# ON THE QUANTIZED DYNAMICS OF FACTORIAL LANGUAGES

CHRISTOPHER BARRETT AND EVGENIOS T.A. KAKARIADIS

Abstract. We study local piecewise conjugacy of the quantized dynamics arising from factorial languages. We show that it induces a bijection between allowable words of same length and thus it preserves entropy. In the case of sofic factorial languages we prove that local piecewise conjugacy translates to unlabeled graph isomorphism of the follower set graphs. Moreover it induces an unlabeled graph isomorphism between the Fischer covers of irreducible subshifts. We verify that local piecewise conjugacy does not preserve finite type nor irreducibility; but it preserves soficity. Moreover it implies identification (up to a permutation) for factorial languages of type 1 if, and only if, the follower set function is one-to-one on the symbol set.

## 1. Introduction

The fruitful interplay between Symbolic Dynamics and Operator Algebras was established in the seminal paper of Cuntz-Krieger [**9**]. Following their work, Matsumoto introduced an effective way for associating operators to subshifts and forming Cuntz-Krieger-type C*-algebras [**32**] that were further examined in-depth in a series of papers [**33, 34, 35**]. This theory was revisited with Carlsen [**4, 5**] and a new view was exploited in more generality in [**35**]. These important works clarified a strong connection between intrinsic properties of subshifts with related C*-algebras.

Matsumoto operators follow from a Fock representation that accommodates more structures. Shalit-Solel [**39**] provided such a context for homogeneous ideals in general and established a rigidity programme for the related (non-selfadjoint) tensor algebras. The origins of this framework are traced back to the seminal work of Arveson [**1**]. Since then, a number of rigidity results have appeared in the literature for tensor algebras of graphs or dynamical systems as in the work of Katsoulis-Kribs [**26**] and Solel [**40**], Davidson-Katsoulis [**11, 12**] that supersedes the work of previous authors [**3, 19, 37, 38**], Davidson-Roydor [**13**], Davidson-Ramsey-Shalit [**14, 15**], Dor-On [**16**], Katsoulis-Ramsey [**28**], and the work of the second author with Davidson [**10**] and Katsoulis [**23**]. In this endeavour Davidson-Katsoulis [**12**] developed the notion of piecewise conjugacy for classical systems as the essential level of equivalence obtained

2010 *Mathematics Subject Classification.* 58F03, 47L75, 46L55, 54H20.

*Key words and phrases:* Sofic subshift, follower set graph, partially defined dynamical system.

from tensor algebras. Piecewise conjugacy allows for comparisons of the systems locally and thus is more tractable than (global) conjugacy. Recently it found significant applications to Number Theory and reconstruction of graphs as exhibited in the work of Cornelissen-Marcolli [**7, 8**] and Cornelissen [**6**].

Along this line of research, the second author with Shalit examined tensor algebras of factorial languages in [**25**]. A factorial language $\Lambda^*$ on $d$ symbols is a subset of the free semigroup $\mathbb{F}_+^d$ such that if $\mu \in \Lambda^*$ then every subword of $\mu$ is also in $\Lambda^*$. To fix notation, the operators $T_\mu$ in discussion act on $\ell^2(\Lambda^*)$ and are defined by

$$T_\mu e_\nu := \begin{cases} e_{\mu\nu} & \text{if } \mu\nu \in \Lambda^*, \\ 0 & \text{otherwise,} \end{cases}$$

for $\mu \in \Lambda^*$. As an intermediate step we use the C*-algebra of *checkers*

$$A := \mathrm{C}^*(T_\mu^* T_\mu \mid \mu \in \Lambda^*).$$

The $*$-endomorphisms on $A$ given by $\alpha_i(a) := T_i^* a T_i$ play an important role in the analysis and the system $(A, \alpha_1, \dots, \alpha_d)$ was coined in [**25**] as *the quantized dynamics of* $\Lambda^*$. Two norm-closed subalgebras of $\mathcal{B}(\ell^2(\Lambda^*))$ can be related to the same language $\Lambda^*$:

I. The *$\mathcal{A}$-tensor algebra* $\mathcal{A}_{\Lambda^*} := \overline{\mathrm{alg}}\{I, T_\mu \mid \mu \in \Lambda^*\}$ in the sense of Shalit-Solel [**39**];
II. The *$\mathcal{T}^+$-tensor algebra* $\mathcal{T}_{\Lambda^*}^+ := \overline{\mathrm{alg}}\{a, T_\mu \mid a \in A, \mu \in \Lambda^*\}$ in the sense of Muhly-Solel [**36**].

Matsumoto's C*-algebra is the quotient of $\mathrm{C}^*(T) := \mathrm{C}^*(T_\mu \mid \mu \in \Lambda^*)$ by the compacts $\mathcal{K}$ in $\ell^2(\Lambda^*)$.

Arveson's Programme on the C*-envelope[1] provides a solid pathway for researching possible Cuntz-Krieger-type C*-algebras. For example the natural analogues related to C*-correspondences are exactly the C*-envelopes of the tensor algebras as proven by Katsoulis-Kribs [**27**]. One of the main results in [**25**] states that the C*-algebra that fits Arveson's Programme for both $\mathcal{A}_{\Lambda^*}$ and $\mathcal{T}_{\Lambda^*}^+$ is the quotient of $\mathrm{C}^*(T)$ by the generalized compacts; rather than quotienting by all compacts as is done in Matsumoto's work. In fact the quantized dynamics trigger a dichotomy: the C*-envelope of both $\mathcal{A}_{\Lambda^*}$ and $\mathcal{T}_{\Lambda^*}^+$ is either the quotient by all compacts or it coincides with $\mathrm{C}^*(T)$, depending on whether the quantized dynamics is injective or not. This is in full analogy to what holds for graph C*-algebras where sinks or vertices emitting infinite edges are excluded from the Cuntz-Krieger relations.

Apart from being a starting point for Cuntz-Krieger-type C*-algebras *via* the C*-envelope machinery, both $\mathcal{A}_{\Lambda^*}$ and $\mathcal{T}_{\Lambda^*}^+$ are rigid for factorial languages. It is shown in [**25**] that they encode the factorial language, yet in two essentially different ways:

[1] Arveson's Programme was initiated in [**2**] and established in [**20**]. See also its formulation in [**24**].

(i) The $\mathcal{A}$-tensor algebras provide a complete invariant for the factorial languages up to a permutation of the symbols.
(ii) The $\mathcal{T}^+$-tensor algebras provide a complete invariant for local piecewise conjugacy (l.p.c.) of the quantized dynamics.

However it was left open how l.p.c. reflects the initial data:

(a) How is l.p.c. interpreted in terms of factorial languages?
(b) What properties are (thus) preserved under l.p.c.?
(c) What is the impact on sofic factorial languages?

In the current paper we answer these questions that add on the impact of the rigidity results of [**25**].

Before we move to the description of our results we stress that languages of subshifts form special examples of factorial languages and several constructions apply to this broader context. Thus terminology related to subshifts is extended accordingly to cover general factorial languages, when possible. Unlike to Carlsen [**4**], Matsumoto [**32**] or Krieger [**30**], our study is based on the allowable words rather than the points of an induced subshift. In fact the dynamical system of the backward shift is not explicitly used for the Fock space quantization and thus no connections between l.p.c. and topological conjugacy arise. The results and examples herein show that they are incomparable. On one hand l.p.c. requires for the languages to have the same number of symbols (Definition 4.1) and so it is not implied by topological conjugacy. On the other hand in Example 5.10 we construct a subshift of finite type that is l.p.c. to the even shift, and so l.p.c. does not imply topological conjugacy.

We begin by giving an updated picture of the quantized dynamics (Section 3). We then show that l.p.c. implies a bijection between allowable words of the same length, and thus it respects entropy (Proposition 4.3, Corollary 4.5). The flexibility of l.p.c. can be seen in the form of this bijection; but has its limitations (see Remark 4.2). Most notably l.p.c. does not preserve finite type as shown in Example 5.10, nor irreducibility as shown in Example 5.21. Consequently l.p.c. does not preserve the zeta function (Remark 5.11).

Nevertheless l.p.c. respects soficity (i.e. the C*-algebra of checkers is finite dimensional) where the theory is rich. There is a well known construction of a labeled finite graph, i.e. *the follower set graph*, that gives a presentation of a sofic factorial language. When the language is of finite type then this construction can be described by a (terminating) algorithm (Follower Set Graph Algorithm). Labeled graph isomorphism is equivalent then with the factorial languages being the same up to a permutation of symbols (Proposition 5.6). On the other hand it is the unlabeled graph isomorphism that coincides with l.p.c. (Theorem 5.8). Combining these results with [**25**] we thus prove the following diagram for two sofic factorial languages $\Lambda^*$ and $M^*$:

$$\begin{tikzcd}
\mathcal{A}_{\Lambda^*} \simeq \mathcal{A}_{M^*} \arrow[r, Leftrightarrow] \arrow[d, Rightarrow] & \text{the labeled f.s.g. of } \Lambda^* \text{ and } M^* \text{ are isomorphic.} \arrow[d, Rightarrow] \\
\mathcal{T}^+_{\Lambda^*} \simeq \mathcal{T}^+_{M^*} \arrow[r, Leftrightarrow] & \text{the unlabeled f.s.g. of } \Lambda^* \text{ and } M^* \text{ are isomorphic.}
\end{tikzcd}$$

The follower set graph construction is rather useful in Theoretical Computer Science as the starting point for computing minimal presentations. Such presentations are unique (up to isomorphism) for irreducible two-sided subshifts and are better known as Fischer covers [**17, 18**]. We show that l.p.c. induces an unlabeled graph isomorphism between the Fischer covers of irreducible two-sided subshifts (Corollary 5.19). It is quite interesting to notice though that we achieve these results without inducing a bijection between intrinsically synchronizing words. A weaker bijection between the collections of follower sets of such words is induced. This is depicted in Remark 5.20 where we show the limitations of our arguments. The same obstructions do not allow to apply our arguments and prove (or disprove) that l.p.c. respects the mixing property.

We further investigate cases where the vertical directions in the diagram above can or cannot be equivalences based on the type and the number of symbols. As commented in [**25**] these arrows cannot be reversed in general and we extend this remark for two-sided subshifts in Example 5.12. The key in these counterexamples is that the follower set function is not one-to-one. Apparently this is the only obstruction for type 1 factorial languages. In Theorem 5.13 we show that unlabeled isomorphism for type 1 factorial languages produces a labeled isomorphism if and only if the follower set function is one-to-one on the symbol set. Consequently then, isomorphism of the $\mathcal{A}$-tensor algebras is equivalent to isomorphism of the $\mathcal{T}^+$-tensor algebras. This condition is satisfied by edge shifts with invertible adjacency matrix. Injectivity of the follower set function is not required for type 1 factorial languages on two symbols. In Remark 5.16 we describe how an unlabeled graph isomorphism implies a labeled graph isomorphism of the follower set graphs in these cases. These results depend on the low complexity of the system. However this does not hold when passing to type 2 factorial languages, even when the number of symbols is small (Examples 5.17 and 5.18).

To facilitate comparisons, we developed a program that takes as an input a set of forbidden words on two symbols and gives the follower set graph as an output[2]. The code and the .exe file can be downloaded from the official webpage of the second author, currently at

`http://www.mas.ncl.ac.uk/∼nek29`.

We remark that our principal objective here was to construct a program that computes easily follower set graphs as a check for our examples and counterexamples. Hence we were not (extremely) concerned about complexity or the required memory, but only about the fact that it terminates when the type is finite.

**Acknowledgements.** This project started as a continuation of [**25**] with Orr Shalit. Following his suggestion it was decided for the paper to go with just

[2] We chose to develop this program for the right version of factorial languages, rather than the left we work with here, as it is accustomed in Theoretical Computer Science to concatenate on the right. Nevertheless, the left version follows easily by reversing the words in the input and the output.

two authors. We thank Orr for the numerous comments and corrections on earlier drafts of the paper.

We also thank Vissarion Fisikopoulos for his valuable feedback and corrections on earlier versions of the algorithm and the program that computes the follower set graphs. A friend in need is a friend indeed.

This paper is part of the MMath project of the first author.

## 2. Preliminaries

2.1. **Languages and subshifts.** Let us fix the terminology and notation we will be using throughout the paper. To this end $\mathbb{F}_+^d$ denotes the free semigroup on the symbol set $\Sigma := \{1, \dots, d\}$ with multiplication given by concatenation. For $\mu = \mu_k \dots \mu_1$ in $\mathbb{F}_+^d$ we write $|\mu| := k$ for the length of $\mu$. The empty word $\emptyset$ is by default in $\mathbb{F}_+^d$, it has zero length and plays the role of the unit.

For $\mu, \nu \in \mathbb{F}_+^d$ we say that $\nu$ is a *subword* of $\mu$ if there are $w, z \in \mathbb{F}_+^d$ such that $\mu = w\nu z$. By default the empty word is a subword of every $\mu \in \Lambda^*$. A *(factorial) language* is a subset $\Lambda^*$ of $\mathbb{F}_+^d$ that satisfies the following property:

$$\text{if } \mu \in \Lambda^* \text{ then every subword of } \mu \text{ is in } \Lambda^*.$$

We will simply use the term “language” since we are going to encounter just factorial ones. Without loss of generality we will always assume that all letters of the symbol set are in $\Lambda^*$. Otherwise we view $\Lambda^*$ to be defined on less symbols, i.e. on $\Sigma$ out the symbols that are not in $\Lambda^*$.

Examples of languages arise from subshifts and below we give a brief description. Several elements from the theory of subshifts apply to the broader context of languages and terminology will be extended to cover languages in general. Apart from two-sided we also consider one-sided subshifts. Several results that hold for the two sided version hold also for the one-sided with almost the same proof. We will mainly discuss left subshifts, but similar comments hold for the right subshifts. In order to make sense of the one-sided subshifts and avoid technicalities we make the following convention. We will write the sequences $x = (x_n) \in \Sigma^{\mathbb{Z}_+}$ from right to left, i.e.

$$\dots x_n \dots x_1 x_0 . = x$$

and likewise for elements in $\Sigma^{\mathbb{Z}}$. This is to comply with operator composition which comes by multiplying on the left.

