Ergod. Th. & Dynam. Sys. (2017), **XX**, 1
Printed in the United Kingdom

Optimal State Amalgamation is NP-Hard

Rafael M. Frongillo
University of Colorado at Boulder
raf@colorado.edu

(*Received* 13 *June* 2017)

Abstract. A state amalgamation of a directed graph is a node contraction which is only permitted under certain configurations of incident edges. In symbolic dynamics, state amalgamation and its inverse operation, state splitting, play a fundamental role in the the theory of subshifts of finite type (SFT): any conjugacy between SFTs, given as vertex shifts, can be expressed as a sequence of symbol splittings followed by a sequence of symbol amalgamations. It is not known whether determining conjugacy between SFTs is decidable.

We focus on conjugacy via amalgamations alone, and consider the simpler problem of deciding whether one can perform k consecutive amalgamations from a given graph. This problem also arises when using symbolic dynamics to study continuous maps, where one seeks to coarsen a Markov partition in order to simplify it. We show that this state amalgamation problem is NP-complete by reduction from the hitting set problem, thus giving further evidence that classifying SFTs up to conjugacy may be undecidable.

1. *Introduction*

Subshifts of finite type (SFT) are a basic class of symbolic dynamical systems which find applications in many fields, including mathematics, information theory, computer science, and physics. In a formative 1970 paper, Williams [30] introduced the concepts of *shift equivalence* and *strong shift equivalence*, showing that two SFTs are eventually conjugate if and only if they are shift equivalent, and are topologically conjugate if and only if they are strongly shift equivalent. Kim and Roush [22] showed in 1988 that shift equivalence is decidable, and as strong shift equivalence implies shift equivalence, the natural question of whether the two conditions were equivalent arose. This question, known as the Williams Conjecture, was settled negatively by Kim and Roush ten years later [21]. Nonetheless, the decidability of strong shift equivalence, and hence the conjugacy of SFTs, is yet unknown.

Williams [30] further showed that any strong shift equivalence may be decomposed into a sequence of state splittings, where a state s is replaced by a pair s_1,s_2, and *amalgamations*, where two states s, s' are collapsed to a single state $[ss']$. In this paper, we study a natural simplification of this problem, where a strong shift equivalence involves only a sequence of amalgamations.† Specifically, we define the decision problem SAP, which is to decide, given a graph G and positive integer k, whether one can perform k consecutive amalgamations in G. By reduction from the hitting set problem [20], we show that this problem is NP-complete. This in turn implies that it is NP-hard to find the graph on the smallest number of states which can be obtained from G by amalgamation alone.

A natural application of state amalgamation arises in the study of continuous maps using symbolic dynamics. Here one labels disjoint regions of the phase space and considers the subshift defined by the label sequences in forward and backward time of points in these regions. One common example is a *Markov partition* [1], where the regions satisfy the Markov property, and the resulting shift space is an SFT. When presented with such a set of regions and its corresponding shift space, one may wish to *coarsen* the regions, by combining several of them together, to simplify the partition while preserving the conjugacy to some SFT. In the case of a Markov partition this coarsening can be done by amalgamations (§ 5).

1.1. *Related Work.* Finding state amalgamations to simplify the "description" of vertex shifts is similar to the goal of minimizing states in sofic shifts [8, 17, 18, 11, 29]. However, despite the term "state" being used in both contexts, the problems are not the same: consider the graph on two states $\{a, b\}$ with two edges (a, b),(b, a) both labeled 0. The state-minimized presentation is just a single state with a self loop labeled 0, but this graph cannot be obtained from the first via state amalgamation. Moreover, state minimization can be performed efficiently [17], while as we show, optimal state amalgamation is NP-hard.

Various generalizations of one-dimensional shift spaces have been introduced, including higher-dimensional shifts [25] as well as tree shifts of finite type [3]. In some of these settings, generalizations of Williams' decomposition theorem have been established [16, 3], showing that conjugacies between shifts of finite type can be expressed as a sequence of splittings followed by amalgamations, suitably defined. Conjugacy of higher-dimensional SFTs has been shown to be undecidable (in fact, Σ^0_1-complete) by Berger [5] who phrased the problem in terms of tilings (see Kari [19] for a general discussion of such problems), and recently Jeandel and Vanier [15] have shown that the simpler problem of determining conjugacy to any given fixed SFT is still undecidable (and Σ^0_1-complete). The same authors also give the recursion-theoretic complexity of related problems, such as deciding factorization and embedding. For tree shifts of finite type, Aubrun and Béal [3] show that conjugacy is decidable; as the decidability of conjugacy for two-sided one-dimensional subshifts of finite type remains an open question, it would be interesting to study optimal state amalgamation in the tree shift setting.

† Importantly, we consider both in- and out-amalgamations, or when thought of as matrices, both row and column amalgamations; further restricting to columns only corresponds to conjugacy of the one-sided subshifts, which can be decided in polynomial time [23].

Finally, the problem of coarsening Markov partitions has been considered in the one-dimensional case by Teramoto and Komatsuzaki [28], who show that a particular method for refining or coarsening a partition preserves conjugacy in that setting. We restrict to amalgamation operations, which ensure conjugacy. Zheng et al. [32] argue that even when finding an initial Markov partition is costly or impractical, approximating the Markov partition can still be useful.

2. *Background and Definitions*

We begin with the basic definitions needed to state our result. We will make use of both graph and matrix representations of SFTs; strong shift equivalence is a linear algebraic notion, while our main result is best viewed in graph-theoretic terms.

