4.

2. Find a combinatorial index pair from these boxes and determine the Conley index for this pair using [18].

3. From the index, compute a subshift to which the original system is semi-conjugate.

This approach is very general, and in principle could be applied to systems of arbitrary dimension. A major benefit to using this method here, however, is that in our setting it is completely automated. As long as one knows roughly where in the phase space the invariant objects of interest are, one can simply plug in the parameters and compute. See Section 4.4 for more details on the implementation.

An advantage to studying hyperbolic parameters of Hénon map (1) for $|b| \leq 1$ is that the nonwandering set is disconnected. This follows from Plykin theory, as discussed in [20, §7.9], from which we know that any connected trapping region (a region N with $f(N) \subset \text{int}(N)$) of the attractor has at least three holes. Considering a disc covering such a hole, we see that since the image each hole must strictly cover another hole, an iterate of this disk must eventually expand, which contradicts the area-preserving or area-shrinking of the maps we are considering. Thus, all such trapping regions must be disconnected.

Since the nonwandering set is disconnected, we can bypass much of the complication in the second step of the DFT method, that of finding an index pair. This is because the invariant set will be naturally isolated; at a fine enough resolution, the collection of boxes that cover the invariant set will already be separated into disjoint regions.

4.2 Reducing large subshifts

When applying the DFT method to complex systems, the resulting semi-conjugate subshift (step 3 above) is often very large. In fact, the main result in [4] is shown via a subshift with 247 symbols. Such large subshifts can have limited practical value; they provide a topological entropy bound and information about the number of orbits of a given period, but these large subshifts often carry little *intuition* about the underlying structure of the dynamics. The question then becomes, how can we distill more useful information from these large subshifts to get a more intuitive understanding of the system?

The answer we propose here is to simplify the resulting subshift using amalgamations, as described in Section 3. Specifically, we add a final step to the DFT method, where we run the semi-conjugate subshift obtained through Algorithm 1. As we will see in Section 5, in practice this procedure can greatly reduce the number of symbols required to describe the system. In some cases this simplification reveals a simple underlying structure, such as connections between a handful of low-period orbits, which would otherwise be impossible to glean from the 200+ symbols of the original subshift. This simplification is also useful in comparing our results to previous work and conjectures; using Algorithm 1, one can attempt to amalgamate our computed subshift A to yield a target subshift B (where here we identify subshifts with their corresponding

11

(a) Aspect ratio: entropy vs. $\log(w_2/w_1)$
Depths 9 (blue) and 10 (green)

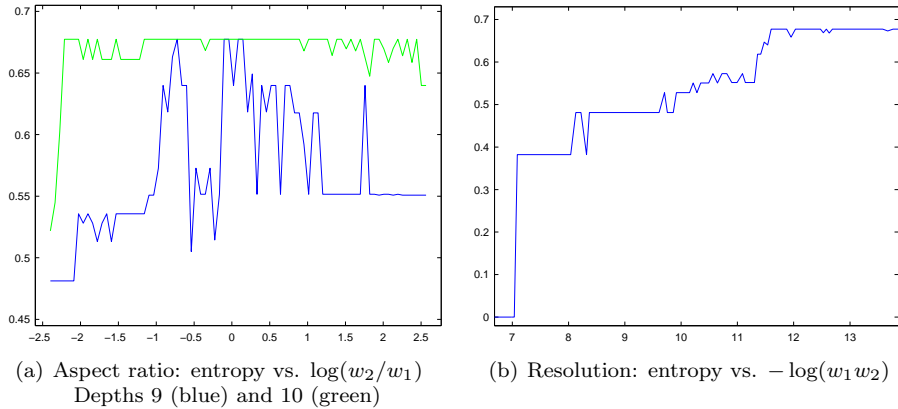(b) Resolution: entropy vs. $-\log(w_1 w_2)$

Figure 3: Sensitivity of the entropy bound with respect to the aspect ratio (a) and the area (b) of the boxes. The map used was Hénon (1) with parameters (5.685974, -1), or plateau 14 from Section 5.1.

transition matrices). We apply this technique in Section 5.1; see Figure 8 in particular.

As a final note, recall that the subshift generated by the DFT method has a useful geometric interpretation: from [4] we know that we can associate a region N_i of the phase space to each symbol s_i of the subshift A, such that any trajectory $(\ldots, s_{-2}, s_{-1}, s_0, s_1, s_2, \ldots)$ in A corresponds to a trajectory in the original system through the regions $(\ldots, N_{-2}, N_{-1}, N_0, N_1, N_2, \ldots)$. Fortunately, amalgamation (and thus Algorithm 1) preserves this property in the following sense. If A' is derived from A by a sequence of amalgamations, each symbol s_i' of A' can be expressed as an amalgamation of symbols of A. Taking N_i' to be the union of the regions corresponding to these symbols, it is easy to see that the same trajectory property holds for A'.

4.3   Robustness and scaling

A natural concern for any approach which involves discretizing the phase space is the robustness of the method with respect to the choice of discretization. In our setting, this discretization is determined by the resolution (depth), aspect ratio[1], and the precise placement of the grid $\mathcal{G}$.