We endow $\Sigma^{\mathbb{Z}_+}$ with the product topology and we fix $\sigma \colon \Sigma^{\mathbb{Z}_+} \to \Sigma^{\mathbb{Z}_+}$ be the backward shift with $\sigma((x_i))_k = x_{k+1}$. With our convention, the map $\sigma$ shifts to the right. The pair $(\Lambda, \sigma)$ is called a *left subshift* if $\Lambda$ is a closed subset of $\Sigma^{\mathbb{Z}_+}$ with $\sigma(\Lambda) \subseteq \Lambda$. Similarly the pair $(\Lambda, \sigma)$ is called a *two-sided subshift* if $\Lambda$ is a closed subset of $\Sigma^{\mathbb{Z}}$ with $\sigma(\Lambda) = \Lambda$.

We write $x_{[m+n-1,m]}$ for the block $x_{m+n-1} \dots x_m$ in $x \in \Lambda$. A word $\mu = \mu_{|\mu|} \dots \mu_1$ is said to *occur* in some (one-sided or two-sided) sequence $x$ if there is an $m$ such that

$$x_{|\mu|-1+m} = \mu_{|\mu|}, \dots, x_m = \mu_1.$$

If a word occurs in some point of $\Lambda$ then it is called *allowable*. The *language of a subshift* $\Lambda$ is defined by

$$\Lambda^* := \{w \in \mathbb{F}_+^d \mid w \text{ occurs in some } x \in \Lambda\}.$$

Since $\Lambda$ is $\sigma$-invariant we have that for every allowable word $\mu$ there exists an $x \in \Lambda$ such that $x_{[|\mu|,0]} = \mu$. We write $\mathcal{B}_n(\Lambda^*)$ for the allowable words of length $n$ in $\Lambda^*$. By following the same arguments as in the two-sided subshifts one can show that if $\Lambda$ defines a left subshift then $\Lambda^*$ is a language such that:

$$\text{for every } \mu \in \Lambda^* \text{ there is a } \emptyset \neq \nu \in \Lambda^* \text{ such that } \nu\mu \in \Lambda^*.$$

Conversely every language with this property defines uniquely a left subshift (see [**39**, Proposition 12.3] and [**25**]).

Subshifts can be described also in terms of forbidden words. Let $\mathfrak{F}$ be a set of words on the symbol set $\Sigma = \{1, \dots, d\}$, and let

$$\Lambda_{\mathfrak{F}} := \{(x_n) \in \Sigma^{\mathbb{Z}_+} \mid \text{ no } \mu \in \mathfrak{F} \text{ occurs in } (x_n)\}.$$

It is known that all two-sided subshifts arise in this way. Likewise this also holds for one-sided subshifts. By setting

$$\mathfrak{F}_k := \{\mu \in \mathfrak{F} \mid \ \mu \text{ does not occur in } (x_n) \in \Lambda, |\mu| \leq k\}$$

we see that $\mathfrak{F}_k \subseteq \mathfrak{F}_{k+1}$ and $\mathfrak{F} = \bigcup_k \mathfrak{F}_k$. Then we have that $\Lambda = \cap_k \Lambda_{\mathfrak{F}_k}$, where the intersection is considered inside the full shift space on $\Sigma$. The elements in $\mathfrak{F}$ are called *the forbidden words of the subshift.* We will call a forbidden word *minimal*, if all of its proper subwords are allowable.

Every set $\mathfrak{F}$ of forbidden words admits a unique basis $F \subseteq \mathfrak{F}$ in the sense that for every $\mu \in \mathfrak{F}$ there are (unique) $\nu, w \in \mathbb{F}_+^d$ and a $\mu' \in F$ such that $\mu = \nu\mu' w$ and $\mu'$ is minimal. We say that $\Lambda$ is a *subshift of finite type* (SFT) if the longest word in the basis of $\mathfrak{F}$ has finite length. We say that an SFT is of *type* $k$ if the longest word in the basis of $\mathfrak{F}$ has length $k+1$. Hence if $\Lambda$ is of type $k$ then any forbidden word of length strictly greater than $k+1$ cannot be minimal.

The notions of forbidden words, minimality, basis and type pass naturally to any language. For example, given a set $\mathfrak{F}$ we can define a language by

$$\Lambda_{\mathfrak{F}}^* = \mathbb{F}_+^d \setminus \{w\mu\nu \mid \mu \in \mathfrak{F}, w, \nu \in \mathbb{F}_+^d\}.$$

Not every language is a language of a subshift but it can be embedded in one by augmenting the symbol set. Suppose that $\Lambda^*$ is defined through a set of forbidden words $\mathfrak{F}$ in $\Sigma$. We introduce a new distinguished symbol $\zeta$ and take the symbol set $\widetilde{\Sigma} = \{1, \dots, d, \zeta\}$. Then *the augmented subshift* $(\widetilde{\Lambda}, \sigma)$ *of* $\Lambda^*$ is the subshift defined by $\mathfrak{F}$ in $\widetilde{\Sigma}^{\mathbb{Z}}$. Since $\zeta$ is not contained in any word in $\mathfrak{F}$ it follows that the language of $\widetilde{\Lambda}$ is

$$\widetilde{\Lambda}^* = \{\zeta^{n_k}\mu_k \dots \mu_2 \zeta^{n_2} \mu_1 \zeta^{n_1} \mid n_1, \dots, n_k \in \mathbb{Z}_+, \mu_1, \dots, \mu_k \in \Lambda^*\}$$

and thus contains $\Lambda^*$.

Recall that a two-sided subshift $(\Lambda, \sigma)$ is called *sofic* if the number of classes in $\Lambda^*$ with respect to the equivalence relation

$$\mu \sim \nu \Leftrightarrow \{w \in \Lambda^* \mid w\mu \in \Lambda^*\} = \{w \in \Lambda^* \mid w\nu \in \Lambda^*\}$$

is finite. Equivalently, if $(\Lambda, \sigma)$ is a factor of an SFT [**17, 41**]. The reader is addressed for example to [**31**, Theorem 3.2.10] for a modern treatment of sofic subshifts. For languages that do not come from subshifts we will use the definition of soficity in terms of the equivalence classes. It is shown in [**25**] that a language $\Lambda^*$ is sofic (resp. of finite type) if and only if its augmented subshift $(\widetilde{\Lambda}, \sigma)$ is sofic (resp. of finite type). This follows by observing that $\emptyset \sim \zeta\mu$ for every $\mu \in \widetilde{\Lambda}^*$.

Every two-sided subshift becomes a compact metric space. Taking the one-sided subshifts to be closed yields the same result in our case. Therefore every sequence in a subshift has a converging subsequence. This often appears in [**31**] as the Cantor's diagonal argument, mainly because metric spaces come later in the presentation of [**31**]. We preserve this terminology to keep connections with Symbolic Dynamics. However it is interesting that this argument works to build the one-sided subshift from a set of predetermined forbidden words. The key is that the one-sided full shift is compact and metrizable with the topology given by

$$\rho(x, y) = \begin{cases} 2 & \text{if } x_0 \neq y_0, \\ 2^{-k} & \text{if } x \neq y \text{ and } k \text{ is maximal so that } x_{[k,0]} = y_{[k,0]}, \\ 0 & \text{if } x = y. \end{cases}$$

2.2. **Fock representation.** We will require some basic theory from Hilbert spaces to show how the quantized dynamics arise from a language. The reader who is not familiar with operator theory may read this subsection in combination with Section 3 where explicit identifications in terms of topological spaces are provided.

Operator algebras associated to subshifts were introduced by Matsumoto [**32**]. Let $\Lambda^*$ be a language on $d$ symbols. Let $H = \ell^2(\Lambda^*)$ and fix the operators $T_i$ such that $T_i e_\mu = e_{i\mu}$ if $i\mu \in \Lambda^*$ or zero otherwise. We fix

$$\mathrm{C}^*(T) := \mathrm{C}^*(I, T_i \mid i = 1, \dots, d).$$

It is convenient to write $T_\mu T_\nu = T_{\mu\nu}$ even when $T_{\mu\nu} = 0$, i.e. $T_{\mu\nu} = 0$ if and only if $\mu\nu \notin \Lambda^*$. We will also write $T_\emptyset = I$. Likewise we write $e_{\mu\nu} = 0$ in $\ell^2(\Lambda^*)$ when $\mu\nu \notin \Lambda^*$. The operators $T_\mu$ satisfy a list of properties:

(i) $T_\mu^* T_\mu$ is an orthogonal projection on $\overline{\operatorname{span}}\{e_\nu \mid \mu\nu \in \Lambda^*\}$.
(ii) $T_\nu T_\nu^*$ is an orthogonal projection on $\overline{\operatorname{span}}\{e_{\nu\mu} \mid \mu \in \Lambda^*\}$.
(iii) If $|\mu| = |\nu|$, then $T_\mu^* T_\nu = 0$ if and only if $\mu \neq \nu$.
(iv) $T_\mu^* T_\mu$ commutes with $T_\nu^* T_\nu$, and with $T_\nu T_\nu^*$.
(v) $T_\mu^* T_\mu \cdot T_i = T_i \cdot T_{\mu i}^* T_{\mu i}$ for all $i = 1, \dots, d$.
(vi) $\sum_{i=1}^d T_i T_i^* + P_\emptyset = I$ where $P_\emptyset$ is the projection on $\mathbb{C}e_\emptyset$.
(vii) The rank one operator $e_\nu \mapsto e_\mu$ equals $T_\mu P_\emptyset T_\nu^*$.

(viii) The ideal $\mathcal{K}(\ell^2(\Lambda^*))$ of compact operators is in $\mathrm{C}^*(T)$.

In **[25]** the second author with Shalit examine several operator algebras related to the operators $T_i$. Among them there are two classes of nonselfadjoint operator algebras:

(i) The tensor algebra $\mathcal{A}_{\Lambda^*}$ in the sense of Shalit-Solel **[39]** is defined as the norm-closed subalgebra of $\mathcal{B}(\ell^2(\Lambda^*))$ generated by $I$ and the $T_i$ for $i = 1, \dots, d$.

(ii) The tensor algebra $\mathcal{T}^+_{\Lambda^*}$ in the sense of Muhly-Solel **[36]** is defined as the norm-closed subalgebra of $\mathcal{B}(\ell^2(\Lambda^*))$ generated by $I$, the $T_i$ for $i = 1, \dots, d$, and the $T_\mu^* T_\mu$ for $\mu \in \Lambda^*$.

The relations above imply that

$$\mathcal{A}_{\Lambda^*} = \overline{\operatorname{span}}\{T_\mu \mid \mu \in \Lambda^*\} \quad \text{and} \quad \mathcal{T}^+_{\Lambda^*} = \overline{\operatorname{span}}\{T_\mu a \mid \mu \in \Lambda^*, a \in A\}$$

for the unital C*-subalgebra $A := \mathrm{C}^*(T_\mu^* T_\mu \mid \mu \in \Lambda^*)$ of $\mathrm{C}^*(T)$.

2.3. **$Q$-projections.** We will be using the projections generated by the $T_i^* T_i$. To this end we introduce the following enumeration. Write all numbers from 0 to $2^d - 1$ by using 2 as a base, but in reverse order. Hence we write $[m]_2 \equiv [m] = [m_1 m_2 \dots m_d]$ so that $2 = [0100 \dots 0]$. Let the (not necessarily one-to-one) assignment

$$[m_1 \dots m_d] \mapsto Q_{[m_1 \dots m_d]} := \prod_{m_i = 1} T_i^* T_i \cdot \prod_{m_i = 0} (I - T_i^* T_i),$$

where $I \in \mathcal{B}(\ell^2(\Lambda^*))$. For example we write

$$Q_0 = Q_{[0 \dots 0]} = \prod_{i=1}^d (I - T_i^* T_i) \quad \text{and} \quad Q_{2^d - 1} = Q_{[1 \dots 1]} = \prod_{i=1}^d T_i^* T_i.$$

The $Q_{[m]}$ are the minimal projections in the C*-subalgebra $\mathrm{C}^*(I, T_i^* T_i \mid i = 1, \dots, d)$ of $\mathrm{C}^*(T)$. Consequently we obtain $\sum_{[m]=0}^{2^d-1} Q_{[m]} = I$. The second author with Shalit have shown in **[25]** the stronger equality (notice that we sum for $[m] \geq 1$ here)

$$T_\mu^* T_\mu = T_\mu^* T_\mu \cdot \sum_{[m]=1}^{2^d - 1} Q_{[m]} = \sum_{[m]=1}^{2^d - 1} Q_{[m]} \cdot T_\mu^* T_\mu$$

for all $\emptyset \neq \mu \in \Lambda^*$. It follows that the unit of $\mathrm{C}^*(T_i^* T_i \mid i = 1, \dots, d)$ coincides with $I \in \mathcal{B}(\ell^2(\Lambda^*))$ if and only if $\Lambda^*$ induces a left subshift [**25**, Lemma 4.4].