2.1. *Subshifts of finite type and strong shift equivalence.*

Given a finite alphabet $\mathcal{A}$, define a *symbol space* $X_{\text{full}} = \mathcal{A}^{\mathbb{Z}}$ to be the set of all bi-infinite sequences of symbols in $\mathcal{A}$. It is well-known that X_{full} is a metric space [24]. When $|\mathcal{A}| = n$, we define *full n-shift* $\sigma : X_{\text{full}} \to X_{\text{full}}$ as the map acting on X_{full} by $(\sigma(x))_i = x_{i+1}$. A *word* is a finite sequence of symbols in $\mathcal{A}$; given a set of "forbidden" words $\mathcal{F} \subseteq \cup_{i\geq 1}\mathcal{A}^i$, the shift space $X_{\mathcal{F}}$ is defined as the set of all bi-infinite sequences $x \in X_{\text{full}}$ such that $(x_i, x_{i+1}, \ldots, x_j) \notin \mathcal{F}$ for all $i, j \in \mathbb{Z}$, $i < j$. If a shift space can be written as $X_{\mathcal{F}}$ for some finite $\mathcal{F}$, it is a *subshift of finite type (SFT)*. (The shift map for $X_{\mathcal{F}}$ in both cases is merely σ restricted to $X_{\mathcal{F}}$.)

Given a directed graph G with vertices being the symbols $V(G) = \mathcal{A}$, and edges $E(G)$, we can define its *vertex shift* as a subshift $X_G \subseteq X_{\text{full}}$, where $x \in X_G$ if and only if $(x_i, x_{i+1}) \in E(G)$ for all $i \in \mathbb{Z}$. That is, X_G consists of all sequences in X_{full} with transitions of σ allowed by the edges of G. Importantly, any SFT is conjugate to some vertex shift.

Theorem 1 ([23, 24]) *For every SFT $X_{\mathcal{F}}$, there is a directed graph G such that $X_{\mathcal{F}}$ is conjugate to X_G.*

Given a vertex shift X_G for a graph G, we will ask whether there is a graph H on fewer vertices such that X_G and X_H are conjugate. To this end, we recall the notion of strong shift equivalence.

Definition 1 (Strong shift equivalence) *Let A and B be nonnegative integer matrices. An* elementary shift equivalence *between A and B is a pair (R, S) such that*

$$A = RS \text{ and } B = SR. \tag{1}$$

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

Strong shift equivalence allows us to classify SFTs up to conjugacy, as the following result of R. F. Williams shows. The *transition matrix* of a graph G

on n vertices, also called the adjacency matrix, is defined as $A \in \{0,1\}^{n\times n}$ such that $A_{ij} = 1$ if and only if $(i,j) \in E(G)$.

THEOREM 2 ([**31**]) *For directed graphs G and H, the corresponding vertex shifts X_G and X_H are conjugate if and only if the transition matrices of G and H are strongly shift equivalent.*

2.2. *Amalgamation.* Theorem 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 challenging problem. Two methods of finding such equivalences are given by Williams [**30**]: 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 together and then removing duplicate edges (i.e. removing edges until there is at most one edge between any pair of vertices). In general, obtaining the smallest element (measured by number of vertices or rows) of a strong shift equivalence class may involve both splittings and amalgamations. We will instead focus on the simpler problem of obtaining H only by amalgamating vertices in G.

We will denote the image (out-neighbors) and preimage (in-neighbors) of a vertex v in G by $N^+(v)$ and $N^-(v)$, respectively. The following definition of amalgamation is adapted from [**24**]; see Figure 1 for an example.

DEFINITION 2 (AMALGAMATION) *Let G be a directed graph on n vertices and let $u, v \in V(G)$. We say that u and v are* amalgamation candidates*, or that they* can amalgamate*, if they satisfy either of the following conditions:*

$$\textbf{\textit{Forward Condition:}} \quad N^+(u) = N^+(v) \textit{ and } N^-(u) \cap N^-(v) = \emptyset, \tag{2}$$

$$\textbf{\textit{Backward Condition:}} \quad N^-(u) = N^-(v) \textit{ and } N^+(u) \cap N^+(v) = \emptyset. \tag{3}$$

The amalgamation *of amalgamation candidates u and v is the graph on $n-1$ vertices given by contracting u and v to a single vertex w, with $N^+(w) = N^+(u) \cup N^+(v)$ and $N^-(w) = N^-(u) \cup N^-(v)$.*

In other words, vertices can amalgamate if they have the same image and disjoint preimages, or the same preimage and disjoint images. As mentioned above, we now show why amalgamations are useful: amalgamations satisfying either condition (2) or condition (3) yield a strong shift equivalence between the transition matrices of the two graphs, and hence a conjugacy between the corresponding subshifts by Theorem 2. For completeness, we give a proof; see also [**24**, § 2].

THEOREM 3. *Let u and v satisfy the forward condition (2) or backward condition (3) for a graph G, and let G' be the graph after amalgamating u and v. Then there is an elementary shift equivalence between the transition matrices of G and G'.*

Proof. Let A be the $n \times n$ transition matrix of G, and let i and j be the indices of u and v, respectively. Note that in matrix notation, the forward condition (2)

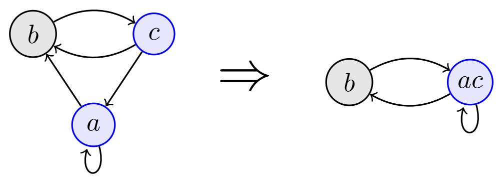

FIGURE 1. A forward amalgamation

is just $Ae_i = Ae_j$ and $(e_i^\top A) \cdot (e_j^\top A) = 0$, and the backward condition (3) is $e_i^\top A = e_j^\top A$ and $(Ae_i) \cdot (Ae_j) = 0$. Here e_i denotes the column vector with a 1 in position i and zeros elsewhere. Note that the backward condition for A is the same as the forward condition for $A^\top$. We let

$$X = \left[\begin{array}{c|c} I_{j-1} & 0 \\ \hline \mathbf{0} & \mathbf{0} \\ \hline 0 & I_{n-j} \end{array}\right], \quad Y = \left[\begin{array}{c|c|c} I_{j-1} & e_i & 0 \\ \hline 0 & \mathbf{0} & I_{n-j} \end{array}\right], \tag{4}$$

where I_k denotes the $k \times k$ identity matrix, and 0 and $\mathbf{0}$ denote zero matrices and vectors, respectively, of the appropriate dimensions so that X is $n \times n-1$ and Y is $n-1 \times n$.