Ideally, slight changes in these scaling parameters would have at most minor effects on the resulting subshift or entropy bound, but there are several reasons why this is unreasonable to expect. Perhaps most obvious is that a subshift of finite type is a discrete object, and one cannot expect any sort of continuity in the scaling parameters when the output itself is discrete. More to the point, the combinatorial isolation of the index pair depends on the topological properties of rigorous numerical bounds on the images of boxes, which can be very sensitive

---

[1]We will use this term to refer to the shape of the boxes in higher dimensions as well.

to the precise grid parameters. A simple example is in achieving disjoint regions; when trying to separate regions A and B of the phase space with $\text{dist}(A, B) = d$, using a grid whose boxes $\mathcal{B}_i$ have width $y_k^i - x_k^i = 4d$ in each dimension k, great care must be taken in placing the grid, or the entire index pair could degenerate. More subtle is a situation where the combinatorial image of A does not overlap B, but does after a slight shift or rescaling of the grid. These issues bring into question the practical robustness of the DFT method with respect to these scaling parameters.

To measure this robustness, we plot the computed entropy lower bound for a particular Hénon map against the area and the aspect ratio of the boxes in Figure 3. There we define $w_1 = y_1^i - x_1^i$ and $w_2 = y_2^i - x_2^i$ to be the width of a box $\mathcal{B}_i$ in dimensions 1 and 2, respectively (recall that in our setting these are independent of i). The grid resolution may be defined as $-\frac{1}{2} \log_2 w_1 w_2$, or more generally as $-\frac{1}{n} \log_2 V(\mathcal{B}_i)$, where V denotes the n-dimensional volume. We choose the this formula so that after normalizing by the volume of the some fixed box $\mathcal{B}$ the notions of resolution and depth align.

Of course, the DFT method may behave very differently on other maps, but the behavior shown is quite typical: the entropy lower bound is roughly monotone increasing with respect to the resolution, and (very) roughly unimodal with respect to the aspect ratio. This behavior is not surprising; the error from discretization decreases with the box area, and both extremes of the aspect ratio (i.e. $w_1 \gg w_2$ and $w_2 \gg w_1$) should result in essentially 1-dimensional information and hence a trivial entropy lower bound for most maps.

While the DFT method appears to be relatively robust with respect to the grid resolution, Figure 3(a) clearly shows a high sensitivity to the aspect ratio. Specifically, small changes in the aspect ratio at depth 9 resulted in large jumps in the entropy lower bound, even when close to the optimal ratio. While the behavior at depth 10 is somewhat more typical, this sensitivity is still something to keep in mind. In particular, for the sake of replication, care should be taken when altering and storing the aspect ratio.

To find an appropriate aspect ratio for a given map, one can perform a scaling parameter exploration at a lower depth, similar to the one in Figure 3(a). This can be expensive for higher-dimensional systems, however. In such cases it may be useful to use the following technique:

1. Start at an initial aspect ratio and compute a semi-conjugate subshift using the DFT method.

2. Use Algorithm 1 or a brute-force search to simplify the subshift, keeping track of the regions which were amalgamated.

3. Examine the amalgamations that were performed, and note whether any particular dimension was amalgamated along more than another; if so, scale the boxes *down* in that direction and repeat from step 1.

This approach is motivated by the intuition that amalgamations should identify scaling imbalances: if the boxes are too small in a particular dimension, the

index pair will likely be amalgamated along that dimension (or more precisely, the symbols corresponding to those index pair regions).

Note also that one can scale a map in a non-constant fashion to better align it with the grid; consider the following map, which is just the Hénon map (1) conjugated by the map $g(x,y) = (\exp(cy)x, y)$.

$$f(x,y) = \Big((a - (\exp(cy)x)^2 + by)\exp(-c\exp(cy)x),\ \exp(cy)x\Big) \qquad (6)$$

Certainly for large values of c, this map would be more aptly analyzed by first conjugating by g^{-1}. More generally, such non-constant scaling may be used to focus on areas af the phase space where isolation is more difficult.

We now return to the resolution scaling discussed above. As mentioned, the DFT method is surprisingly robust with respect to the resolution of the grid, and the entropy bound is roughly monotone increasing with the resolution. Similar plots shown in [4] suggested monotonicity, but had insufficient data; it is likely that a continuous resolution scaling would fill in these plots to reveal a rough monotonicity in that case as well.

This monotonicity is of course beneficial behavior, as one would like the precision of the bound to increase with the precision of the box covering, but it is especially useful in hyperbolic settings. By [20, Theorem 9.6.1], hyperbolic systems admit a finite Markov partition of the invariant set, and since the non-wandering set is disconnected (see above), in theory this partition is obtainable when the boxes are small enough. Thus we expect to obtain the true entropy value at a high enough resolution, and by seeing where the entropy levels off we can be confident, though not certain, that our lower bound is the actual value. We will apply this intuition in Section 5.