## 3. The quantized dynamics on the allowable words

Let $\Lambda^*$ be a language on $d$ symbols. We fix once and for all the unital C*-subalgebra

$$A := \mathrm{C}^*(T_\mu^* T_\mu \mid \mu \in \Lambda^*)$$

of $\mathrm{C}^*(T)$. Then $A = \overline{\cup_l A_l}$ is a unital commutative AF algebra for

$$A_l := \mathrm{C}^*(T_\mu^* T_\mu \mid \mu \in \mathcal{B}_l(\Lambda^*)).$$

The C*-algebra $A$ can be characterized by using $\Lambda^*$. For $l \geq 0$ let $\sim_l$ be the equivalence relation on $\Lambda^*$ given by the rule

$$\mu \sim_l \nu \Leftrightarrow \{w \in \mathcal{B}_l(\Lambda^*) \mid w\mu \in \Lambda^*\} = \{w \in \mathcal{B}_l(\Lambda^*) \mid w\nu \in \Lambda^*\}.$$

Let the discrete space $\Omega_l = \Lambda^* / \sim_l$ and write $[\mu]_l$ for the points in $\Omega_l$. Every $\mu \in \Lambda^*$ splits $\mathcal{B}_l(\Lambda^*)$ into the set of the $w_i \in \mathcal{B}_l(\Lambda^*)$ for which $w_i\mu \in \Lambda^*$ and its complement. There is a finite number of such splittings since $\mathcal{B}_l(\Lambda^*)$ is finite. They completely identify single points in $\Omega_l$, and hence $\Omega_l$ is a (discrete) finite space. Furthermore the mapping

$$\vartheta \colon \Omega_{l+1} \to \Omega_l : [\mu]_{l+1} \mapsto [\mu]_l$$

is a well defined (continuous) and onto map. We can then form the projective limit $\Omega$ by the directed sequence

$$\begin{tikzcd} \Omega_0 & \Omega_1 \arrow[l, "\vartheta"'] & \Omega_2 \arrow[l, "\vartheta"'] & \dots \arrow[l, "\vartheta"'] & \Omega \arrow[l] \end{tikzcd}$$

for which we obtain the following identification. In **[25]** the second author with Shalit have shown that

$$A_l \simeq C(\Omega_l) \quad \text{and} \quad A \simeq C(\Omega).$$

We write $\Omega_{\Lambda^*}$ for $\Omega$ when we want to highlight the language $\Lambda^*$ to which $\Omega$ is related. From now on we will tacitly identify $A_l$ with $C(\Omega_l)$ and $A$ with $C(\Omega)$.

In the sequel we will use the same notation $Q_{[m]}$ for the subspaces of $\Omega$ that correspond to the projections $Q_{[m]}$ of $\mathrm{C}^*(T)$. To make this precise if $[m] = [m_1 \dots m_d]$ is the binary expansion of a number from 0 to $2^d - 1$ then the subspace corresponding to $Q_{[m]}$ consists of the points $[\mu] \in \Omega$ for which:

(i) $i\mu \in \Lambda^*$ for all $i$ with $m_i = 1$; and
(ii) $i\mu \notin \Lambda^*$ for all $i$ with $m_i = 0$.

We record here the following proposition from **[25]** for future reference.

**Proposition 3.1.** **[25]** *Let $\Lambda^*$ be a language on $d$ symbols. Then $\Omega$ is finite if and only if $\Lambda^*$ is a sofic language. If, in particular, $\Lambda^*$ is of finite type $k$ then $A_l = A_k$ for all $l \geq k+1$.*

Therefore if $\Lambda^*$ is sofic then there exists a stabilizing step $k$ for which $A_l = A_k$ for all $l \geq k+1$. When $\Lambda^*$ is in particular of finite type, then we get the following proposition for the possible equivalent classes.

**Proposition 3.2.** *Let $\Lambda^*$ be a language on $d$ symbols. If $\Lambda^*$ is of finite type $k$ then for all $\mu \in \Lambda^*$ with $|\mu| \geq k$ we have that*

$$[\mu_1 \dots \mu_{|\mu|}]_k = [\mu_1 \dots \mu_k]_k.$$

**Proof.** Fix an allowable word $\mu = \mu_1 \dots \mu_k \dots \mu_{|\mu|}$. By the properties of the language, if $w\mu \in \Lambda^*$ then $w\mu_1 \dots \mu_k \in \Lambda^*$ as well. Conversely let $w \in \Lambda^*$ such that $w\mu_1 \dots \mu_k \in \Lambda^*$. To reach contradiction suppose that $w\mu \notin \Lambda^*$. Therefore there is an $n_1$ and an $n_2$ such that

$$\nu := w_{n_1} \dots w_k \, \mu_1 \dots \mu_k \, \mu_{k+1} \dots \mu_{n_2} \notin \Lambda^*.$$

Choose $n_1$ and $n_2$ so that $\nu$ is a minimal forbidden word. Since $\mu \in \Lambda^*$ (hence $\mu_1 \dots \mu_{n_2} \in \Lambda^*$) we have that $w_{n_1} \dots w_k \neq \emptyset$; hence $|w| + 1 - n_1 \geq 1$. Similarly, since $w\mu_1 \dots \mu_k \in \Lambda^*$ (hence $w_{n_1} \dots w_k \mu_1 \dots \mu_k \in \Lambda^*$) we have that $\mu_{k+1} \dots \mu_{n_2} \neq \emptyset$; hence $n_2 \geq k+1$. Therefore $\nu$ is a forbidden word of length at least

$$(|w| + 1 - n_1) + n_2 \geq k + 2.$$

Since $\Lambda^*$ is of type $k$, the word $\nu$ cannot be minimal, which gives the required contradiction. ∎

We define the maps $\alpha_i \colon A \to A$ such that $\alpha_i(a) = T_i^* a T_i$. It is clear that every $\alpha_i$ is a positive map and takes values in $A$ since

$$\alpha_i(T_\mu^* T_\mu) = T_{\mu i}^* T_{\mu i} \in A$$

for all $\mu \in \Lambda^*$. In fact every $\alpha_i$ is a $*$-endomorphism of $A$, since $T_\mu^* T_\mu$ commutes with $T_i T_i^*$ and $T_i T_i^* T_i = T_i$. The $\alpha_i$ induce the required covariant relation

$$a T_i = T_i \alpha_i(a) \text{ for all } a \in A, i = 1, \dots, d.$$

used in the study of $\mathcal{T}_{\Lambda^*}^+$. We use the identification of $A$ with $C(\Omega)$ to get a translation of each $\alpha_i$ as a continuous map $\varphi_i$ partially defined on $\Omega$.

Let $A^i$ be the direct summand $T_i^* T_i A$ of $A$, with unit $T_i^* T_i$. Then $A^i$ is the direct limit of $A_l^i = A_l \cap A^i$, and the corresponding projective limit $\Omega^i$ is determined by the spaces

$$\Omega_l^i := \{[\mu]_l \in \Omega_l \mid i\mu \in \Lambda^*\}$$

and the map

$$\vartheta \colon \Omega_{l+1}^i \to \Omega_l^i : [\mu]_{l+1} \to [\mu]_l.$$

Hence $\alpha_i \colon A \to A^i$ is a unit preserving map from $A = C(\Omega)$ into $A^i := T_i^* T_i A = C(\Omega^i)$, and therefore induces a continuous map $\varphi_i : \Omega^i \to \Omega$.

**Proposition 3.3.** *Let $\Lambda^*$ be a language on $d$ symbols. With the above notation we have that $\alpha_i|_{A_l} \colon A_l \to A_l^i$ is induced by $\varphi_i \colon \Omega_l^i \to \Omega_l$ such that $\varphi_i([\mu]_{l+1}) = [i\mu]_l$.*

**Proof.** We have to show that $\alpha_i(f) = f\varphi_i$ for all $f \in A_l = C(\Omega_l)$. It suffices to do so for $f = T_\mu^* T_\mu$ with $\mu \in \mathcal{B}_l(\Lambda^*)$. To this end we have that

$$\alpha_i(T_\mu^* T_\mu) = T_{\mu i}^* T_{\mu i} = \chi_{\mathcal{A}}$$

for the set $\mathcal{A} = \{[w]_{l+1} \mid \mu i w \in \Lambda^*\}$. On the other hand we have that $T_\mu^* T_\mu = \chi_{\mathcal{B}}$ for the set $\mathcal{B} = \{[w]_l \mid \mu w \in \Lambda^*\}$. Hence we compute

$$\chi_{\mathcal{B}}\left(\varphi_i([w]_{l+1})\right) = \chi_{\mathcal{B}}([iw]_l) = \begin{cases} 1 & \text{if } \mu i w \in \Lambda^*, \\ 0 & \text{otherwise,} \end{cases}$$

which shows that $\chi_{\mathcal{B}}\varphi_i = \chi_{\mathcal{A}} = \alpha_i(\chi_{\mathcal{B}})$. ∎

The universal property of the projective limit implies that this information is enough to describe $\varphi_i$. Indeed we have that

$$\Omega = \{(\omega_n)_{n\geq 0} \mid \omega_n = [\mu]_n, \mu \in \Lambda^*\}.$$

Therefore for $\omega \in \Omega^i$, i.e. $\omega_n = [\mu]_n$ such that $i\mu \in \Lambda^*$, we get

$$\varphi_i(\omega) = ([i\mu]_n).$$

In the particular case when $\Lambda^*$ is sofic let $k$ be the step so that $A_l = A_k$ for all $l \geq k+1$. Then we have that $\Omega_l = \Omega_k$ for $l \geq k+1$, and hence

$$\Omega \simeq \{[\mu]_k \mid \mu \in \Lambda^*\}.$$

The projective limit description gives that the induced $\varphi_i$ is then given by $[\mu]_k \mapsto [i\mu]_k$, for $\mu \in \Lambda^*$ with $i\mu \in \Lambda^*$.

**Definition 3.4.** Let $\Lambda^*$ be a language on $d$ symbols. We call the $(A, \alpha) \equiv (A, \alpha_1, \dots, \alpha_d)$, or alternatively the $(\Omega, \varphi) \equiv (\Omega, \varphi_1, \dots, \varphi_d)$, the *quantized dynamics of* $\Lambda^*$.

**Remark 3.5.** The intersection of the subspaces corresponding to the $T_\mu^* T_\mu$ is always non-empty since $\prod_{\mu\in\Lambda^*} T_\mu^* T_\mu e_\emptyset = e_\emptyset$. In fact the intersection consists of the single point $([\emptyset]_n) \in \Omega$. It is evident that $([\emptyset]_n)$ is in the domain of all $\varphi_\mu$ with $\mu \in \Lambda^*$.

## 4. Local piecewise conjugacy

Let $\Lambda^*$ and $M^*$ be languages. Fix their associated quantized dynamics $(\Omega_{\Lambda^*}, \varphi)$ and $(\Omega_{M^*}, \psi)$, and let $Q$ and $P$ be the corresponding systems of projections. Recall that a $*$-isomorphism $\gamma\colon C(\Omega_{M^*}) \to C(\Omega_{\Lambda^*})$ induces a homeomorphism $\gamma_s\colon \Omega_{\Lambda^*} \to \Omega_{M^*}$ on the spectra.

We have the following notion of conjugacy for the quantized dynamics. For convenience we use the notation

$$\operatorname{supp}[m] := \{i \in \{1, \dots, d\} \mid m_i = 1\}$$

where $[m] = [m_1 \dots m_d]$ is the binary expansion of $m \in \{0, 1, \dots, 2^d - 1\}$.

**Definition 4.1.** We say that the systems $(\Omega_{\Lambda^*}, \varphi)$ and $(\Omega_{M^*}, \psi)$ are *$Q$-$P$-locally piecewise conjugate* if:

(i) there exists a homeomorphism $\gamma_s\colon \Omega_{\Lambda^*} \to \Omega_{M^*}$; and
(ii) for every $x \in Q_{[m]}$ there is a neighbourhood $x \in \mathcal{U} \subseteq Q_{[m]}$ and an $[n] \in \{0, 1, \dots, 2^d - 1\}$ such that $|\operatorname{supp}[n]| = |\operatorname{supp}[m]|$, $\gamma_s(\mathcal{U}) \subseteq P_{[n]}$, and

$$\gamma_s \varphi_i|_{\mathcal{U}} = \psi_{\pi(i)} \gamma_s|_{\mathcal{U}},$$

for a bijection $\pi\colon \operatorname{supp}[m] \to \operatorname{supp}[n]$.

Equivalently, every $Q_{[m]} \subseteq \Omega_{\Lambda^*}$ has an open cover $\{\mathcal{U}_\pi\}_\pi$ indexed by the one-to-one correspondences $\pi\colon \operatorname{supp}[m] \to \{1, \dots, d\}$ such that $\gamma_s(\mathcal{U}_\pi) \subseteq P_{[n]}$ for all $[n]$ with $\operatorname{supp}[n] = \pi(\operatorname{supp}[m])$, and $\gamma_s \varphi_i|_{\mathcal{U}_\pi} = \psi_{\pi(i)} \gamma_s|_{\mathcal{U}_\pi}$. We do not exclude the case where $\mathcal{U}_\pi = \emptyset$ for some $\pi$. As an immediate consequence if an $\omega \in Q_{[m]}$ is in the intersection of $\mathcal{U}_\pi$ with $\mathcal{U}_{\pi'}$ then

$$\psi_{\pi(i)}\gamma_s(\omega) = \gamma_s\varphi_i(\omega) = \psi_{\pi'(i)}\gamma_s(\omega).$$

for all $i \in \operatorname{supp}[m]$.