Assume the forward condition is satisfied for i and j in A, where $i < j$. Then we obtain $B = YAX$ by amalgamating i and j, which one verifies is the transition matrix for G'. We will show that $(AX, Y) : A \to B$, meaning that AX and Y give an elementary shift equivalence from A to B. We have $(Y)(AX) = B$ immediately, so it remains to show $(AX)(Y) = A$. Note that

$$XY = \left[\begin{array}{c|c|c} I_{j-1} & e_i & 0 \\ \hline \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \hline 0 & \mathbf{0} & I_{n-j} \end{array}\right], \tag{5}$$

and thus $AXYe_k = Ae_k$ if $k \neq j$ and $AXYe_j = Ae_i$. By the forward condition (2) we have $Ae_j = Ae_i$, so in fact $(AX)Y = A$.

Now assume instead that the backward condition is satisfied; we will show $(X^\top A, Y^\top) : A \to B$, where here $B = X^\top AY^\top$. Again, $(X^\top A)(Y^\top) = B$ is trivial. By the remark above, i and j satisfy the forward condition for $A^\top$, and so by the above computation we have $A^\top XY = A^\top$. Thus $Y^\top (X^\top A) = (A^\top XY)^\top = (A^\top)^\top = A$. □

To keep track of the origins of amalgamated vertices, we will associate each vertex with a set of its "ancestor" vertices in the original graph.

DEFINITION 3 (AMALGAMATION SEQUENCE) *A sequence of labeled graphs $G_0, \ldots, G_k$, with labels $S_0, \ldots, S_k$, is an* amalgamation sequence *for a graph G if:*

1. $G_0 = G$ and $S_0(v) = \{v\}$ for all v, and

2. Each G_{i+1} is obtained from G_i by amalgamating candidates $u, v \in V(G_i)$ to some $v' \in V(G_{i+1})$ with label $S_{i+1}(v') = S_i(u) \cup S_i(v) \subseteq V(G)$. (We set $S_{i+1}(\cdot) = S_i(\cdot)$ otherwise.)

We will write $G \Rightarrow G_0$ to imply that G_0 is obtained from G as above.

It is easy to see that the labels of vertices in $V(G_i)$ form a set partition of $V(G)$. Our bookkeeping is now sufficient to define the concept of eventual amalgamation, which we will use extensively in our reduction.

Definition 4 (Eventual Amalgamation) *A set of vertices $S \subseteq V(G)$ can* eventually amalgamate *in G if there is some amalgamation sequence $G_0, \ldots, G_k$ of G and some $v \in V(G_k)$ with $S \subseteq S(v)$.*

Thus, given a vertex shift defined by G, if a set of vertices $S \subseteq \mathcal{A}$ can eventually amalgamate, then there is some 1-block conjugacy which maps S to a single symbol.

2.3. *The decision problem.* We can now define our decision problem, which asks whether an amalgamation sequence of length at least k is possible in G. Note that, by standard arguments, the complexity of this the problem is the same as that of finding the *largest* value of k for which an amalgamation sequence exists.

Definition 5 (State Amalgamation Problem) *Let a directed graph G and an integer k be given. The* state amalgamation problem*, denoted SAP, is to decide whether there is a valid amalgamation sequence of length k for G.*

We now introduce the problem *hitting set* we will reduce from, which is simply the dual of the standard *set cover* problem, both shown to be NP-complete in Karp's seminal paper [20]. (For the definition of the class NP and related background, see e.g. Goldreich [13].) Here and in the reduction we use the notation $[m] := \{1, 2, \ldots, m\}$.

Definition 6 (Hitting Set) *Let $\mathcal{C} = \{S_1, \ldots, S_m\}$ be a collection of nonempty sets with $\bigcup_i S_i = U$. Given a subset $S \subseteq U$ we define its* hit set *as* $\mathrm{hit}(S) = \{i | S \cap S_i \neq \emptyset\}$*. Given $\mathcal{C}$, U, and an integer t, the* hitting set problem*, denoted HittingSet, is to decide whether there is a set H of cardinality t such that* $\mathrm{hit}(H) = [m]$.

The following fact will be useful later for our proof: if no hitting set of size t exists, then we can upper bound how many sets are covered by any H in terms of $|H|$, m, and t.