4.4 Implementation and efficiency

The computations in this paper were performed in Matlab, using Intlab [21], GAIO [6], and CHomP [10]. The machines used had memory between 1 and 2 gigabytes and clock speeds between 1 and 2.2 gigahertz. The runtimes varied, as we will describe below, but ranged from a few seconds at low grid resolutions to several hours for high resolutions.

As mentioned above, it is useful to observe how the entropy bound changes as we increase the grid resolution, but of course this procedure is not without cost. One would expect the runtime of the bound computation to increase, perhaps dramatically, with the resolution of the discretization. Empirically, the running time seems to grow roughly as n^r where n is the dimension of the invariant set, and r is the resolution of the grid, as defined above.

Fortunately, as we have mentioned several times, the DFT method and the new techniques we add here are all completely automated. By this we mean that one needs only to specify the map and the parameters to be studied (and the discretization parameters discussed above), and the rest of the computation, all the way to the semi-conjugate subshift, is done without any further human
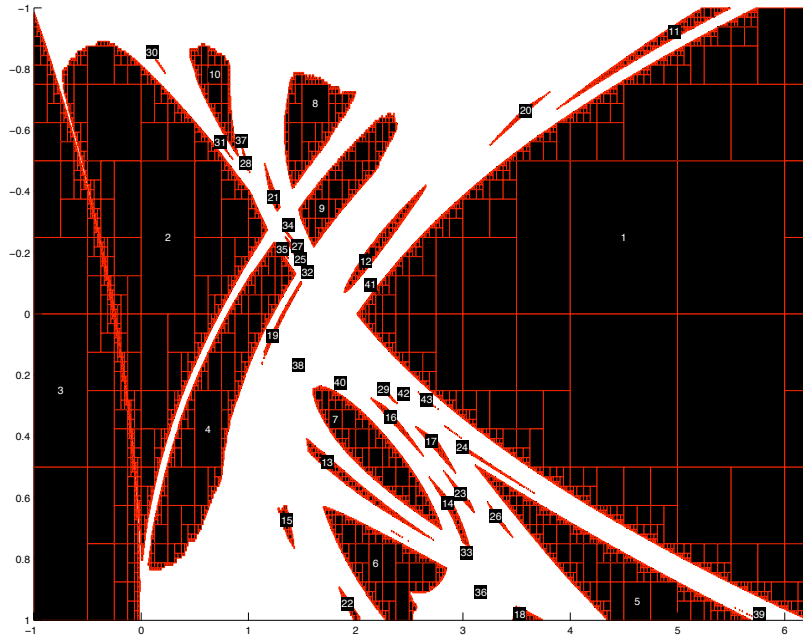
Figure 4: Hyperbolic plateaus for Hénon from [1], with a on the horizontal axis and b on the vertical axis. The label for each plateau is centered over the parameter values used to represent the plateau.

action or input. Thus, although the computations may be time-consuming for high resolutions, it is computation time, not human time.

This lack of manual intervention enables vast explorations of map and discretization parameters. For this paper, a total of 58 parameter values of the Hénon map were studied: 43 in Section 5 and 15 in Section 5.1. For each of the 58 parameter values, we computed entropy bounds at 64 or 72 different resolutions, yielding a total of over 3900 separate computations. This exploration would have been infeasible without complete automation (or a large team of researchers).

5 Hyperbolic plateaus of Hénon

We now apply the methods outlined in Section 4 to the real-valued Hénon map

$$f_{a,b}(x,y) = (a - x^2 + by, x) \tag{7}$$

for parameter values (a, b) such that $f_{a,b}$ is (uniformly) hyperbolic. Note that this *excludes* the classical parameters $(1.4, 0.3)$; for rigorous topological lower bounds in the classical case, see [4] and [17].