**Remark 4.2.** We can use local piecewise conjugacy to define inductively maps $f_n \colon \mathcal{B}_n(\Lambda^*) \to \mathcal{B}_n(M^*)$. For every step we start at the point $\omega := ([\emptyset]_n)$ which is in the intersection of all subspaces corresponding to $T_\mu^* T_\mu$ for $\mu \in \Lambda^*$. By definition there exists a neighborhood $\mathcal{U}_\pi \subseteq Q_{[1\dots1]}$ containing $\omega$ such that

$$\gamma_s\varphi_i(\omega) = \psi_{\pi(i)}\gamma_s(\omega) \text{ for all } i = 1, \dots, d.$$

We write $\pi_{\emptyset,0}$ for the bijection $\pi$ and set $f_1 = \pi_{\emptyset,0}$. Consequently we obtain

$$\gamma_s\varphi_{\mu_1}(\omega) = \psi_{\pi_{\emptyset,0}(\mu_1)}\gamma_s(\omega).$$

For $f_2$ let a word $\mu = \mu_2\mu_1 \in \mathcal{B}_2(\Lambda^*)$. Since $\mu_2\mu_1 \in \Lambda^*$ then $\omega$ is in the domain of $\varphi_{\mu_2}\varphi_{\mu_1} = \varphi_{\mu_1\mu_2}$. We apply the same argument for $\omega_1 = \varphi_{\mu_1}(\omega)$ and find a bijection $\pi_{\mu,1}$ coming from the neighborhood $\mathcal{U}_{\pi_{\mu,1}}$ of $\omega_1$ such that

$$\gamma_s\varphi_{\mu_2}(\varphi_{\mu_1}(\omega)) = \psi_{\pi_{\mu,1}(\mu_2)}\gamma_s(\varphi_{\mu_1}(\omega)) = \psi_{\pi_{\mu,1}(\mu_2)}\psi_{\pi_{\emptyset,0}(\mu_1)}\gamma_s(\omega).$$

In particular we get that $\gamma_s(\omega)$ is in the domain of

$$\psi_{\pi_{\mu,1}(\mu_2)}\psi_{\pi_{\emptyset,0}(\mu_1)} = \psi_{\pi_{\mu,1}(\mu_2)\pi_{\emptyset,0}(\mu_1)}$$

and hence the word $\pi_{\mu,1}(\mu_2)\pi_{\emptyset,0}(\mu_1)$ is in $M^*$. This procedure gives a well defined map $f_2 \colon \mathcal{B}_2(\Lambda^*) \to \mathcal{B}(M^*)$. The same argument applies for a word $\mu_1$ of length $n$ (instead of just a letter) and a letter $\mu_2$, and gives a map $f_n \colon \mathcal{B}_n(\Lambda^*) \to \mathcal{B}_n(M^*)$ that is defined by

$$f_n(\mu_n \dots \mu_1) := \pi_{\mu,n-1}(\mu_n) \dots \pi_{\emptyset,0}(\mu_1).$$

The notation $\pi_{\mu,k} \equiv \pi_{\mu_k}$ denotes that this bijection comes from the neighborhood $\mathcal{U}_{\pi_{\mu_k}}$ of the point $\varphi_{\mu_k \dots \mu_1}(\omega)$.

Notice that each $f_n$ depends on the point $\omega := ([\emptyset]_n)$. However it also depends on the orbit of $\omega$. For a word $\mu = \mu_n \dots \mu_1 \in \Lambda^*$ we have to keep track where we are at under $f_{n-1}$, since the $n$th bijection depends on where $\varphi_{\mu_{n-1}} \cdots \varphi_{\mu_1}(\omega)$ sits. It is clear that

$$f_n(\mu_n \dots \mu_1) = \pi_{\mu,n-1}(\mu_n) f_{n-1}(\mu_{n-1} \dots \mu_1),$$

but in general $f_n(\mu_n \dots \mu_1)$ may be different than $f_1(\mu_n) f_{n-1}(\mu_{n-1} \dots \mu_1)$. We present such a case in Example 5.10.

Local piecewise conjugacy respects certain properties of languages. This is very pleasing as several invariants for the usual topological conjugacy of subshifts depend on these data.

**Proposition 4.3.** *If $\Lambda^*$ and $M^*$ are locally piecewise conjugate languages then the maps $f_n$ defined in Remark 4.2 are bijections. Consequently $|\mathcal{B}_n(\Lambda^*)| = |\mathcal{B}_n(M^*)|$ for all $n \in \mathbb{N}$.*

**Proof.** First we show that the $f_n$ are one-to-one. To this end suppose that

$$\pi_{\mu,n-1}(\mu_n)\dots\pi_{\emptyset,0}(\mu_1)=\pi_{\nu,n-1}(\nu_n)\dots\pi_{\emptyset,0}(\nu_1).$$

Then we get that $\pi_{\mu,i-1}(\mu_i)=\pi_{\nu,i-1}(\nu_i)$ for all $i=1,\dots,n$. In particular we have that $\pi_{\emptyset,0}(\mu_1)=\pi_{\emptyset,0}(\nu_1)$ and therefore $\mu_1=\nu_1$. Consequently we get that $\pi_{\mu,1}=\pi_{\nu,1}$, hence $\mu_2=\nu_2$. Inductively we have that $\mu_i=\nu_i$ for all $i=1,\dots n$. Therefore we get $|\mathcal{B}_n(\Lambda^*)|\leq|\mathcal{B}_n(M^*)|$. By symmetry we obtain equality which implies that the $f_n$ are bijections. ∎

**Remark 4.4.** Since the functions $f_n$ are bijections then $\gamma_s(\omega)$ is in the domain of all $\psi_\nu$ for $\nu\in M^*$, for $\omega=([\emptyset]_n)$. Hence $\gamma_s$ fixes the points $([\emptyset]_n)$ of $\Omega_{\Lambda^*}$ and $\Omega_{M^*}$.

The entropy of a language $\Lambda^*$ is given by $h(\Lambda^*):=\lim_n n^{-1}\log_2|\mathcal{B}_n(\Lambda^*)|$. The following corollary is immediate.

**Corollary 4.5.** *Local piecewise conjugate languages share the same entropy.*

Local piecewise conjugacy respects the class of sofic languages. However, as we will see in Example 5.10, it does not preserve the class of SFT's.

**Proposition 4.6.** *Let $\Lambda^*$ and $M^*$ be locally piecewise conjugate languages. If $\Lambda^*$ is sofic then so is $M^*$.*

**Proof.** Immediate by Proposition 3.1. ∎

## 5. Applications to the follower set graph

The quantized dynamics can be described by a finite graph in the case of the sofic languages. Fix a sofic language $\Lambda^*$ on $d$ symbols. For every $\mu\in\Lambda^*$ let

$$F(\mu):=\{w\in\Lambda^*\mid w\mu\in\Lambda^*\}$$

be the follower set of $\mu$. We write $F_{\Lambda^*}(\mu)$ when we want to highlight the language to which we refer[3]. Recall that the elements in $\Omega$ are of the form $([\mu]_n)$ for $\mu\in\Lambda^*$. Since $([\mu]_n)=([\nu]_n)$ if and only if $F(\mu)=F(\nu)$ we have a bijection between $\Omega$ and the set $\{F(\mu)\mid\mu\in\Lambda^*\}$. By definition soficity is equivalent to $\Omega$ being finite. In this case Proposition 3.1 provides the existence of a stabilizing step $k$ such that $\Omega_l=\Omega_k$ for all $l\geq k+1$; i.e. every $F(\mu)$ can be identified with $[\mu]_k$.

The *follower set graph* $\mathfrak{G}_{\Lambda^*}=(G_{\Lambda^*},\mathfrak{L})$ of a sofic language on $d$ symbols is an edge-labeled graph, where $\mathfrak{L}$ is a colouring map from the edge set $G^{(1)}_{\Lambda^*}$ onto $\{1,\dots,d\}$, defined as follows:

(i) The vertices of $G_{\Lambda^*}$ are given by the follower sets.
(ii) We draw an edge labeled $i$ from $F(\mu)$ to $F(i\mu)$ if and only if $F(i\mu)\neq\emptyset$.

[3] We will rarely use this notation, as the language will be clear from the context.

Therefore we have an edge (labeled $i$) between $\omega = [\mu]_k$ and $\omega' = [\nu]_k$ if $[i\mu]_k = [\nu]_k$, i.e. if $\varphi_i(\omega) = \omega'$. It follows that the labeled graph gives a representation of the quantized dynamics. Notice that this procedure gives also the follower set graph in the case of a two-sided subshift [**31**, Section 3.2] (when everything is written in the opposite direction).

Notice that we do not use soficity for the construction of the follower set graph and one may be tempted to produce a follower set graph for any language. However this is not the right thing to do. As we saw in Section 3, the spectrum of $A$ is totally disconnected in general and configuring the dynamics with a discrete structure would not comply with the topology. By Proposition 3.1 it is exactly when $\Lambda^*$ is sofic that the spectrum is discrete and we are able to do so without losing information.

**Example 5.1.** Let $\Lambda^*$ be of type 1. Then $\Omega \simeq \Omega_1 = \{[i_1]_1, \dots, [i_r]_1\}$ for some symbols $i_1, \dots, i_r \in \{1, \dots, d\}$. For the follower set graph we have an edge labeled $j$ from $[i]_1 = F(i)$ to $[ji]_1 = [j]_1 = F(j)$ whenever $ji \in \Lambda^*$. Notice here the use of Proposition 3.2.

The case of type 1 languages requires less information to store, as the label of an edge coincides with the one-lettered word that labels the vertex where the edge terminates.

**Remark 5.2.** There is a strong connection between the follower set graph of a sofic language $\Lambda^*$ and that coming from its augmented subshift $(\widetilde{\Lambda}, \sigma)$. Suppose that $\Lambda^*$ is on $d$ symbols and $\mathfrak{G}$ is its follower set graph. Recall that $F(\emptyset) = F(\zeta\mu)$ for every $\mu \in \widetilde{\Lambda}^*$ for the added distinguished symbol $\zeta$. Therefore the follower set graph of $\widetilde{\Lambda}^*$ contains $\mathfrak{G}$, shares the same vertex set, and contains in addition edges from every vertex to $F(\emptyset)$ labeled by $\zeta$.

When the language is of finite type then we can find a finite set of forbidden words that describes it. By using this, we can determine the follower set graph in a finite number of steps. Indeed the dual of Proposition 3.2 suggests that the determination of every $F(\mu)$ can be achieved in finite steps, even though $F(\mu)$ may be infinite in principle.

**Proposition 5.3.** *Let $\Lambda^*$ be a language of finite type $k$. For every $\emptyset \neq \mu \in \Lambda^*$ and $w_n \dots w_1 \in \Lambda^*$ with $n \geq k$ we have that*

$$w\mu \in \Lambda^* \quad \textit{if and only if} \quad w_k \dots w_1 \mu \in \Lambda^*.$$

**Proof.** The proof is the same to that of Proposition 3.2 once the words are reversed. ∎

Therefore, in order to distinguish between the $F(\mu)$ it suffices to distinguish between the finite sets

$$\{w \in \mathcal{B}_k(\Lambda^*) \mid w\mu \in \Lambda^*\}.$$

Consequently we obtain the following algorithm for constructing the follower set graph of a language of type $k$.

**Follower Set Graph Algorithm.** Let $\Lambda^*$ be a language of finite type and let $\mathcal{F}$ be a finite set of forbidden words that defines $\Lambda^*$. The algorithm takes $\mathcal{F}$ as input and has output the follower set graph. We have two cases:

∎ **Case 1.** If $F = \emptyset$ then $\Lambda^* = \mathbb{F}_+^d$ and its follower set graph is the Hawaiian ring on $d$ edges (lines 1–9 in the pseudocode).

∎ **Case 2.** If $F$ is non-empty we write

$$k := \max\{|\mu| \mid \mu \in \mathcal{F}\} - 1.$$

Then $\Lambda^*$ is a language of type $k$ (lines 10–16 in the pseudocode). We then generate the set $\mathcal{B}_k(\Lambda^*)$ of allowable words of length $k$ (lines 17–24 in the pseudocode). We then proceed to forming the follower set graph in two steps:

- **Step 1:** Determining the vertex set $\{F(\mu) \mid \mu \in \Lambda^*\}$ (lines 25–44 in the pseudocode). It suffices to compute the $F(\mu)$ for $|\mu| \leq k$ (Proposition 3.2). Nevertheless we will also compute the $F(\mu)$ for $|\mu| = k + 1$. This will be helpful for writing the edges. To this end we construct a table indexed by $\mathcal{B}_k(\Lambda^*)$ whose entries show whether an allowable word can follow another one. If two rows are the same then the follower sets of the corresponding words determine the same vertex on the graph. It is convenient to multi-label a vertex by using all the $F(\mu)$, for $|\mu| \leq k + 1$, that coincide.

- Form a table indexed by $\mathcal{B}_k(\Lambda^*)$ (followerTable). We fix an enumeration $\{\mu_1, \dots, \mu_N\}$ of these words, including the void word which we set to be $\mu_1$. For the $(\mu_i, \mu_j)$ entry of the table write $\mu_j\mu_i$. If $\mu_j\mu_i \in \Lambda^*$ then write TRUE for the entry; otherwise write FALSE. In this way we create a second table (truthTable). By Proposition 5.3 these finite entries are sufficient for identifying the follower sets.

- Compute the $F(\mu)$ for $\mu$ of length $k + 1$ (lines 45–56 in the pseudocode). If $\mu$ is forbidden then do nothing. If $\mu = \mu_1\mu_k \dots \mu_{k+1}$ is allowable then add the label $F(\mu)$ to the vertex that has $F(\mu_1 \dots \mu_k)$ among its labels.

- **Step 2:** Determining the edges of the follower set graph (lines 57–69 in the pseudocode). We read the rows of the truthTable from top to bottom. If we pass to a row that is the same with one already read, we move to the next (i.e. we identify the words with the same follower sets.)

- Start from $\mu_l = \mu_1 = \emptyset$ and repeat for $l = 2, \dots, N$. Let $\mu_l$ be a word from $\{\mu_1, \dots, \mu_N\}$. Start with $i = 1$ and repeat for $i = 2, \dots, d$. Write an edge labeled $i$ from the vertex with label $F(\mu_l)$ to the vertex with label $F(i\mu_l)$ if there is a vertex that contains the label $F(i\mu_l)$. Otherwise do nothing and go to $i + 1$. After we finish for $i = d$ (the last step), we repeat for $\mu_{l+1}$.

**Remark 5.4.** In the next pages we give the pseudocode for the algorithm. Notice that in the output each node is a set of words, and each word associated with a node generates the same follower set. We remind that we were not concerned about the technical features of this algorithm, but more about that it does terminate.