Lemma 4. *Suppose for some $t \leq m$ there is no H with $|H| \leq t$ and* $\mathrm{hit}(H) = [m]$*. Then for all $H \subseteq U$, $|\mathrm{hit}(H)| - |H| < m - t$.*

Proof. Given any $H \subseteq U$, for each $S_i \notin \mathrm{hit}(H)$ pick $s_i \in S_i$ and let $\hat{H} = (\bigcup \{s_i\}) \cup H$. Then $\mathrm{hit}(\hat{H}) = [m]$, so $\hat{H}$ is a hitting set. Thus by assumption we have,

$$t - m < |\hat{H}| - m = |H| + |\mathcal{C} \setminus \mathrm{hit}(H)| - m = |H| - |\mathrm{hit}(H)| ,$$

which gives the result. □

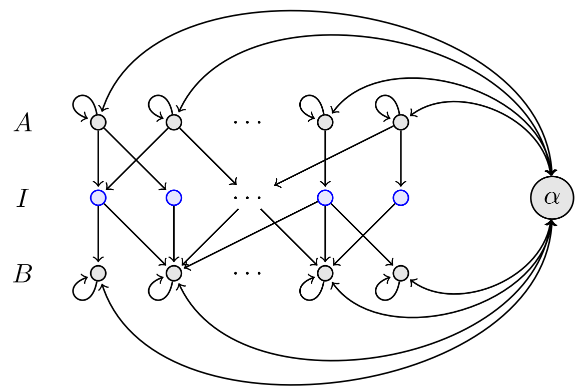

FIGURE 2. A graph satisfying the structure property.

3. *Reduction*

We will show that SAP is NP-complete by reduction from HittingSet. Throughout we will use the notation $v : [A_*, B_*]$ to denote "$N^-(v) = A_*$ and $N^+(v) = B_*$," or if the identity of the vertex is irrelevant we will simply write $[A_*, B_*]$ to denote an arbitrary such vertex. Additionally, when A_* or B_* are singleton sets, e.g. $A_* = \{a\}$, we will omit the braces and simply write $v : [a, B_*]$, etc.

Our proof relies heavily on the following graph structure, which creates a controlled environment where amalgamations are restricted. In particular, Lemma 5 shows that any amalgamations must be among vertices in I. See Figure 2 for an illustration of the property.

DEFINITION 7 (STRUCTURE PROPERTY) *A directed graph G satisfies the* structure property *if we can write $V(G) = \{\alpha\} \cup A \cup I \cup B$ such that:*

1. $N^-(\alpha) = N^+(\alpha) = A \cup B$
2. $\forall a \in A,\ N^-(a) = \{a, \alpha\},\ N^+(a) \subseteq I \cup \{a, \alpha\},\ \alpha \in N^+(a)$
3. $\forall b \in B,\ N^+(b) = \{b, \alpha\},\ N^-(b) \subseteq I \cup \{b, \alpha\},\ \alpha \in N^-(b)$
4. $\forall i \in I,\ N^-(i) \subseteq A,\ N^+(i) \subseteq B.$

LEMMA 5. *Let G satisfy the structure property with $V(G) = \{\alpha\} \cup A \cup I \cup B$. Then no $s \in V(G) \setminus I$ can be eventually amalgamated with any other $s' \in V(G)$.*

Proof. The proof follows by induction from two claims:

> **Claim 1.** Any amalgamation in G must be within I.
>
> *Proof.* The vertex α clearly has a unique image and preimage, and thus cannot be amalgamated. For any $x \in A \cup B$, we have $\alpha \in N^+(x) \cap N^-(x)$, so no two nodes in $A \cup B$ can have disjoint preimages or images. Moreover, since $\alpha \notin N^-(i) \cup N^+(i)$ for all $i \in I$, no i and $x \in A \cup B$ can have the same preimage or image, so vertices in $A \cup B$ cannot amalgamate either.
>
> **Claim 2.** If G' is obtained from G by amalgamation, G' satisfies the structure property with $V(G') = \{\alpha\} \cup A \cup I' \cup B$ for some I'.

> *Proof.* Consider any amalgamation u, v in G, resulting in G'. By the first claim, we have $u, v \in I$. It is now clear that amalgamating u and v maintains the structure properties; property (1) and (4) hold trivially, and (2) and (3) could only be violated if new edges were added from $A \cup B$ to $A \cup B$, which is not possible since $u, v \in I$. Thus, G' also satisfies the structure property with the same α, A, and B.

Thus, as initially amalgamations must be within I, and this remains true after any number of amalgamations, the result is shown. □

The other main ingredient of our reduction is a "weight" widget. As we will see in Lemma 6, these widgets act as weighted switches, producing K amalgamations if activated, and at most 1 otherwise. This gap allows us to magnify the decisions which have meaning with respect to the hitting set problem.

Definition 8 (Weight Widget) *Let G satisfy the structure lemma with $V(G) = \{\alpha\} \cup A \cup I \cup B$, and let $K > 0$ be a fixed even integer. Then for $A_* \subseteq A$ and $B_* \subseteq B$, the* weight widget $w = \mathsf{weight}[A_*, B_*]$ *is the following collection of vertices:*

- $A_w = \{a_1 \ldots, a_{K/2}\} \subset A$
- $I_w = \{w_1, \ldots, w_K\} \subset I$
- $B_w = \{b_1, \ldots, b_{K/2}\} \subset B$,

where $A_w \cap A_* = B_w \cap B_* = \emptyset$, and for all $i \in \{1, \ldots, K/2\}$ we have

- $w_{2i-1} : [A_* \cup \{a_1, \ldots, a_{i-1}\},\ b_i]$
- $w_{2i} : [a_i,\ B_* \cup \{b_1, \ldots, b_i\}]$.

Moreover, we require these to be the only images of A_w in I, i.e. $I \cap N^+(a) \subseteq I_w$ for all $a \in A_w$, and similarly for the preimages of B_w. See Figure 3 for an illustration.