Using the hyperbolic plateaus of Arai, we select representative parameter values to study for each plateau, as shown in Figure 4. For each plateau, we use the continuous resolution scaling approach mentioned in Section 4.3, to get
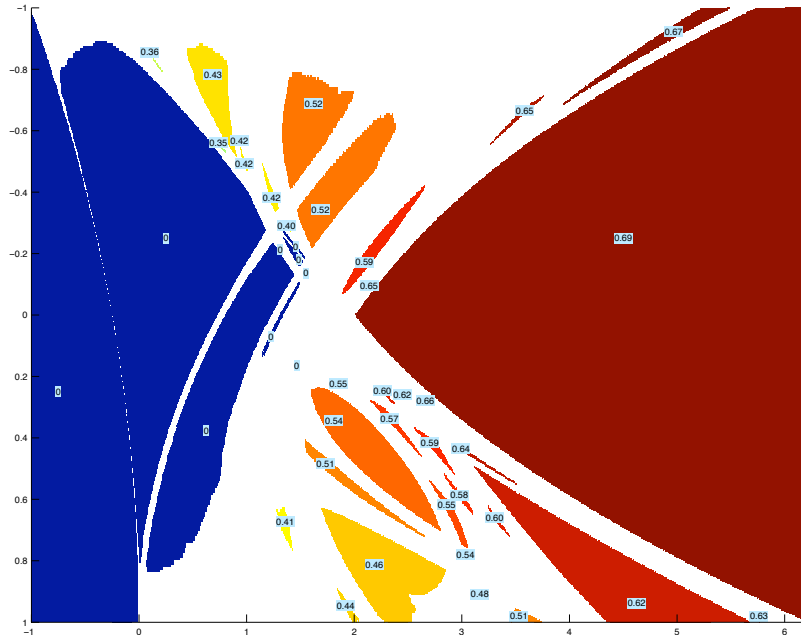
Figure 5: Rigorous lower bounds for topological entropy for the hyperbolic plateaus labeled 1 through 43 in Figure 4.
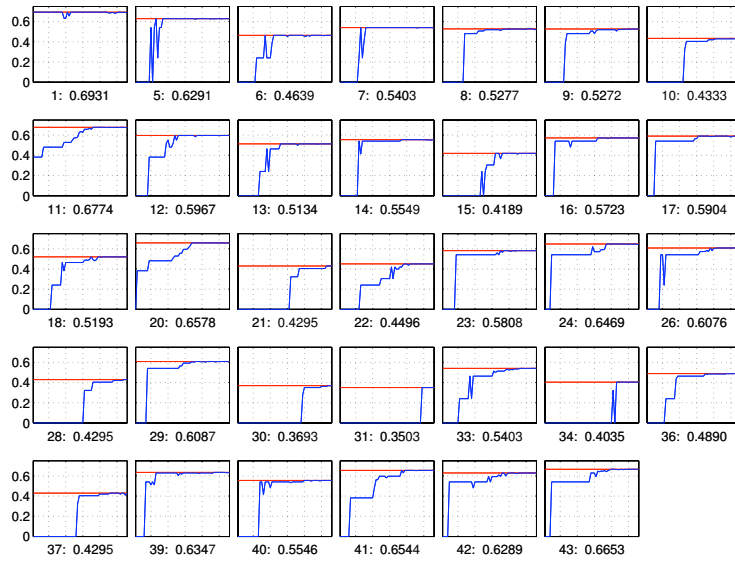

Figure 6: Entropy versus resolution used. The red horizontal line in each represents the maximum lower bound computed (printed below each plot). Plots are omitted 0-entropy plateaus (2, 3, 4, 19, 25, 27, 32, 35, 38).

16

a feel for how close our bounds might be to the actual values. The entropy bounds we compute constitute our main result, summarized in Theorem 5.1.

Theorem 5.1. *Let $F_i = \{f_{a,b} \mid (a,b) \in R_i\}$, where R_i is the ith plateau in Figure 4. Then for all i and all $f \in F_i$ we have $h(f) \geq h_i$, where the h_i are defined below:*

$$h_1 = 0.6931 \quad h_5 = 0.6291 \quad h_6 = 0.4639 \quad h_7 = 0.5403 \quad h_8 = 0.5277$$
$$h_9 = 0.5270 \quad h_{10} = 0.4333 \quad h_{11} = 0.6774 \quad h_{12} = 0.5967 \quad h_{13} = 0.5134$$
$$h_{14} = 0.5549 \quad h_{15} = 0.4189 \quad h_{16} = 0.5723 \quad h_{17} = 0.5904 \quad h_{18} = 0.5193$$
$$h_{20} = 0.6578 \quad h_{21} = 0.4295 \quad h_{22} = 0.4496 \quad h_{23} = 0.5808 \quad h_{24} = 0.6469$$
$$h_{26} = 0.6076 \quad h_{28} = 0.4295 \quad h_{29} = 0.6087 \quad h_{30} = 0.3693 \quad h_{31} = 0.3503$$
$$h_{33} = 0.5403 \quad h_{34} = 0.4035 \quad h_{36} = 0.4890 \quad h_{37} = 0.4295 \quad h_{39} = 0.6347$$
$$h_{40} = 0.5546 \quad h_{41} = 0.6544 \quad h_{42} = 0.6289 \quad h_{43} = 0.6653.$$

Proof. For each R_i we selected $(a_i, b_i) \in R_i$ as a representative (these choices are shown in Figure 4). We then computed the map on boxes for f_{a_i,b_i} at different resolutions, and for each resolution we computed a rigorous lower bound for topological entropy using the DFT method; these bounds are summarized in Figure 6. Finally, by [1] we know each R_i is (uniformly) hyperbolic and so for each i we can apply the maximum lower bound achieved for (a_i, b_i) to all of R_i. □

Figures 1 and 5 show an overview of our results. Note that the entropy values shown are merely lower bounds, and not necessarily the true values. Since we are computing these bounds for hyperbolic parameter values, however, we know from Section 4.3 that if our entropy lower bound levels off as the grid resolution increases, we have strong evidence that we have obtained the correct value. Typical index pairs from these computations are shown in Figure 7, with colored regions corresponding to symbols in the resulting subshift.