**Algorithm:** Generate FollowerSetGraph(F, symbolSet)

```
Input  : $F$ = the list of forbidden words generating the language.
         $symbolSet = \{1, \dots, d\}$.
Output: $Graph = (Nodes, Edges)$ where:
         – each node in $Nodes$ is the set of words of length less than or equal
         to $k + 1$ which have the same follower set; and
         – each edge in $Edges$ is of the form (source, destination, label).

// Deal with the case that $F$ is the empty set.
if $F$ is empty then
    initialize $node$ as a list containing $symbolSet$ and the empty word
    initialize $nodes$ as a list containing only $node$
    initialize $edges$ as empty list
    foreach $letter \in symbolSet$ do
        insert $(node, node, letter)$ into $edges$
    end
    return$(nodes, edges)$
end
// If $F$ is non-empty find value of $k$ from $F$.
initialize $maxLength$ as 1
foreach $word \in F$ do
    if length of $word > maxLength$ then
        $maxLength \longleftarrow$ length of $word$
    end
end
initialize $k$ as $maxLength - 1$
// Generate $mu$, the set of allowed words of length at most $k$.
initialize $mu$ as a list containing only the empty word
foreach $n \in \{1, 2, \dots, k\}$ do
    foreach $word \in$ set of all possible words over $symbolSet$ of length $n$ do
        if $word$ contains no element of $F$ as a subword then
            insert $word$ into $mu$
        end
    end
end
// Form tables.  Indexed by base word and concatenated word.
foreach word $a \in mu$ do
    foreach word $b \in mu$ do
        $followerTable[a][b] \longleftarrow$ concatenate $b$ with $a$
        if $followerTable[a][b]$ contains no element of $F$ as a subword then
            $truthTable[a][b] \longleftarrow True$
        else
            $truthTable[a][b] \longleftarrow False$
        end
    end
end
```

**Algorithm:** Generate FollowerSetGraph(F, symbolSet), continued

```
// Each node of the graph is a set of words sharing the same
   follower set.
initialize $Nodes$ as empty list;
foreach $row \in$ unique rows of $truthTable$ do
    initialize $node$ as an empty list;
    foreach word $a \in mu$ do
        if $truthTable[a] = row$ then
            insert $a$ into $node$;
        end
    end
    insert $node$ into $Nodes$;
end
// Add words of length $k+1$ to the appropriate node.
foreach $node \in Nodes$ do
    foreach $word \in node$ do
        if length of $word = k$ then
            foreach $letter \in symbolSet$ do
                $newWord \longleftarrow$ concatenate $word$ with $letter$;
                if $newWord$ contains no element of $F$ as a subword then
                    insert $newWord$ into $node$;
                end
            end
        end
    end
end
// Create edges.
initialize $Edges$ as empty list;
foreach $node \in Nodes$ do
    foreach $letter \in symbolSet$ do
        $newWord \longleftarrow$ concatenate $letter$ with any word in $node$ of length less
          than $k+1$;
        // Find which node $newWord$ belongs to, if any
        foreach $node' \in Nodes$ do
            if $newWord \in node'$ then
                $edge \longleftarrow (node, node', letter)$;
                insert $edge$ into $Edges$;
                break;
            end
        end
    end
end
return$(Nodes, Edges)$;
```

Let us illustrate with an example how the follower set graph is constructed from the Follower Set Graph Algorithm.

**Example 5.5.** Let the symbol set be $\{0,1\}$. Let $\Lambda^*$ be the language defined by the words $\mathcal{F} = \{101, 110\}$. We form the followerTable of the Follower Set Graph Algorithm with all the words of length at most two. As we add words on the left, it is convenient to put the labels at the right of the followerTable:

| 11 | 01 | 10 | 00 | 1 | 0 | $\emptyset$ | |
|---|---|---|---|---|---|---|---|
| 11 | 01 | 10 | 00 | 1 | 0 | $\emptyset$ | $\emptyset$ |
| 110 | 010 | 100 | 000 | 10 | 00 | 0 | 0 |
| 111 | 011 | 101 | 001 | 11 | 01 | 1 | 1 |
| 1100 | 0100 | 1000 | 0000 | 100 | 000 | 00 | 00 |
| 1110 | 0110 | 1010 | 0010 | 110 | 010 | 10 | 10 |
| 1101 | 0101 | 1001 | 0001 | 101 | 001 | 01 | 01 |
| 1111 | 0111 | 1011 | 0011 | 111 | 011 | 11 | 11 |

For simplicity we can just indicate the forbidden words by a black square in the truthTable of the Follower Set Graph Algorithm:

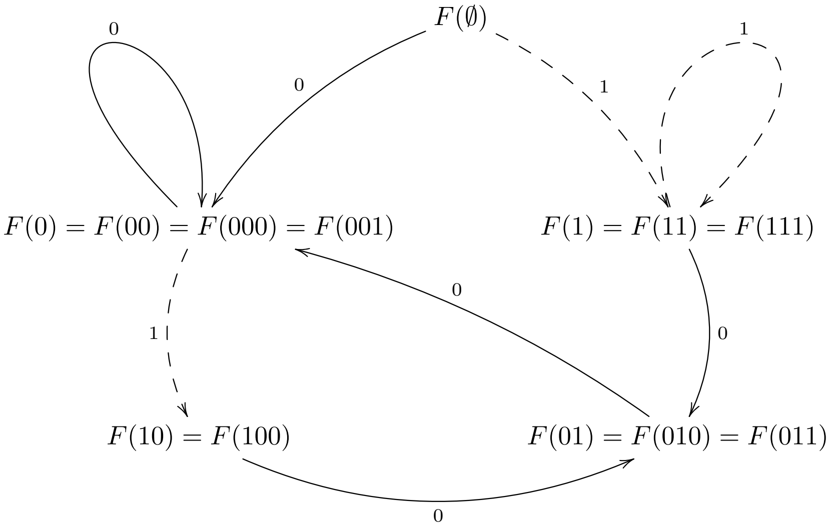
| 11 | 01 | 10 | 00 | 1 | 0 | $\emptyset$ | |
|---|---|---|---|---|---|---|---|
| | | | | | | | $\emptyset$ |
| ■ | | | | | | | 0 |
| | | ■ | | | | | 1 |
| ■ | | | | | | | 00 |
| ■ | ■ | ■ | | ■ | | | 10 |
| ■ | ■ | | | ■ | | | 01 |
| | | ■ | | | | | 11 |

This gives the following five vertices:

$$\begin{aligned}
&F(\emptyset),\\
&F(0) = F(00) = F(000) = F(001),\\
&F(1) = F(11) = F(111),\\
&F(10) = F(100),\\
&F(01) = F(010) = F(011),
\end{aligned}$$

where we used Proposition 3.2 to provide the classes for the allowable words of length 3. Notice here that $F(101)$ and $F(110)$ are meaningless as 101 and

110 are forbidden words. Then the follower set graph is:



By definition, the follower set graph is *(left-)resolving*, i.e. different edges with the same source carry different labels. Recall that a graph is called *follower-separated* if distinct vertices have distinct follower sets (set of paths starting on the vertex). This is also a property that the follower set graph has. Indeed, if two vertices, say $F(\mu)$ and $F(\nu)$ have the same follower sets on the graph then every path $w$ that starts from $\mu$ can also start at $\nu$. This shows that $w\mu \in \Lambda^*$ if and only if $w\nu \in \Lambda^*$ for all $w \in \Lambda^*$, hence $F(\mu) = F(\nu)$.

Resolving graphs go by the name of *Shannon graphs* in the literature. Another type of a Shannon graph for subshifts is produced through the *Krieger cover* [**30**]. The Krieger cover is constructed by taking the follower sets on one-way infinite words. This is the important difference with the follower set graph here, as we consider the vertices to be the follower sets on the *finite* words. For a nice exposition (among others) on the Krieger cover, the reader can see also [**21**].

Given two languages we can consider their labeled graphs or the ambient unlabeled graphs. Graph isomorphism within each class is translated to a different level of equivalence.

5.1. **Labeled graph isomorphism.** Let $\mathfrak{G}_1 = (G_1, \mathfrak{L}_1)$ and $\mathfrak{G}_2 = (G_2, \mathfrak{L}_2)$ be two edge-labeled graphs such that the labels $\mathfrak{L}_1, \mathfrak{L}_2$ take values on the same symbol set $\{1, \dots, d\}$. The labeled graphs $\mathfrak{G}_1$ and $\mathfrak{G}_2$ are called *labeled graph isomorphic* if there is a graph isomorphism $(\partial\gamma, \gamma)\colon G_1 \to G_2$ and a bijection $\pi$ on the symbol sets such that

$$\mathfrak{L}_2(\gamma(e)) = \pi(\mathfrak{L}_1(e)) \text{ for all edges } e.$$

(We use the notation $\partial\gamma\colon G_1^{(0)} \to G_2^{(0)}$ and $\gamma\colon G_1^{(1)} \to G_2^{(1)}$ for the graph isomorphism.)

**Proposition 5.6.** *Let $\Lambda^*$ and $M^*$ be sofic languages. Then their follower set graphs are isomorphic if and only if there is a bijection on their symbol sets that preserves the allowable words.*

**Proof.** Suppose there is an edge-labeled graph isomorphism. By construction there is at least one vertex, i.e. the vertex $F_{\Lambda^*}(\emptyset)$ (resp. $F_{M^*}(\emptyset)$), that emits edges of all labels. Hence $\max\{s^{-1}(v) | v \text{ vertex}\}$ coincides with the number of the symbols, thus $d = d'$. The graph isomorphism identifies $F_{\Lambda^*}(\emptyset)$ with some $F_{M^*}(\nu)$. Then the edge $e = (F_{\Lambda^*}(\emptyset), F_{\Lambda^*}(i))$ labeled $i$ corresponds to the edge

$$\gamma(e) = (F_{M^*}(\nu), F_{M^*}(\pi(i)\nu))$$

labeled $\pi(i)$. Suppose $ji \in \Lambda^*$. Then the edge $f = (F_{\Lambda^*}(i), F_{\Lambda^*}(ji))$ labeled $j$ corresponds to the edge

$$\gamma(f) = (F_{M^*}(\pi(i)\nu), F_{M^*}(\pi(j)\pi(i)\nu))$$

labeled $\pi(j)$. Therefore we get that $\pi(j)\pi(i) \in M^*$. Hence $\pi$ respects the allowable words of length 2. Inductively we get that permutation of the edges along the vertices extends to a match on the allowable words of any length. Notice here that $\pi^{-1}$ is the associated labeling for the inverse graph isomorphism.

Conversely, the existence of a bijection $\pi$ on the symbol sets implies that $d = d'$. Furthermore $\pi$ extends to the allowable words such that $\pi(\mu\nu) = \pi(\mu)\pi(\nu)$. We define the edge-labeled graph homomorphism by

$$\partial\gamma(F_{\Lambda^*}(\mu)) := F_{M^*}(\pi(\mu)).$$

The map $\partial\gamma$ is well defined. Indeed if $F_{\Lambda^*}(\mu) = F_{\Lambda^*}(\nu)$ then we get that

$$\{w \in \Lambda^* \mid w\mu \in \Lambda^*\} = \{w \in \Lambda^* \mid w\nu \in \Lambda^*\}.$$

Applying $\pi$ and working backwards we have that $F_{M^*}(\pi(\mu)) = F_{M^*}(\pi(\nu))$. It follows that $\partial\gamma$ is also one-to-one, hence a bijection due to the properties of $\pi$. If there is an edge $e = (F_{\Lambda^*}(\mu), F_{\Lambda^*}(i\mu))$ labeled $i$ then we have that $i\mu \in \Lambda^*$ and thus $\pi(i\mu) = \pi(i)\pi(\mu) \in M^*$. Therefore there is an edge labeled $\pi(i)$ connecting $F_{M^*}(\pi(\mu))$ and $F_{M^*}(\pi(i\mu))$; we write $\gamma(e)$ for that edge. Since the edges on the same source have different labels we have that $\gamma(e)$ is unique. In this way, we extend $\partial\gamma$ to a map $\gamma$ on the edges. Then $(\partial\gamma, \gamma)$ gives the required edge-labeled graph isomorphism. ∎

**Remark 5.7.** Proposition 5.6 is straightforward when $\Lambda^*$ and $M^*$ are languages of subshifts. In this case $(X_{\Lambda^*})^* = \Lambda^*$ and the bijection gives a 1-block code between the subshifts. However here we provide the same result even when $(X_{\Lambda^*})^* \subset \Lambda^*$ and with no reference to possible subshifts that the languages induce.

### 5.2. **Unlabeled graph isomorphisms.**

By an unlabeled graph isomorphism between two edge-labeled graphs we will mean a simple isomorphism of the ambient graphs with the labels omitted.

**Theorem 5.8.** *Let $\Lambda^*$ and $M^*$ be sofic languages. Then their unlabeled follower set graphs are isomorphic if and only if their quantized dynamics are locally piecewise conjugate.*

**Proof.** By assumption the spaces $\Omega_{\Lambda^*}$ and $\Omega_{M^*}$ are (discrete) finite spaces. First suppose that the unlabeled follower set graphs are isomorphic. Then we obtain an induced bijection $\gamma_s$ on the vertices, and thus a homeomorphism $\gamma_s \colon \Omega_{\Lambda^*} \to \Omega_{M^*}$. Let a point $\omega \in Q_{[m]}$. Then the number of the edges emitted by $\omega$ coincides with the number of edges emitted by $\gamma_s(\omega)$. Due to the follower set graph construction, this implies that $\gamma_s(\omega) \in P_{[n]}$, with $|\operatorname{supp}[n]|$ coinciding with the number of emitted edges from $\gamma_s(\omega)$, thus with $|\operatorname{supp}[m]|$. Now the graph isomorphism implies a bijection, say $\pi$, between edges. For convenience let $\operatorname{supp}[m] = \{i_1, \dots, i_r\}$ and $\operatorname{supp}[n] = \{j_1, \dots, j_r\}$, such that $\pi(i_l) = j_l$ for all $l = 1, \dots r$. Then the terminal vertex of $i_l$ is mapped to the terminal of $\pi(i_l)$. Consequently we obtain

$$\gamma_s \phi_{i_l}(\omega) = \psi_{\pi(i_l)} \gamma_s(\omega) \text{ for all } l = 1, \dots, r.$$

Taking $\mathcal{U} = \{\omega\}$ gives the required local piecewise conjugacy.