If $G = G_0, \ldots, G_k$ is an amalgamation sequence, we say that widget $\mathsf{weight}[A_*, B_*]$ *is* activated *in G_k if $w_1 : [A_*, b_1]$ has amalgamated (possibly with several vertices) to some $v : [A_*, B_* \cup \{b_1\}]$. We call w_1 the* head node *and the nodes $w_i : i > 1$ the* internal nodes *of the weight widget.*

Lemma 6. *If $w = \mathsf{weight}[A_*, B_*]$ is a weight widget in G, then*
1. $v : [A_*, B_*]$ and I_w can amalgamate for a total of K amalgamations.
2. If w is not activated, no internal node of w can amalgamate.

Proof. The first claim is quite straightforward. Let v_k be the head node after k amalgamations, meaning $v_k \in V(G_k)$ and $S_k(v_k) = \{v, w_1, \ldots, w_k\}$. By construction, v_k and w_{k+1} have identical images and disjoint preimages if k is odd, and vice versa if k is even. Thus, after K amalgamations, the entire weight widget has collapsed to $v_K : [A_* \cup A_w, B_* \cup B_w]$.

For the second claim, note that by construction w_k cannot amalgamate if w_{k-1} has not amalgamated. By assumption, no nodes in $I \setminus \{w_2, \ldots, w_K\}$ have preimages in A_w or images in B_w, so we conclude that no internal nodes can amalgamate if

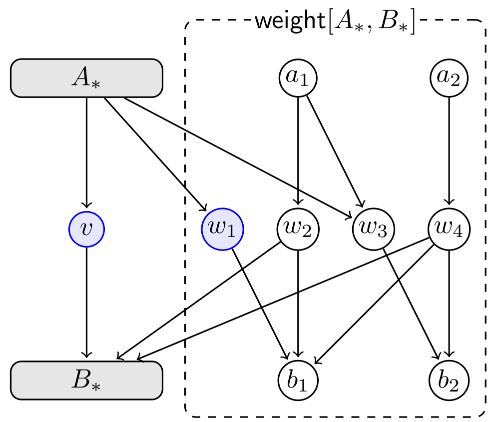

FIGURE 3. A weight widget weight$[A_*, B_*]$ for $K = 4$, and a potential activator v.

the head node w_1 has not amalgamated. Now consider a series of non-activation amalgamations of the head node; since $N^+(w_1) = \{b_1\}$ is a unique image, the resulting amalgamated node must be of the form $[A_*, B' \cup \{b_1\}]$, which will never amalgamated with $w_2 : [a_1, B_* \cup \{b_1\}]$ unless $B' = B_*$, which would mean w was activated. Since no internal node can amalgamate before w_2, we are done. □

We now have the tools to prove our main result.

THEOREM 7. SAP *is NP-complete.*

Proof. We trivially have that SAP is in NP: given a sequence of proposed amalgamations in the form of vertex pairs, we simply check the forward and backward conditions in $O(n^2)$ time, perform the amalgamation in $O(n^2)$ time, and proceed for $O(n)$ iterations, where $n = |V(G)|$. As HittingSet is NP-hard [**20**], it is enough to show a polynomial-time many-one reduction to SAP, in which case we will write SAP $\geq_m$ HittingSet [**13**].

Let collection of sets $\mathcal{C} = \{S_1, \ldots, S_m\}$ and size t be given from HittingSet, and define $U = \cup_i S_i$, $n = |U|$, and $m = |\mathcal{C}|$. Furthermore, we set the parameter $K = 6nm$ for the weight widgets. We now construct a graph G with $V(G) = A \cup I \cup B \cup \{\alpha\}$ in polynomial time as follows.

- Start with $A = [m]$, $I = \emptyset$, $B = U \cup \{\beta\}$ for a new vertex β.
- Add a vertex $v_{is} : [i, s]$ for each S_i and $s \in S_i$. That is, add a node v_{is} to I with $i \to v_{is} \to s$, where $i \in A$ and $s \in B$.
- For each $i \in [m]$, add $v_{i\beta}$ to I with $i \to v_{i\beta} \to \beta$, so that $v_{i\beta} : [i, \beta]$.
- Add weight widget $w =$ weight$[i, \{s, \beta\}] = (A_w, I_w, B_w)$ for each S_i and $s \in S_i$ (i.e., add A_w to A, I_w to I, and B_w to B).
- Add weight widget weight$[\mathrm{hit}(s), s]$ for each $s \in U$.
- Add node α with edges to and from A and B, and add loops to α, A, and B, to satisfy the structure property.

In other words, letting W be the set of all weight widgets added, we will have $A = [m] \cup \bigcup_{w \in W} A_w$, $I = \{v_{is} : i \in [m], s \in S_i\} \cup \{v_{i\beta} : i \in [m]\} \cup \bigcup_{w \in W} I_w$, and $B = U \cup \{\beta\} \cup \bigcup_{w \in W} B_w$.

We will show that there is a hitting set of the given size t if and only if there is an amalgamation sequence for G of length $N \geq (m + n - t)K$. The rough idea behind the reduction is that each vertex v_{is} can choose either "$s \in H$ hitting S_i" or "$s \in U \setminus H$", and this choice is then magnified by the corresponding weight widget. We will see that we can activate $|\mathrm{hit}(H)| + |U \setminus H|$ weight widgets, for a total of $m + (n - t)$ if H is a hitting set of size t, and strictly fewer if no such H exists, by Lemma 4. By construction, K is large enough that non-widget amalgamations are insignificant, from which the result will follow.