In Figure 6, we show plots of entropy bounds computed versus the resolution, for each of the 43 parameter values with a nonzero bound. Using the above heuristic, it seems that for most of the plateaus, the bounds we computed should be exact or very close. A few notable exceptions are plateaus 9, 10, 21, 28, 30, and 31, since the plots do not seem to have leveled off, and we would expect many of these bounds to improve with further computations. For plateaus 33, 34, 36, 37, and 39, it is unclear whether the plot has leveled off. As we saw in Section 4.3, the DFT method is fairly robust with respect to the grid resolution, and the entropy lower bound is roughly monotonic in the resolution. The plots in Figure 6 reaffirm this, with very few exceptions.

While we have computed a large array of lower bounds, covering a vast portion of the parameter space of the Hénon map, a recent result of Arai gives a method of rigorously computing *exact* entropy values for (uniformly) hyperbolic Hénon [2]. In fact, in that work he computes values for plateaus numbered 5, 7, 11, and 12 in Figure 4, which match our lower bounds exactly. In that it computes exact entropy values, the method of [2] is certainly superior to the DFT method in the case we study here, and indeed it would be very interesting

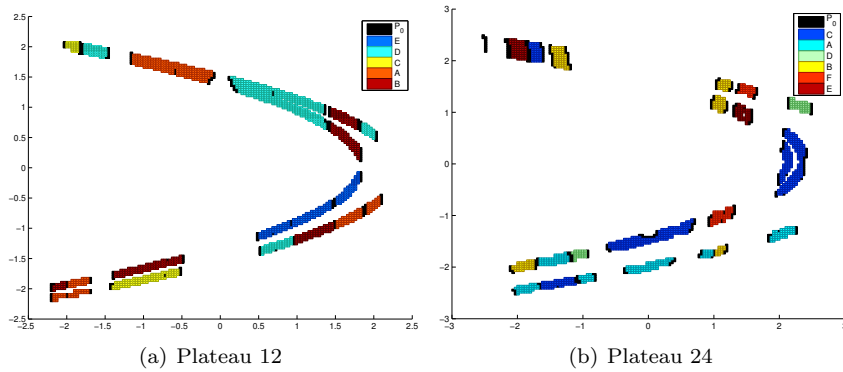(a) Plateau 12　　　　　　(b) Plateau 24

Figure 7: Two index pairs from the computations. The black is the exit set P_0 while the colors correspond to the amalgamated regions after applying Algorithm 1.

to use it to test the accuracy of our other bounds. However, it is important to note that since the method of [2] relies heavily on hyperbolicity, it is not as generally applicable. In particular, it could not be applied to the Hénon map for the classical parameter values, studied in [4], as the map is not uniformly hyperbolic for those parameters.

While much of the parameter space in Figure 5 is covered by our lower bounds, there is still much of the parameter space which is not hyperbolic, or has not yet been proven to be hyperbolic. Thus, it remains in future work to lower-bound the entropy in the remaining white regions. At first this seems like a daunting or impossible task, since in the nonhyperbolic regions, we no longer have plateaus of topological entropy, and thus cannot extend a bound from a single set of parameters to an entire region. Fortunately, the DFT method can be applied to intervals of parameter values $[a_1, a_2] \times [b_1, b_2]$, yielding a single lower bound which applied to the entire interval, as demonstrated in [4] and [8]. Thus, it would be of great interest to compute lower bounds on an interval tiling of the Hénon parameter space, and then compare these bounds to those computed here; as mentioned in Section 4.4, the complete automation of our method enables such parameter explorations.

5.1　Area-preserving Hénon maps

When $b = -1$, the Hénon maps are area-preserving and orientation-preserving. This case has been well-studied, especially in the Physics literature. Starting in 1991, Davis, MacKay, and Sannami (DMS) [3] conjectured that Hénon was hyperbolic for three values of a (5.4, 5.59, 5.65) and conjectured conjugacies to symbolic dynamics for these three values as well. In 2002, de Carvalho and Hall in [5] replicated the results for $a = 5.4$ using a pruning approach. In 2004, Hagiwara and Shudo in [12] used a different pruning method to replicate all
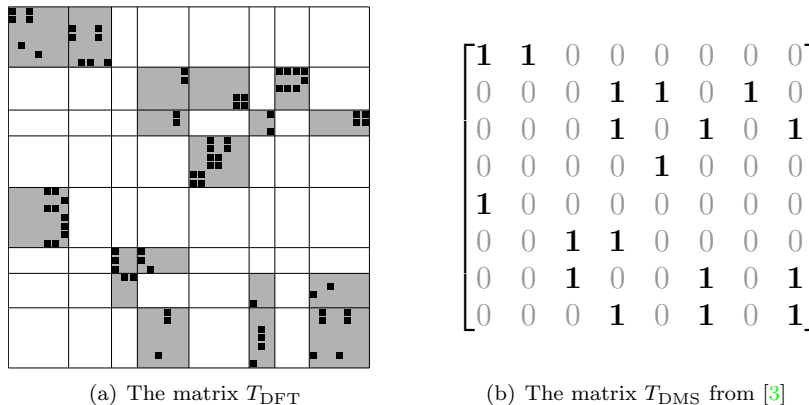
(a) The matrix T_{DFT}