Conversely suppose that $(\Omega_{\Lambda^*}, \varphi)$ and $(\Omega_{M^*}, \psi)$ are locally piecewise conjugate. Then the ambient spaces are homeomorphic, i.e. there is a bijection on the vertex sets. Moreover for every point $\omega \in Q_{[m]}$ we have $\gamma_s(\omega) \in P_{[n]}$ with $|\operatorname{supp}[m]| = |\operatorname{supp}[n]|$, such that

$$\gamma_s \varphi_i(\omega) = \psi_{\pi(i)} \gamma_s(\omega) \text{ for all } i \in \operatorname{supp}[m].$$

Recall that $\varphi_i$ is defined on $\omega$ if and only if $i \in \operatorname{supp}[m]$. Hence local piecewise conjugacy gives that the number of edges that $\omega$ emits is $|\operatorname{supp}[m]|$, and thus it equals the number of edges that $\gamma_s(\omega)$ emits, which is $|\operatorname{supp}[n]|$. The above equation then shows that for the edge labeled $i$ there is a unique edge labeled $\pi(i)$, such that the end point of the $i$-edge is mapped to the endpoint of the $\pi(i)$-edge. As edges are preserved under $\gamma_s$ we get the required graph isomorphism. ∎

**Remark 5.9.** There is a conceptual difference between labeled and unlabeled graph isomorphism. In the first case the isomorphism on the edges is given by the bijection on the labels, and is the same on all vertices. However in the second case the bijection changes each time we pass to another vertex.

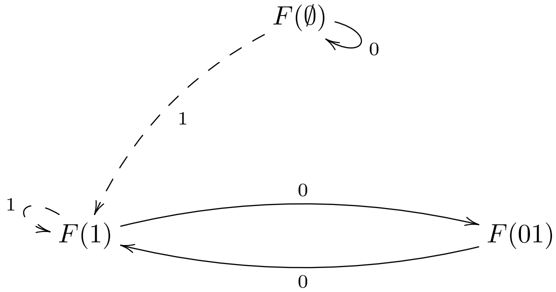
Removing the labels from a graph representation $\mathfrak{G} = (G, \mathfrak{L})$ of a subshift $\Lambda$ does not preserve type. In the case of sofic subshifts this procedure amounts to producing finite covers and hints that unlabeled graph isomorphisms of the follower set graphs *should not*[4] preserve SFT's. The following example clarifies this point.

[4] Notice the subtle point here: If $\Lambda$ is a sofic subshift then removing all labels from its follower set graph $\mathfrak{G} = (G, \mathfrak{L})$ produces an edge-shift $X_G$, and $\Lambda$ is a factor of $X_G$. However it is not ensured that $G$ is the follower set graph of $X_G$.

**Example 5.10.** Let $\Lambda^*$ be the language of the even shift on $\{0,1\}$, i.e. the forbidden words are of the form $10^{2n+1}1$ for all $n \geq 0$. Then the vertices of the follower set graph are of the form

$$F(\mu) = \begin{cases} F(\emptyset) & \text{if } \mu \text{ contains no 1's,} \\ F(1) & \text{if } \mu \text{ begins with } 0^{2k}1 \text{ for some } k \geq 0, \\ F(01) & \text{if } \mu \text{ begins with } 0^{2k+1}1 \text{ for some } k \geq 0. \end{cases}$$

for $\Lambda^*$. Therefore the follower set graph takes up the form



For more details see [**31**, Example 3.2.7, Figure 3.2.2].

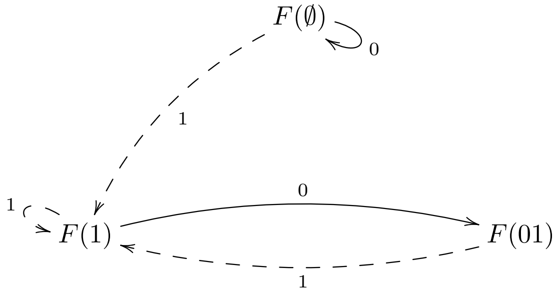
On the other hand let $M^*$ on $\{0,1\}$ be defined by the forbidden word 001. Then the truthTable of the Follower Set Graph Algorithm is given by:

| 11 | 01 | 10 | 00 | 1 | 0 | $\emptyset$ | |
|---|---|---|---|---|---|---|---|
| | | | | | | | $\emptyset$ |
| | | | | | | | 0 |
| | | | ■ | | | | 1 |
| | | | | | | | 00 |
| | | | ■ | | | | 10 |
| | | ■ | ■ | | ■ | | 01 |
| | | | ■ | | | | 11 |

(with FALSE indicated by a black box). Therefore we have the vertices

$$F(\emptyset) = F(0) = F(00) = F(000),$$
$$F(1) = F(10) = F(11) = F(100) = F(101) = F(110) = F(111),$$
$$F(01) = F(010) = F(011).$$

Then the follower set graph for $M^*$ is



It is immediate that the unlabeled graphs for $\Lambda^*$ and $M^*$ are isomorphic. However $\Lambda^*$ is not a language of finite type.

Notice also how the functions $f_n \colon \mathcal{B}_n(\Lambda^*) \to \mathcal{B}_n(M^*)$ of Remark 4.2 depend on $n$ in this example. For example, we have that

$$f_4(0010) = 1010 \neq 0010 = f_1(0)f_3(010).$$

Recall that the image of $w$ under $f_n$ is taken through the unlabeled graph isomorphism, when $w$ corresponds to *the* path $w$ beginning at $\omega = F(\emptyset)$.

**Remark 5.11.** Recall that a point $x$ in a two-sided subshift $\Lambda$ has *period* $n$ if there exists an $n \in \mathbb{N}$ such that $\sigma^n(x) = x$. The number of points with period $n$ is denoted by $p_n(\Lambda)$. The *zeta function* of $\Lambda$ is given by

$$\zeta_\Lambda(t) := \exp\Big( \sum_{n=1}^{\infty} \frac{p_n(\Lambda)}{n} t^n \Big).$$

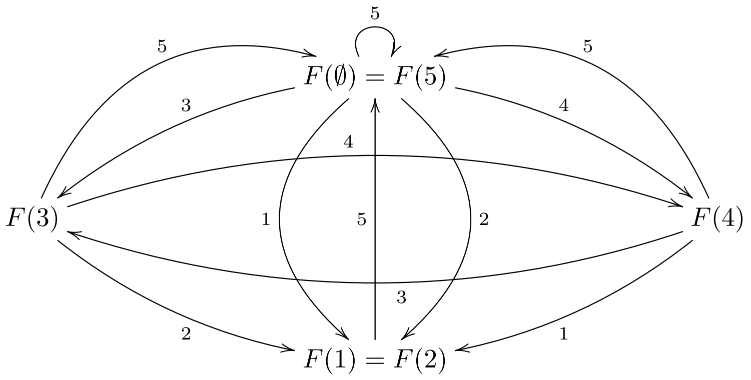
The zeta function is not preserved by local piecewise conjugacy. In Example 5.10 we show that the even shift is locally piecewise conjugate to a subshift of finite type. By [**31**, Theorem 6.4.6] the zeta function of any subshift of finite type is the reciprocal of a polynomial, whereas [**31**, Example 6.4.5] implies that the zeta function of the even shift is $\zeta(t) = (1+t)(1-t-t^2)^{-1}$. We would like to thank Ian Putnam for this remark.

5.3. **Graph isomorphism for type 1 languages.** Let us examine further the case of type 1 languages. We begin with an example of two languages with no isomorphic labeled follower set graphs.

**Example 5.12.** Consider the language $\Lambda^*$ on 5 symbols determined by the forbidden words

$$\{11, 21, 31, 41, 12, 22, 32, 42, 13, 33, 24, 44\}.$$

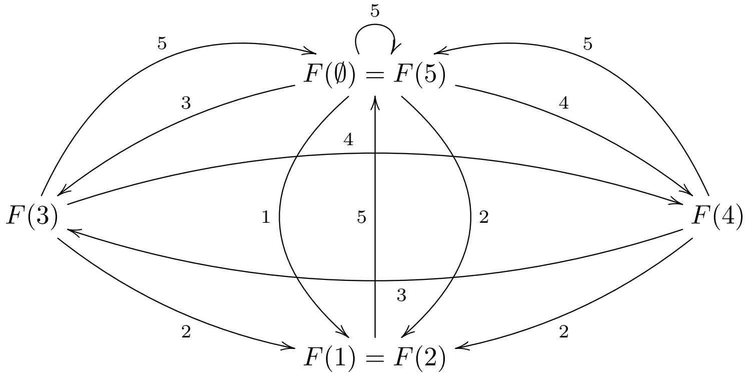
Then the resulting follower set graph is



for $\Lambda^*$. Consider also the language $M^*$ on 5 symbols determined by the forbidden words

$$\{11, 21, 31, 41, 12, 22, 32, 42, 13, 33, 14, 44\}.$$

The only difference with the forbidden words of $\Lambda^*$ is to consider 14 in place of 24. Similarly we get the follower set graph



for $M^*$. The unlabeled graphs are isomorphic. The only difference in the labeled graphs is the lower right arrow which carries different labels.

As the follower set graphs are irreducible it can be seen that the languages $\Lambda^*$ and $M^*$ are also the languages of two-sided irreducible subshifts. In fact these are the augmented versions of [**25**, Example 9.8].

A key feature in the example above is that there are two symbols that have the same follower sets. It appears that this is the only obstruction.

**Theorem 5.13.** *Let $\Lambda^*$ and $M^*$ be languages of type* 1*. Suppose that $\Lambda^*$ is on $d$ symbols and that $F_{\Lambda^*}(i) \neq F_{\Lambda^*}(j)$ for $i \neq j$, with $i, j = 1, \dots d$. The following are equivalent:*

(i) *The follower set graphs are isomorphic;*
(ii) *The unlabeled follower set graphs are isomorphic.*

**Proof.** Of course item (i) implies item (ii). For the converse recall that unlabeled graph isomorphism imposes that $M^*$ is on $d$ symbols as well. Furthermore the vertex sets must have the same size.

We have two cases. If $F_{\Lambda^*}(i) \neq F_{\Lambda^*}(\emptyset)$ for all $i$ then the same must hold for the graph of $M^*$, as $d+1$ is the maximum size of the vertex sets. In this case both $F_{\Lambda^*}(\emptyset)$ and $F_{M^*}(\emptyset)$ are the unique sources for the graph, hence related by the graph isomorphism. If there is an $i$ such that $F_{\Lambda^*}(i) = F_{\Lambda^*}(\emptyset)$ then the number of the $F_{M^*}(j)$ is $d$, and thus there exists a unique $j$ such that $F_{M^*}(j) = F_{M^*}(\emptyset)$.

In any case we get that the graph isomorphism induces a bijection between $\{F_{\Lambda^*}(i) \mid i = 1, \dots, d\}$ and $\{F_{M^*}(i) \mid i = 1, \dots, d\}$. Without loss of generality we may relabel for $M^*$ so that this bijection sends $F_{\Lambda^*}(i)$ to $F_{M^*}(i)$. As we remarked in Example 5.1 the labels on the edges are pre-determined by their range. Notice that by hypothesis every vertex receives at most one edge. Therefore the unlabeled graph isomorphism respects the label of the edges, and the proof is complete. ∎

**Remark 5.14.** For type 1 subshifts there is a strong connection between the followerTable of the Follower Set Graph Algorithm and the representation of the subshift as an edge shift. Let us recall how this follows from [**31**, Theorem 2.3.2]. To allow comparisons we denote the graph of the edge shift by $G_e(\Lambda)$.

Let $\Lambda$ be a two-sided subshift of type 1. The vertices of $G_e(\Lambda)$ are the symbols of $\Lambda$. We write an edge between the vertex $i$ and $j$ if (and only if) $ji \in \Lambda^*$, and label the resulting edge by $j$. It is evident that this graph is given by a 0-1 adjacency matrix.

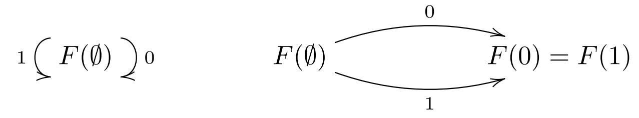
If $A$ is the followerTable of the Follower Set Graph Algorithm where we replace the allowable words by 1 and the forbidden words by 0, then the adjacency matrix of $G_e(\Lambda)$ is taken by deleting the row and the column that corresponds to $\emptyset$ in $A$ (which all have entries equal to 1).

Working under the condition that $F$ is one-to-one on the symbol set (of Theorem 5.13) we distinguish two cases:

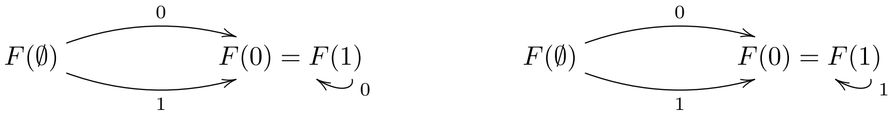
- Case 1. If $F(\emptyset) = F(i)$ for some symbol $i$ then $G_e(\Lambda)$ coincides with the graph $G$ of the follower set graph $\mathfrak{G} = (G, \mathfrak{L})$ of $\Lambda$.
- Case 2. If $F(\emptyset) \neq F(i)$ for all symbols $i$ then $G_e(\Lambda)$ coincides with the subgraph $G$ of the follower set graph $\mathfrak{G} = (G, \mathfrak{L})$ of $\Lambda$, once we erase the vertex $F(\emptyset)$ and the emitting edges.

**Remark 5.15.** Theorem 5.13 applies to edge shifts with invertible adjacency matrices. Indeed let $A_e$ be the adjacency matrix of $G_e(\Lambda)$. If there are $i \neq j$ with $F(i) = F(j)$ then we have that two rows of the matrix from the Follower Set Graph Algorithm coincide. Thus the same holds for $A_e$ and thus $\det A_e = 0$.

**Remark 5.16.** Theorem 5.13 holds in the particular case of languages of type 1 on two symbols $\{0, 1\}$ without the assumption on the follower set function. Indeed the cases where $F(0) = F(1)$ produce the following graphs



for the sets of forbidden words $\emptyset$ and $\{00, 10, 01, 11\}$, and



for the sets of forbidden words $\{00, 01\}$ and $\{11, 10\}$, respectively. It follows that also in these cases the follower set graphs are unique up to a permutation of symbols.

However one direction of Theorem 5.13 does not hold in general even for languages of finite type. We highlight this in the following two examples.