First assume that H is a hitting set for $\mathcal{C}$, and $|H| = t$. Then for each $i \in [m]$ there is some $s \in S_i \cap H$, so we can amalgamate $v_{is} : [i, s]$ with $v_{i\beta} : [i, \beta]$ to a new vertex $[i, \{s, \beta\}]$ which by Lemma 6 we can amalgamate with $\mathsf{weight}[i, \{s, \beta\}]$ for K amalgamations. We do this for all $i \in [m]$, so we have mK amalgamations so far. At this point, the vertices in I associated to each $s \in U \setminus H$ remain untouched, so we can amalgamate $\{v_{is} \mid i \in \mathrm{hit}(s)\}$ as they all share the same image and have disjoint preimages. We now have a vertex $[\mathrm{hit}(s), s]$ for each $s \in U \setminus H$, and by Lemma 6 we can amalgamate each with $\mathsf{weight}[\mathrm{hit}(s), s]$ for another $(n - t)K$ amalgamations. In total we thus have at least $(m + n - t)K$ amalgamations.

For the converse, assume there is no hitting set H of size t, and there is an amalgamation sequence for G of length N. Define:

$$\overline{H} = \{s \mid \mathsf{weight}[\mathrm{hit}(s), s] \text{ is activated}\}$$
$$M = \{i \mid \exists s \in S_i \ \mathsf{weight}[i, \{s, \beta\}] \text{ is activated}\}$$

and let $H = U \setminus \overline{H}$. Note that each v_{is} can amalgamate with at most one weight widget, and for each $i \in [m]$ there can be at most one $\mathsf{weight}[i, \{s, \beta\}]$ widget activated. Thus for all v_{is}, either ($i \in M$ and $s \in H$), ($i \notin M$ and $s \notin H$), or ($i \notin M$ and $s \in H$). Thus, we have ($i \in M \implies s \in H$), so

$$M \subseteq \{i \mid \exists\, v_{is}\; s \in H\} = \mathrm{hit}(H). \tag{6}$$

We now count how many amalgamations we could have. By Lemma 6, each activated weight widget leads to at most K amalgamations, for a total of $(|M| + |\overline{H}|)K$ amalgamations. Also by Lemma 6, no internal nodes of the non-activated weight widgets have amalgamated, so the remainder of the amalgamations are restricted to head nodes and $I_{\text{non-weight}} = \{v_{is} : i \in [m], s \in U\} \cup \{v_{i\beta} : i \in [m]\}$, for a total of at most $(n - |\overline{H}|) + (nm - |M|) + |I_{\text{non-weight}}| \leq n + nm + nm + m < K$ additional amalgamations. Thus, we obtain our result:

$$\begin{aligned} N &< (|M| + |\overline{H}| + 1)K \\ &\leq (|\mathrm{hit}(H)| + n - |H| + 1)K \qquad (7) \\ &\leq (m + n - t)K, \qquad (8) \end{aligned}$$

where (7) follows from equation (6), and (8) is by Lemma 4. □

4. *Extension to Edge Shifts*

While we focus on vertex shifts, Theorem 7 can be extended to edge shifts, as we now briefly describe. An edge shift is given by sequences of edges from bi-infinite walks on a directed *multi-graph*, which allows multiple edges between two vertices. Following Lind and Marcus [**24**], we will denote by $\mathsf{i}(e)$ and $\mathsf{t}(e)$ the initial and terminal vertex for edge e. Given a directed multi-graph G with edges $E(G)$ and vertices $V(G)$, we let $\mathcal{A} = E(G)$ and define the corresponding *edge shift* as a subshift $\hat{X}_G \subseteq X_{\text{full}}$, where $x \in \hat{X}_G$ if and only if for all $i \in \mathbb{Z}$ we have $\mathsf{t}(x_i) = \mathsf{i}(x_{i+1})$. As before we define $\hat{N}^-(v)$ and $\hat{N}^+(v)$ to be the in- and out-neighbors of v, now multisets. (E.g., in Figure 4, $\hat{N}^-(cd) = \{\{a, a, b\}\}$.) As with vertex shifts, every SFT is conjugate to some edge shift [**24**].

The analogous form of the amalgamation conditions (2) and (3) in this case simplify: vertices u and v are amalgamation candidates if either $\hat{N}^+(u) = \hat{N}^+(v)$ (forward) or $\hat{N}^-(u) = \hat{N}^-(v)$ (backward), as multisets. The forward amalgamation is then given by combining u and v to a new node w, setting $\hat{N}^+(w) = \hat{N}^+(v)$ and $\hat{N}^-(w) = \hat{N}^-(u) \cup \hat{N}^-(v)$, a multiset union; similarly for the backward amalgamation. See Figure 4 for an illustration. As before, amalgamation produces a conjugacy [**24**, Theorem 2.4.10] and all conjugacies between edge shifts can be represented as a sequence of amalgamations and of their mirror operation, state splittings [**24**, Theorem 7.1.2].

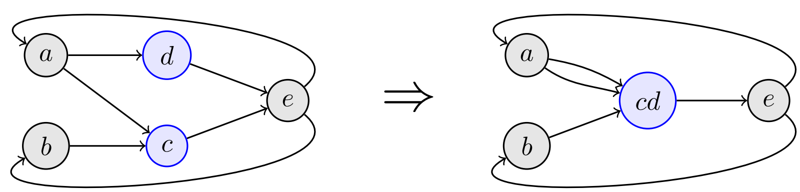

FIGURE 4. A forward edge amalgamation

The construction of Theorem 7 already suffices for edge shifts, as we now argue. First, observe that vertices $A \cup B \cup \{\alpha\}$ all have unique $\hat{N}^+$ and $\hat{N}^-$ multisets (indeed, sets); this follows from the proof of Lemma 5, and the observation that $a \in N^-(a)$ and $b \in N^-(b)$ for all $a \in A$, $b \in B$. Hence, even with the relaxed amalgamation condition, these vertices cannot be eventually amalgamated, by the same argument. Similarly, in Lemma 6, the vertices $w_2, \ldots, w_K$ have unique $\hat{N}^+$ and $\hat{N}^-$ multisets (from inspection of Definition 8), and thus cannot amalgamate immediately with each other. Moreover, the proofs of the two claims of Lemma 6 go through without change: the weight widget can still be activated for K total amalgations, as we have only weakened the amalgamation condition; by uniqueness of $\hat{N}^+$ and $\hat{N}^-$, we still conclude that w_k cannot amalgamate before w_{k-1}; and finally, the argument that w_2 cannot amalgamate unless the widget is activated did not rely on disjointness of preimages, and thus continues to hold. We conclude that SAP is NP-complete over the class of edge shifts as well.