$$\begin{bmatrix} \mathbf{1} & \mathbf{1} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \mathbf{1} & \mathbf{1} & 0 & \mathbf{1} & 0 \\ 0 & 0 & 0 & \mathbf{1} & 0 & \mathbf{1} & 0 & \mathbf{1} \\ 0 & 0 & 0 & 0 & \mathbf{1} & 0 & 0 & 0 \\ \mathbf{1} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \mathbf{1} & \mathbf{1} & 0 & 0 & 0 & 0 \\ 0 & 0 & \mathbf{1} & 0 & 0 & \mathbf{1} & 0 & \mathbf{1} \\ 0 & 0 & 0 & \mathbf{1} & 0 & \mathbf{1} & 0 & \mathbf{1} \end{bmatrix}$$

(b) The matrix T_{DMS} from [3]

Figure 8: Amalgamation of the 42×42 symbol matrix obtained using the DFT method. The black squares in (a) denote the nonzero entries of T_{DFT} while the gray regions represent the amalgamated symbols, which one can easily see match T_{DMS} exactly.

three of the values that DMS studied, and two more. They also give estimates of the topological entropy for $4 \leq a \leq 5.7$, which are displayed in Figure 10. Finally in 2007 some rigorous results appeared by Arai in [1], where he proved that there are 16 hyperbolic regions for $b = -1$, covering the parameters studied by DMS and the two others studied by Hagiwara and Shudo. Arai goes on in [2] to prove that the subshift conjectured by DMS for $a = 5.4$ is actually conjugate.

While our method cannot prove exact topological entropy values or conjugacies, we can attempt to verify that the entropy of the subshifts given in [3] are lower bounds, and perhaps show that the subshifts themselves are semi-conjugate. We focus first on the $a = 5.4$ case, where Davis et al. conjectured that f_a is conjugate to the subshift corresponding to the transition matrix T_{DMS} in Figure 8(b), which has topological entropy $h(T_{\text{DMS}}) \approx 0.6774$. Note that this is the same plateau as plateau 11 from the previous section.

Using our technique, we prove that $f_{5.4}$ is semi-conjugate to a 42×42 symbol matrix T_{DFT}, depicted in Figure 8(a). The topological entropy of this matrix is the same value, $h(T_{\text{DFT}}) \approx 0.6774$. The fact that $h(T_{\text{DMS}}) = h(T_{\text{DFT}})$ suggests that the subshifts given by T_{DMS} and T_{DFT} might be conjugate, and indeed we prove this in Theorem 5.2.

Theorem 5.2. *The map $f_{5.4}$ is semi-conjugate to the subshift T_{DMS} given in Figure 8(b). Moreover the symbols of T_{DMS} correspond to the regions labeled in Figure 9, which are the same regions conjectured by DMS.*

Proof. Applying Algorithm 1 to T_{DFT}, we obtain a strong shift equivalence between T_{DMS} and T_{DFT}, which shows that the corresponding subshifts are indeed conjugate. Moreover, the amalgamated vertices can be chosen so that we obtain the same partition that was used by Davis et al., which is shown in
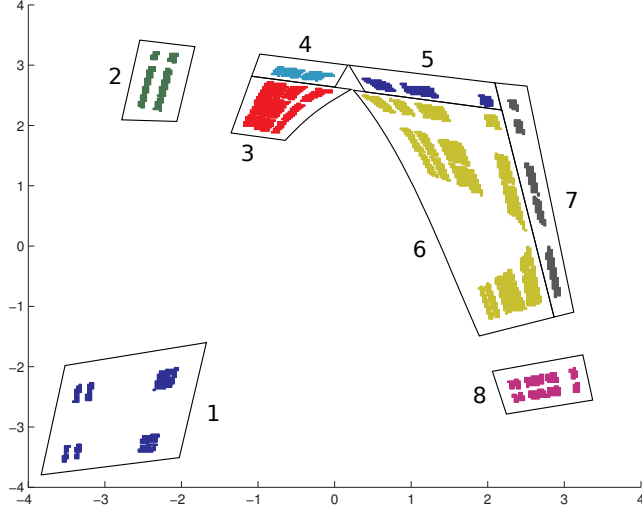
Figure 9: Index pair for $a = 5.4$ and $b = -1$, with collections of regions labeled to indicate the symbols for the smaller, shift-equivalent symbol system.

Figure 9, with the regions labeled so as to match the symbols (row indices) of T_{DMS}. □

Our method also gives lower bounds which match the values conjectured by Davis et al. for $a = 5.59$ and $a = 5.65$, as well as the value conjectured by Hagiwara and Shudo for $a = 4.58$. In addition to the these values, we also focus on 11 other values, which all together correspond to the first 15 area-preserving plateaus computed by Arai (the 16th is the maximal entropy plateau, which is plateau 1 in the previous section). Our results are summarized in the following theorem.