**Example 5.17.** Let the language $\Lambda^*$ on two symbols $\{0, 1\}$ be defined by the forbidden words

$$000, 010, 001, 101, 011.$$

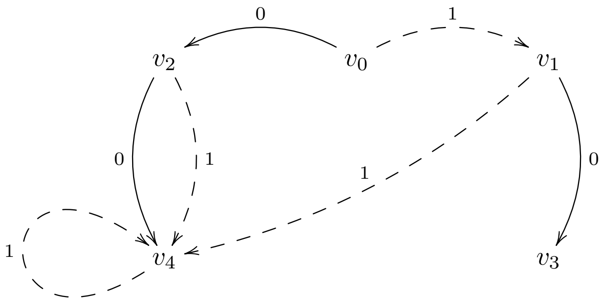
Then the corresponding truthTable of the Follower Set Graph Algorithm is:

| 11 | 01 | 10 | 00 | 1 | 0 | $\emptyset$ | |
|---|---|---|---|---|---|---|---|
| | | | | | | | $\emptyset$ |
| | ■ | | ■ | | | | 0 |
| | ■ | ■ | ■ | | | | 1 |
| | ■ | ■ | ■ | | ■ | | 00 |
| | ■ | ■ | ■ | | ■ | | 10 |
| ■ | ■ | ■ | ■ | ■ | ■ | | 01 |
| | ■ | ■ | ■ | | ■ | | 11 |

and gives the vertices

$$v_0 = F(\emptyset), v_1 = F(1), v_2 = F(0), v_3 = F(01),$$
$$v_4 = F(00) = F(10) = F(11) = F(100) = F(110) = F(111).$$

Then the follower set graph of $\Lambda^*$ is



On the other hand let the language $M^*$ on two symbols $\{0, 1\}$ be defined by the forbidden words

$$000, 010, 001, 100, 011.$$

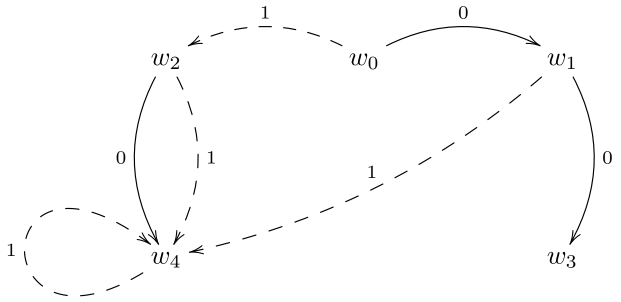
Then the corresponding truthTable of the Follower Set Graph Algorithm is:

| 11 | 01 | 10 | 00 | 1 | 0 | $\emptyset$ | |
|---|---|---|---|---|---|---|---|
| | | | | | | | $\emptyset$ |
| | ■ | ■ | ■ | | | | 0 |
| | ■ | | ■ | | | | 1 |
| ■ | ■ | ■ | ■ | ■ | ■ | | 00 |
| | ■ | ■ | ■ | | ■ | | 10 |
| | ■ | ■ | ■ | | ■ | | 01 |
| | ■ | ■ | ■ | | ■ | | 11 |

and gives the vertices

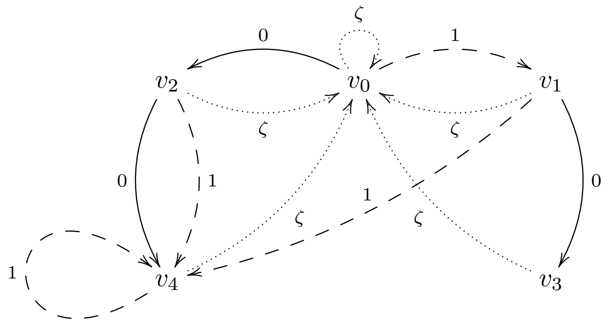
$$w_0 = F(\emptyset), w_1 = F(0), w_2 = F(1), w_3 = F(00),$$
$$w_4 = F(10) = F(01) = F(11) = F(101) = F(111) = F(110).$$

Then the follower set graph of $M^*$ is

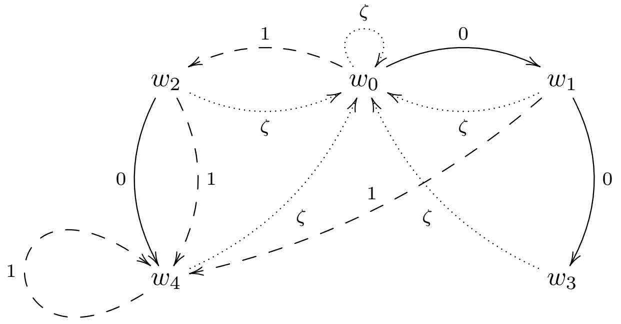


It is clear that there is only one unlabeled graph isomorphism; the one sending $v_i$ to $w_i$. If it lifted to a labeled graph isomorphism then the 0 label would match to the 0 label, as it appears from $v_4$ and $w_4$. However this does not comply with the labels on $v_0$ and $w_0$.

The two-sided subshifts coming from the languages $\Lambda^*$ and $M^*$ are formed on a single point. For creating a more interesting counterexample in the category of two-sided subshifts we may use the augmentations defined in $\{0, 1, \zeta\}$. In this case we obtain



as the follower set graph of $\widetilde{\Lambda}^*$ and



as the follower set graph of $\widetilde{M}^*$. Again there is not a labeled graph isomorphism between those.

**Example 5.18.** Let the language $\Lambda^*$ on two symbols $\{0, 1\}$ be defined by the forbidden words

$$000, 100, 010, 101, 011, 111.$$

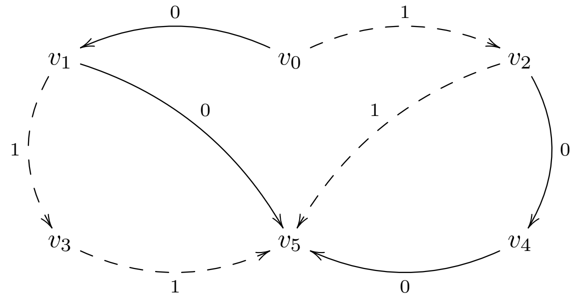
Then the corresponding truthTable of the Follower Set Graph Algorithm is:

| 11 | 01 | 10 | 00 | 1 | 0 | ∅ | |
|---|---|---|---|---|---|---|---|
| | | | | | | | ∅ |
| | ■ | ■ | ■ | | | | 0 |
| ■ | ■ | ■ | | | | | 1 |
| ■ | ■ | ■ | ■ | ■ | ■ | | 00 |
| ■ | ■ | ■ | ■ | | ■ | | 10 |
| ■ | ■ | ■ | ■ | ■ | | | 01 |
| ■ | ■ | ■ | ■ | ■ | ■ | | 11 |

and gives the vertices

$$v_0 = F(\emptyset), v_1 = F(0), v_2 = F(1), v_3 = F(10), v_4 = F(01)$$
$$v_5 = F(00) = F(11) = F(110) = F(001).$$

Then the follower set graph of $\Lambda^*$ is



On the other hand let the language $M^*$ on two symbols $\{0, 1\}$ be defined by the forbidden words

$$000, 110, 010, 101, 001, 111.$$

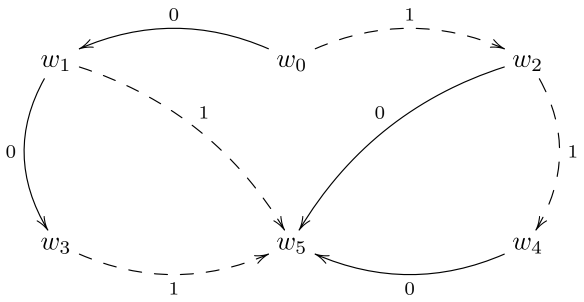
Then the corresponding truthTable of the Follower Set Graph Algorithm is:

| 11 | 01 | 10 | 00 | 1 | 0 | ∅ | |
|---|---|---|---|---|---|---|---|
| | | | | | | | ∅ |
| ■ | ■ | | ■ | | | | 0 |
| ■ | | ■ | ■ | | | | 1 |
| ■ | ■ | ■ | ■ | | ■ | | 00 |
| ■ | ■ | ■ | ■ | ■ | ■ | | 10 |
| ■ | ■ | ■ | ■ | ■ | ■ | | 01 |
| ■ | ■ | ■ | ■ | ■ | | | 11 |

and gives the vertices

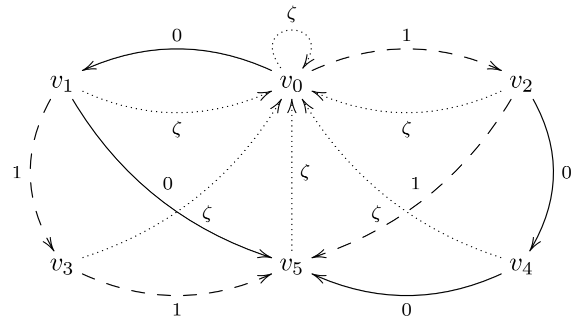
$$w_0 = F(\emptyset), w_1 = F(0), w_2 = F(1), w_3 = F(00), w_4 = F(11)$$
$$w_5 = F(10) = F(01) = F(100) = F(011).$$

Then the follower set graph of $M^*$ is

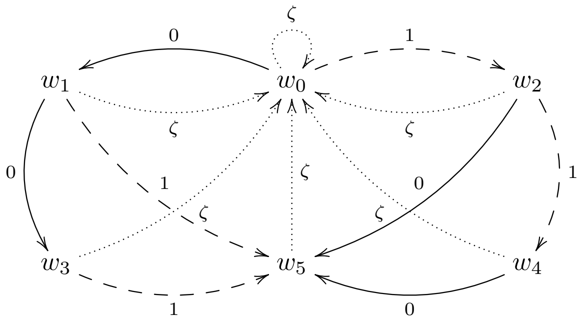


We see that that there are two unlabeled graph isomorphisms. The first one sends $v_i$ to $w_i$ and the second one is the composition with the reflection along the vertical line that passes through $w_0$ and $w_5$. Both of them do not lift to a labeled graph isomorphism as the path 11 connecting $v_1$ with $v_5$ consists of two edges with the same label whereas its image 10 has two edges of different labels.

The languages $\Lambda^*$ and $M^*$ do not arise from subshifts, but once more we can use their augmentations to produce a counterexample in this class. We thus have



for $\widetilde{\Lambda}^*$, and



for $\widetilde{M}^*$. Again the follower set graphs are not labeled graph isomorphic.

5.4. **Irreducible two-sided sofic subshifts.** Let us examine further the case of irreducible two-sided sofic subshifts. Recall that a subshift is called *irreducible* if it has a presentation through a labeled graph $\mathfrak{G} = (G, \mathfrak{L})$ so that $G$ is irreducible. Among all presentations, Fischer [**17, 18**] has shown that there exists a minimal resolving one, that is unique up to label graph isomorphism; see also [**31**, Theorem 3.3.18]. Minimality is taken with respect to the number of vertices of the ambient graph. This presentation is also known as the *Fischer cover* of the subshift. Fischer covers are follower-separated; for example see [**31**, Corollary 3.319]. Uniqueness of the Fischer cover fails for reducible sofic subshifts; Jonoska [**22**] provides such a counterexample for subshifts of finite type.

The Fischer cover can be induced by the Krieger cover or by the follower set graph. A way to obtain it from the follower set graph is as follows. Recall that a word $\mu$ is called *intrinsically synchronizing* if:

$$\text{whenever } \nu\mu \in \Lambda^* \text{ and } \mu\nu' \in \Lambda^* \text{ then } \nu\mu\nu' \in \Lambda^*.$$

Then the minimal resolving presentation is the labeled subgraph of the follower set graph formed by using just the follower sets of the intrinsically synchronizing words[5] (see [**31**, Exercise 3.3.4]).

We will see how the graph isomorphism we get from Theorem 5.8 induces an unlabeled graph isomorphism of the Fischer covers. Moreover that it respects the vertices labeled by follower sets of intrinsically synchronizing words. For the latter we will require some terminology and results from [**31**]. Recall that we follow the left version of their notation.

Given a labeled graph $\mathfrak{G} = (G, \mathfrak{L})$ we say that a path $\mu$ is *synchronizing* for $\mathfrak{G}$ if:

$$\text{all paths labeled } \mu \text{ terminate at the same vertex.}$$

Recall that we read paths from right to left. Suppose in addition that $\mathfrak{G}$ is resolving and follower separated. In this case every path $w$ can be extended on the left to a synchronizing path $\mu = uw$ for $\mathfrak{G}$ by [**31**, Proposition 3.3.16]. Under the same assumption if $\mu$ is synchronizing for $\mathfrak{G}$ then every path $u\mu$ is synchronizing for $\mathfrak{G}$ by [**31**, Lemma 3.3.15]. The connection between synchronizing paths and intrinsically synchronizing words is given in [**31**, Exercise 3.3.3]. That is, if $\mathfrak{G}$ is the minimal resolving presentation of a two-sided irreducible sofic shift $\Lambda$ then a path $w$ is synchronizing for $\mathfrak{G}$ if and only if the word $w$ is intrinsically synchronizing for $\Lambda$.

Now we have set the context for proving the next corollary.

**Corollary 5.19.** *Let $\Lambda$ and $M$ be two-sided irreducible sofic subshifts. If $\Lambda$ and $M$ are locally piecewise conjugate then there is an unlabeled graph isomorphism between their Fischer covers.*

[5] It is worth mentioning that a subshift is of finite type if and only if all sufficiently long words are intrinsically synchronizing [**31**, Exercise 3.3.5]. This property emphasizes the value of Jonoska's [**22**] counterexample.