5. *Markov Partitions and Future Work*

This state amalgamation problem arises naturally when studying symbolic dynamics derived from itinerary functions. Following Lind and Marcus [24], consider a continuous invertible $f : M \to M$ on some metric space M, and disjoint open sets $R_a \subseteq M$ for each $a \in \mathcal{A}$ such that $M = \cup_{a\in\mathcal{A}}\overline{R_a}$, where $\overline{R_a}$ denotes the closure of R_a. Letting $U = \cup_{a\in\mathcal{A}} R_a$, we define $S = \cap_{n\in\mathbb{Z}} f^n(U)$ to be the set of points which stay within U in forward and backward time. On such points we may then define the *itinerary* function $\rho : S \to \mathcal{A}^{\mathbb{Z}}$ by $\rho(z)_n = a$ where $f^n(z) \in R_a$; that is, for all $n \in \mathbb{Z}$ and $z \in S$ we have $f^n(z) \in R_{\rho(z)_n}$. The closure of the range of ρ thus forms a subshift $X = \overline{\rho(S)} \subseteq \mathcal{A}^{\mathbb{Z}}$. By construction, X carries geometric meaning, in that each symbol corresponds to a specific region in M: for all $a \in \mathcal{A}$ we have $\rho^{-1}(\{x \in X : x_0 = a\}) \subseteq R_a$.

A particular case of interest is when $\{R_a\}_{a\in\mathcal{A}}$ is a *Markov partition*, meaning that ρ is injective, and the resulting shift X is a vertex shift (1-step SFT), which satisfies the usual Markov property that the set of possible next symbols depends only on the current symbol: for all $a, b \in \mathcal{A}$, either $f(R_a) \cap R_b = \emptyset$ or $f(R_a) \supseteq R_b$. Markov partitions were introduced independently by Berg [4], Adler [2], and Sinai [26, 27], and later studied by Bowen [6] and others as a means of studying continuous maps via symbolic dynamics: under the above conditions, the map $\rho : S \to X$ is a conjugacy [1, Prop. 5.12].

If the number of regions $|\mathcal{A}|$ is too large, one may wish to simplify its representation while preserving this geometric meaning, as well as the Markov property. For example, automated proof systems routinely produce shifts on hundreds of symbols [10, 12]. A natural way to simplify these shifts is by *coarsening* the partition, i.e. combining various regions together and labeling them with the corresponding set of symbols, in such a way that the resulting partition is still Markov. The resulting subshift X' is thus related to X by a 1-block code, whose alphabet $\mathcal{A}'$ corresponds to a partition of $\mathcal{A}$. Naturally, each symbol $a' = \{a_1, \dots, a_k\}$ corresponds to the region $R_{a'} = R_{a_1} \cup \dots \cup R_{a_k}$, so we define $\rho' : S \to (\mathcal{A}')^{\mathbb{Z}}$ analogously, and we still retain the geometric property that $(\rho')^{-1}(\{x \in X' : x_0 = a'\}) \subseteq R_{a'}$. For the resulting partition to retain the Markov property, however, the resulting subshift X' should be conjugate to X and again be a vertex shift, and one way to ensure this is to express the 1-block code as a sequence of state amalgamations.†

Given the above motivation, we mention a potentially easier problem which would still be relevant for studying Markov partitions. Consider a situation where a particular Markov partition P and vertex shift X is conjectured for a continuous map, and in an effort to validate this conjecture, one proves the existence of a different Markov partition P' and vertex shift X', with many more regions/symbols. In this case, the original conjecture can be proved if we can find a 1-block conjugacy from X' to X. In contrast to SAP, however, here one can specify exactly which regions of P' lie in which regions of P, and thus one can construct the block

† Note that preserving conjugacy and retaining the Markov property of the shift is also desirable even when the map ρ is only a semi-conjugacy, as in [10, 12].

map explicitly, and simply ask whether the corresponding sliding block code is a conjugacy. This approach is precisely what we use in previous work [**12**] to verify that a Markov partition conjectured by Davis, MacKay, and Sannami [**9**] is semi-conjugate to the Hénon map at parameter values $(a = 5.4, b = -1)$. (Other sources of such conjectures include pruning theory [**14**].) The above discussion motivates the *state partition problem (SPP)*, whose complexity is unknown: given a graph G and a partition of the vertices $P = \{P_1, P_2, \ldots, P_k\}$, decide whether there exists a sequence of amalgamations in which each P_i is eventually amalgamated.

Successfully coarsening Markov partitions more generally involves finding 1-block conjugacies, which may be a harder problem, in the sense that it may be higher in the polynomial hierarchy or even undecidable. In particular, the complexity of the problems analogous to SAP and SPP would both be of interest: given a vertex shift, find the 1-block conjugacy to the smallest possible vertex shift (measured by number of vertices), and if additionally given an explicit partition of the vertices, decide whether the induced 1-block code is a conjugacy. The complexity of these problems would shed light on the complexity of conjugacy between vertex shifts more generally, which remains open [**7**]. Intuitively, the construction behind Theorem 7 could lead to techniques for encoding undecidable problems in vertex shifts, showing the undecidability of general conjugacy. Finally, as mentioned in the introduction, as conjugacy is decidable for tree shifts of finite type, it would be interesting to study all four of these problems in that setting.