Theorem 5.3. *The following entropy bounds hold for the Hénon maps* $f_a = f_{a,-1}$. *Here we write* $h(f_{[a_0,a_1]}) \geq v$ *to mean* $\forall a \in [a_0, a_1], h(f_a) \geq v$.

1. $h\left(f_{[4.5383,\ 4.5386]}\right) \geq 0.6373$ 2. $h\left(f_{[4.5388,\ 4.5430]}\right) \geq 0.6373$
3. $h\left(f_{[4.5624,\ 4.5931]}\right) \geq 0.6391$ 4. $h\left(f_{[4.6189,\ 4.6458]}\right) \geq 0.6404$
5. $h\left(f_{[4.6694,\ 4.6881]}\right) \geq 0.6429$ 6. $h\left(f_{[4.7682,\ 4.7993]}\right) \geq 0.6459$
7. $h\left(f_{[4.8530,\ 4.8604]}\right) \geq 0.6466$ 8. $h\left(f_{[4.9666,\ 4.9692]}\right) \geq 0.6527$
9. $h\left(f_{[5.1470,\ 5.1497]}\right) \geq 0.6718$ 10. $h\left(f_{[5.1904,\ 5.5366]}\right) \geq 0.6774$
11. $h\left(f_{[5.5659,\ 5.6078]}\right) \geq 0.6814$ 12. $h\left(f_{[5.6343,\ 5.6769]}\right) \geq 0.6893$
13. $h\left(f_{[5.6821,\ 5.6858]}\right) \geq 0.6893$ 14. $h\left(f_{[5.6859,\ 5.6860]}\right) \geq 0.6893$
15. $h\left(f_{[5.6917,\ 5.6952]}\right) \geq 0.6893$

Proof. Using the DFT method, we computed bounds for a single a value for each plateau; the representatives chosen were the following: 4.5385, 4.5409,
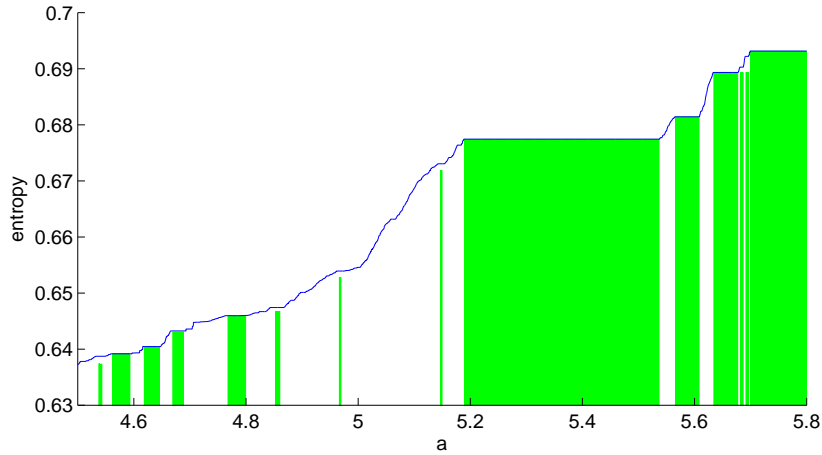
Figure 10: Topological entropy of the area-preserving Hénon maps, where $b = -1$. The estimates produced by Hagiwara and Shudo are in blue, and our rigorous lower bounds are in green.

4.5800, 4.6323, 4.6788, 4.7838, 4.8600, 4.9679, 5.1483, 5.4000, 5.5900, 5.6500, 5.6839, 5.6859, 5.6934. Combining these bounds with the hyperbolic plateaus computed in [1], and using Theorem 2.4, we can extend the bounds to their corresponding plateau. □

Figure 10 shows a plot of the lower bounds from Theorem 5.3, shown against the estimates computed by Shudo and Hagiwara in [12]. The 4 cases discussed above correspond to plateaus 3, 10, 11, and 12. For these plateaus, our lower bounds match the estimates exactly, and our bounds for plateaus 4, 5, and 6 are very close. This is roughly what we expect given the resolution plots shown in Figure 11. An interesting trend we see in these data is that the algorithm performed better on the larger plateaus. This is perhaps because the stable and unstable manifolds seem to be more transverse the farther the parameters are from a bifurcation, thus making it easier to isolate the important regions of the phase space.
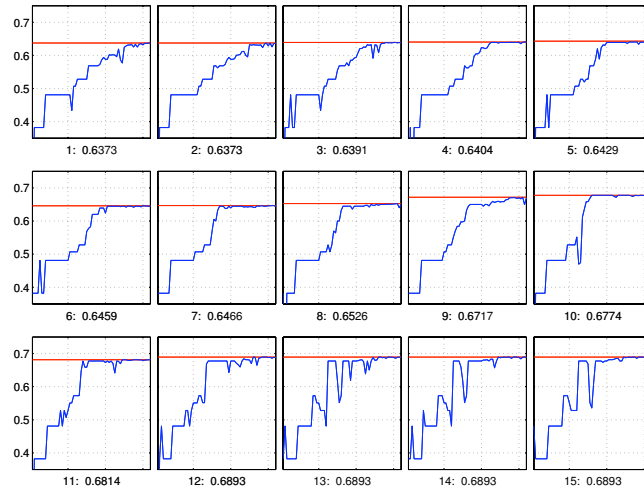
Acknowledgements

Figure 11: Entropy against resolution, as in Figure 6.

References

[1] Zin Arai. On hyperbolic plateaus of the Hénon map. *Experiment. Math.*, 16(2):181–188, 2007.

[2] Zin Arai. On loops in the hyperbolic locus of the complex hénon map. Preprint, 2007.

[3] M. J. Davis, R. S. MacKay, and A. Sannami. Markov shifts in the Hénon family. *Phys. D*, 52(2-3):171–178, 1991.

[4] Sarah Day, Rafael Frongillo, and Rodrigo Treviño. Algorithms for rigorous entropy bounds and symbolic dynamics. *SIAM Journal on Applied Dynamical Systems*, 7:1477–1506, 2008.

[5] André de Carvalho and Toby Hall. How to prune a horseshoe. *Nonlinearity*, 15(3):R19–R68, 2002.

[6] Michael Dellnitz, Gary Froyland, and Oliver Junge. The algorithms behind GAIO-set oriented numerical methods for dynamical systems. In *Ergodic theory, analysis, and efficient simulation of dynamical systems*, pages 145–174, 805–807. Springer, Berlin, 2001.

[7] John Franks and David Richeson. Shift equivalence and the Conley index. *Trans. Amer. Math. Soc.*, 352(7):3305–3322, 2000.

[8] R. Frongillo and R. Treviño. Efficient automation of index pairs in computational conley index theory. *SIAM Journal on Applied Dynamical Systems*, 11:82, 2012.

[9] Rafael M. Frongillo. The hardness of state amalgamation in strong shift equivalence. Preprint, 2012.

[10] CHomP group. Computational Homology Project (CHomP). http://chomp.rutgers.edu/.

[11] John Guckenheimer and Philip Holmes. *Nonlinear oscillations, dynamical systems, and bifurcations of vector fields*, volume 42 of *Applied Mathematical Sciences*. Springer-Verlag, New York, 1990. Revised and corrected reprint of the 1983 original.

[12] Ryouichi Hagiwara and Akira Shudo. An algorithm to prune the area-preserving Hénon map. *J. Phys. A*, 37(44):10521–10543, 2004.

[13] Suzanne Lynch Hruska. A numerical method for constructing the hyperbolic structure of complex Hénon mappings. *Found. Comput. Math.*, 6(4):427–455, 2006.

[14] Tomasz Kaczynski, Konstantin Mischaikow, and Marian Mrozek. *Computational homology*, volume 157 of *Applied Mathematical Sciences*. Springer-Verlag, New York, 2004.

[15] Douglas Lind and Brian Marcus. *An Introduction to Symbolic Dynamics and Coding*. Cambridge University Press, Cambridge, 1995.

[16] M. Mazur and J. Tabor. Computational hyperbolicity. *Discrete and Continuous Dynamical Systems (DCDS-A)*, 29(3):11751189, 2011.

[17] S. Newhouse, M. Berz, J. Grote, and K. Makino. On the estimation of topological entropy on surfaces. *Contemporary Mathematics*, 469:243–270, 2008.

[18] P. Pilarczyk. *Homology Computation-Software and Examples*. Jagiellonian University, 1998. (http://www.im.uj.edu.pl/∽pilarczy/homology.htm).

[19] Joel W. Robbin and Dietmar Salamon. Dynamical systems, shape theory and the Conley index. *Ergodic Theory Dynam. Systems*, 8*(Charles Conley Memorial Issue):375–393, 1988.

[20] Clark Robinson. *Dynamical systems*. Studies in Advanced Mathematics. CRC Press, Boca Raton, FL, 1995. Stability, symbolic dynamics, and chaos.

[21] S.M. Rump. INTLAB - INTerval LABoratory. In Tibor Csendes, editor, *Developments in Reliable Computing*, pages 77–104. Kluwer Academic Publishers, Dordrecht, 1999. http://www.ti3.tu-harburg.de/rump/.

[22] D. Sterling, H. R. Dullin, and J. D. Meiss. Homoclinic bifurcations for the Hénon map. *Phys. D*, 134(2):153–184, 1999.

[23] R. F. Williams. Classification of one dimensional attractors. In *Global Analysis (Proc. Sympos. Pure Math., Vol. XIV, Berkeley, Calif., 1968)*, pages 341–361. Amer. Math. Soc., Providence, R.I., 1970.

[24] R. F. Williams. Classification of subshifts of finite type. In *Recent advances in topological dynamics (Proc. Conf. Topological Dynamics, Yale Univ., New Haven, Conn., 1972; in honor of Gustav Arnold Hedlund)*, pages 281–285. Lecture Notes in Math., Vol. 318. Springer, Berlin, 1973.