Acknowledgements. I would like to thank Mike Boyle, Doug Lind, Brian Marcus, Omer Tamuz, Sarah Day, and Ronnie Pavlov for their thoughtful comments and encouragement, Dick Karp for his support, and the Symbolic Dynamics Reading Group at CU Boulder for their feedback and enthusiasm.

References

[1] Roy Adler. Symbolic dynamics and Markov partitions. *Bulletin of the American Mathematical Society*, 35(1):1–56, 1998.

[2] Roy L. Adler and Benjamin Weiss. Entropy, a complete metric invariant for automorphisms of the torus. *Proceedings of the National Academy of Sciences*, 57(6):1573–1576, 1967.

[3] Nathalie Aubrun and Marie-Pierre Béal. Tree-shifts of finite type. *Theoretical Computer Science*, 459:16–25, November 2012.

[4] Kenneth Richard Berg. *On the conjugacy problem for K-systems.* Ph.D. Dissertation, University of Minnesota, 1967.

[5] Robert Berger. *The undecidability of the domino problem.* Number 66. American Mathematical Soc., 1966.

[6] Rufus Bowen. Markov partitions for Axiom A diffeomorphisms. *American Journal of Mathematics*, 92(3):725–747, 1970.

[7] Mike Boyle. Open problems in symbolic dynamics. *Contemporary mathematics*, 469:69–118, 2008.

[8] Mike Boyle, Bruce Kitchens, and Brian Marcus. A note on minimal covers for sofic systems. *Proceedings of the American Mathematical Society*, pages 403–411, 1985.

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

[10] S. Day, R. Frongillo, and R. Trevino. Algorithms for rigorous entropy bounds and symbolic dynamics. *SIAM Journal on Applied Dynamical Systems*, 7(4):1477–1506, 2008.

[11] Doris Fiebig, Ulf-Rainer Fiebig, and Nataša Jonoska. Multiplicities of covers for sofic shifts. *Theoretical Computer Science*, 262(1-2):349–375, July 2001.

[12] Rafael Frongillo. Topological Entropy Bounds for Hyperbolic Plateaus of the Henon Map. *SIAM Undergraduate Research Online*, 7, 2014.

[13] Oded Goldreich. *Computational Complexity: A Conceptual Perspective*. Cambridge University Press, April 2008.

[14] Ryouichi Hagiwara and Akira Shudo. An algorithm to prune the area-preserving Hénon map. *Journal of Physics A: Mathematical and General*, 37(44):10521, 2004.

[15] Emmanuel Jeandel and Pascal Vanier. Hardness of conjugacy, embedding and factorization of multidimensional subshifts. *Journal of Computer and System Sciences*, 81(8):1648–1664, 2015.

[16] Aimee Johnson and Kathleen Madden. The decomposition theorem for two-dimensional shifts of finite type. *Proceedings of the American Mathematical Society*, 127(5):1533–1543, 1999.

[17] Natasa Jonoska and Brian Marcus. Minimal presentations for irreducible sofic shifts. *IEEE transactions on information theory*, 40(6):1818–1825, 1994.

[18] Nataša Jonoska. Sofic shifts with synchronizing presentations. *Theoretical Computer Science*, 158(1):81–115, May 1996.

[19] Jarkko Kari. Tiling problem and undecidability in cellular automata. In *Computational Complexity*, pages 3198–3211. Springer, 2012.

[20] Richard M Karp. *Reducibility among combinatorial problems*. Springer, 1972.

[21] K. H. Kim and F. W. Roush. The Williams conjecture is false for irreducible subshifts. *Electron. Res. Announc. Amer. Math. Soc*, 3:105–109, 1997.

[22] Ki Hang Kim and Fred W. Roush. Decidability of shift equivalence. In *Dynamical systems*, pages 374–424. Springer, 1988.

[23] Bruce Kitchens. *Symbolic dynamics: one-sided, two-sided and countable state Markov shifts*. Springer, 1998.

[24] Douglas Lind and Brian Marcus. *An introduction to symbolic dynamics and coding*. Cambridge University Press, 1999.

[25] Klaus Schmidt. Multi-dimensional symbolic dynamical systems. In *Codes, systems, and graphical models*, pages 67–82. Springer, 2001.

[26] Ya G. Sinai. Construction of Markov partitions. *Functional Analysis and Its Applications*, 2(3):245–253, 1968.

[27] Ya G. Sinai. Markov partitions and C-diffeomorphisms. *Functional Analysis and its Applications*, 2(1):61–82, 1968.

[28] Hiroshi Teramoto and Tamiki Komatsuzaki. How does a choice of Markov partition affect the resultant symbolic dynamics? *Chaos: An Interdisciplinary Journal of Nonlinear Science*, 20(3):037113, 2010.

[29] Paul Trow. Determining presentations of sofic shifts. *Theoretical computer science*, 259(1):199–216, 2001.

[30] 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.

[31] R. F. Williams. Classification of subshifts of finite type. *Annals of Mathematics*, 98(1):120–153, 1973.

[32] Jiongxuan Zheng, Joseph D. Skufca, and Erik M. Bollt. Comparing dynamical systems by a graph matching method. *Physica D: Nonlinear Phenomena*, 255:12–21, July 2013.