*Furthermore, the unlabeled graph isomorphism induces a bijection between the collections*

$$\{F_\Lambda(\mu) \mid \ \mu \text{ is an intrinsically synchronizing word for } \Lambda\}$$

*and*

$$\{F_M(\nu) \mid \ \nu \text{ is an intrinsically synchronizing word for } M\}.$$

**Proof.** Let $\mathfrak{G}_\Lambda = (G_\Lambda, \mathcal{L}_1)$ and $\mathfrak{G}_M = (G_M, \mathcal{L}_2)$ be the follower set graphs of $\Lambda$ and $M$, respectively. Let $\mathfrak{H}_\Lambda$ be the labeled graph that remains from $\mathfrak{G}_\Lambda$ by using only the intrinsically synchronizing words, i.e. the Fischer cover of $\Lambda$. The graph isomorphism of Theorem 5.8 then gives an isomorphism of the ambient graph $H_\Lambda$ of $\mathfrak{H}_\Lambda$ onto a subgraph $H_M$ of $G_M$. Let $\mathfrak{H}_M$ be the labeled subgraph induced by $H_M$ inside $\mathfrak{G}_M$. Notice that both $\mathfrak{H}_\Lambda$ and $\mathfrak{H}_M$ are resolving as subgraphs of the resolving follower set graphs and thus

$$h(X(\mathfrak{H}_\Lambda)) = h(X(H_\Lambda)) \quad \text{and} \quad h(X(\mathfrak{H}_M)) = h(X(H_M))$$

by [**31**, Proposition 4.13].

First we claim that $\mathfrak{H}_M$ gives a presentation of $M$. If $X(\mathfrak{H}_M) \neq M$ then $X(\mathfrak{H}_M)$ is a proper subshift of $M$ and therefore [**31**, Corollary 4.4.9] yields

$$h(X(\mathfrak{H}_M)) < h(M).$$

Since graph isomorphism of $H_\Lambda$ with $H_M$ respects entropy we get that

$$h(\Lambda) = h(X(\mathfrak{H}_\Lambda)) = h(X(H_\Lambda)) = h(X(H_M)) = h(X(\mathfrak{H}_M)) < h(M).$$

However this contradicts $h(M) = h(\Lambda)$ of Proposition 4.3. Hence $\mathfrak{H}_M$ is a presentation of $M$. Notice here that the number of vertices of $\mathfrak{H}_M$ coincides with the minimal number of vertices required to describe $\Lambda$, i.e.

$$|(\mathfrak{H}_M)^{(0)}| = |(H_M)^{(0)}| = |(H_\Lambda)^{(0)}| = |(\mathfrak{H}_\Lambda)^{(0)}|.$$

Secondly we claim that $\mathfrak{H}_M$ is minimal for $M$ with respect to the number of vertices. Otherwise we could find a subgraph $\mathfrak{H}'_M$ of $\mathfrak{G}_M$ on less vertices than that of $\mathfrak{H}_M$. Then the unlabeled graph isomorphism would carry over, as above, to a presentation $\mathfrak{H}'_\Lambda$ of $\Lambda$, giving

$$|(\mathfrak{H}'_\Lambda)^{(0)}| = |(\mathfrak{H}'_M)^{(0)}| < |(\mathfrak{H}_M)^{(0)}| = |(\mathfrak{H}_\Lambda)^{(0)}|.$$

However this contradicts minimality of $\mathfrak{H}_\Lambda$ for $\Lambda$, and thus $\mathfrak{H}_M$ is a minimal presentation of $M$.

So far we have proved that $\mathfrak{H}_M$ is a minimal resolving presentation of $M$. Initially $\mathfrak{H}_M$ is isomorphic to the subgraph of $\mathfrak{G}_M$ obtained by using the vertices labeled by follower sets of intrinsically synchronizing words. We will show now that these two sub-graphs of $\mathfrak{G}_M$ are actually equal. Due to minimality it suffices to show that the vertices of $\mathfrak{H}_M$ correspond to follower sets of intrinsically synchronizing words.

We will use that $\mathfrak{H}_M$ is follower-separated and irreducible as a minimal presentation of $M$ [**31**, Corollary 3.3.19]. Let $F(w)$ be a vertex of $\mathfrak{H}_M$. We will show that

$$\exists \text{ an intrinsically synchronizing word } \nu \in M^* \text{ such that } F(w) = F(\nu).$$

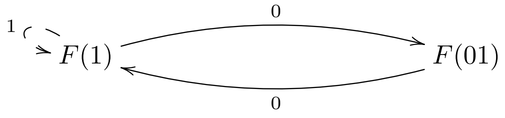
Consider a labeled path $u_1$ in $\mathfrak{H}_M$ starting at a vertex $J$ and extend it to a synchronizing path $u_2u_1$ in $\mathfrak{H}_M$. That is, all paths labeled $u_2u_1$ end at the same vertex, say $J'$ in $\mathfrak{H}_M$. By irreducibility of $\mathfrak{H}_M$ there is a path $u_3$ connecting $J'$ with $F(w)$ so that the path $u_3u_2u_1$ is allowable in $\mathfrak{H}_M$. Moreover this is an extension of a synchronizing path and thus it is a synchronizing path in $\mathfrak{H}_M$, i.e. all paths in $\mathfrak{H}_M$ representing $u_3u_2u_1$ end at $F(w)$. Minimality of $\mathfrak{H}_M$ implies that $u_3u_2u_1$ is an intrinsically synchronizing word for $M$. As $\mathfrak{H}_M$ represents $M$, the follower set of $u_3u_2u_1$ in $M$ coincides with the paths in $\mathfrak{H}_M$ starting at the vertex $F(w)$. However the collection of these paths is exactly the set $F(w)$ due to the follower set graph construction. Hence we conclude that $F(w) = F(u_3u_2u_1)$ and the proof is complete. ∎

**Remark 5.20.** Referring to the proof of Corollary 5.19, we do not claim a direct bijection between intrinsically synchronizing words. It is unclear whether the function of Proposition 4.3 respects this property. The main obstacle is that the word

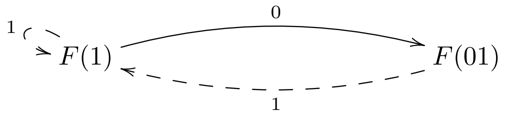
$$f_n^{-1}(f_{|\nu|}(\nu)\mu f_{|\nu'|}(\nu')) \quad \text{for } n = |\nu| + |\mu| + |\nu'|$$

gives a word $\nu''\mu'\nu'$ instead of $\nu\mu'\nu'$ (as indicated in Example 5.10). A similar obstacle does not allow checking directly whether the mixing property is also preserved under local piecewise conjugacy.

**Example 5.21.** We require both $\Lambda$ and $M$ be irreducible in Corollary 5.19. This is because local piecewise conjugacy does not preserve irreducibility. For an example recall the even shift and the subshift of finite type constructed in Example 5.10. The even shift $\Lambda$ is irreducible and the subgraph



gives its Fischer cover. However the subshift $M$ on $\{001\}$ is not irreducible. Any subgraph on less vertices produces a proper subshift of $M$. In particular the unlabeled graph isomorphism from the Fischer cover of $\Lambda$ to $M$ produces the irreducible subgraph



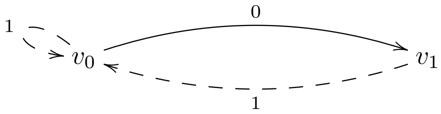
which does not represent $0^\infty \in M$.

**Example 5.22.** The converse of Corollary 5.19 does not hold. That is, irreducible subshifts that are not locally piecewise conjugate may have Fischer covers that admit an unlabeled graph isomorphism.

For a counterexample let $M$ be defined by the forbidden word $\{00\}$. Then the truthTable of $M$ is

| 1 | 0 | ∅ |  |
|---|---|---|---|
|  |  |  | ∅ |
|  | ■ |  | 0 |
|  |  |  | 1 |

and thus its follower set graph is given by



where $v_0 = F(\emptyset) = F(1)$ and $v_1 = F(0)$. Consequently $M$ is irreducible (having one irreducible presentation) and its Fischer cover coincides with its follower set graph. Comparing with the even shift $\Lambda$ we see that the Fischer covers of $\Lambda$ and $M$ admit an unlabeled graph isomorphism, but this does not hold for their follower set graphs.

## References

[1] W. Arveson, *Operator algebras and measure preserving automorphisms*, Acta Math. **118**, (1967), 95–109.

[2] W. B. Arveson, *Subalgebras of C*-algebras,* Acta Math. **123** (1969) 141–224.

[3] W. Arveson and K. Josephson, *Operator algebras and measure preserving automorphisms II*, J. Funct. Anal. **4** (1969), 100–134.

[4] T. M. Carlsen, *C*-algebras associated with subshifts*, Internat. J. Math. **19:1** (2008), 47–70.

[5] T. M. Carlsen and K. Matsumoto, *Some remarks on the C*-algebras associated with subshifts*, Math. Scand. **95:1** (2004), 145–160.

[6] G. Cornelissen, *Curves, dynamical systems and weighted point counting*, Proc. Nat. Acad. Sci. U.S.A. **110:4** (2013), 9669–9673.

[7] G. Cornelissen and M. Marcolli, *Quantum statistical mechanics, L-series and anabelian geometry*, preprint (arXiv:math.NT/1009.0736).

[8] G. Cornelissen and M. Marcolli, *Graph reconstruction and quantum statistical mechanics*, J. Geom. Phys. **72** (2013), 110–117.

[9] J. Cuntz and W. Krieger, *A class of C*-algebras and topological Markov chains*, Invent. Math. **56:3** (1980), 251–268.

[10] K. R. Davidson and E. T.A. Kakariadis, *Conjugate Dynamical Systems on C*-algebras*, IMRN **2014:5** (2014), 1289–1311.

[11] K. R. Davidson and E. G. Katsoulis, *Isomorphisms between topological conjugacy algebras*, J. reine angew. Math. **621** (2008), 29–51.

[12] K. R. Davidson and E. G. Katsoulis, *Operator algebras for multivariable dynamics*, Mem. Amer. Math. Soc. **209: 982** (2011).

[13] K. R. Davidson and J. Roydor, *Isomorphisms of tensor algebras of topological graphs*, Indiana Univ. Math. J. **60:4** (2011), 1249–1266.

[14] K. R. Davidson, C. Ramsey and O. M. Shalit, *The isomorphism problem for some universal operator algebras*, Adv. Math. **228:1** (2011), 167–218.

[15] K. R. Davidson, C. Ramsey and O. M. Shalit, *Operator algebras for analytic varieties*, Trans. Amer. Math. Soc. **367:2** (2015), 1121–1150.

[16] A. Dor-On, *Isomorphisms of tensor algebras arising from weighted partial systems*, to appear in Transactions AMS (preprint arXiv: 1507.08400).

[17] R. Fischer, *Sofic systems and graphs*, Monatsh. Math. **80:3** (1975), 179–186.

[18] R. Fischer, *Graphs and symbolic dynamics*, Colloq. Math. Soc. Janos Bolyai: Topics in Information Theory **16** (1975), 229–243.

[19] D. Hadwin and T. Hoover, *Operator algebras and the conjugacy of transformations*, J. Funct. Anal. **77** (1988), 112–122.

[20] M. Hamana, *Injective envelopes of operator systems*, Publ. Res. Inst. Math. Sci. **15:3** (1979), 773–785.

[21] R. Johansen, *On flow equivalence of sofic shifts*, PhD Thesis, University of Copenhagen (2012).

[22] N. Jonoska, *Sofic shifts with synchronizing presentations*, Theor. Computer Sci. **158** (1995), 81–115.

[23] E. T.A. Kakariadis and E. G. Katsoulis, *Isomorphisms invariants for multivariable C*-dynamics*, J. Noncommut. Geom. 8:3 (2014), 771-787.

[24] E. T.A. Kakariadis and J. R. Peters, *Representations of C*-dynamical systems implemented by Cuntz families*, Münster J. Math. **6** (2013), 383–411.

[25] E. T.A. Kakariadis and O. M. Shalit, *On operator algebras associated with monomial ideals in noncommuting variables*, preprint: arXiv.org 1501.06495.

[26] E. G. Katsoulis and D. W. Kribs, *Isomorphisms of algebras associated with directed graphs*, Math. Ann. **330:4** (2004), 709–728.

[27] E. G. Katsoulis and D. W. Kribs, *Tensor algebras of C*-correspondences and their C*-envelopes*, J. Funct. Anal. **234:1** (2006), 226–233.

[28] E. G. Katsoulis and C. Ramsey, *Limit algebras and integer-valued cocycles*, to appear in J. London Math. Soc. (preprint arXiv:1601.03423).

[29] T. Katsura, *On C*-algebras associated with C*-correspondences*, J. Funct. Anal. **217:2** (2004), 366–401.

[30] W. Krieger, *On sofic systems I*, Israel J. Math. **48:4** (1984), 305–330.

[31] D. Lind and B. Marcus, *An Introduction to Symbolic Dynamics and Coding*, Cambridge University Press, Cambridge, 1995.

[32] K. Matsumoto, *On C*-algebras associated with subshifts*, Internat. J. Math. **8:3** (1997), 357–374.

[33] K. Matsumoto, *K-Theory for $C^*$-algebras associated with subshifts*, Math. Scand. **82:2** (1998), 237–255.

[34] K. Matsumoto, *Dimension groups for subshifts and simplicity of the associated C*-algebras*, J. Math. Soc. Japan **51:3** (1999), 679–698.

[35] K. Matsumoto, *C*-algebras associated with presentations of subshifts*, Doc. Math. **7** (2002), 1–30.

[36] P. S. Muhly and B. Solel, *Tensor algebras over C*-correspondences: representations, dilations and C*-envelopes*, J. Funct. Anal. **158:2** (1998), 389–457.

[37] J. Peters, *Semicrossed products of C*-algebras*, J. Funct. Anal. **59** (1984), 498–534.

[38] S. Power, *Classification of analytic crossed product algebras*, Bull. London Math. Soc. **24:4** (1992), 368–372.

[39] O. M. Shalit and B. Solel, *Subproduct Systems*, Doc. Math. **14** (2009), 801–868.

[40] B. Solel, *You can see the arrows in a quiver operator algebra*, J. Aust. Math. Soc. **77:1** (2004), 111–122.

[41] B. Weiss, *Subshifts of finite type and sofic systems*, Monatsh. Math. **77:3** (1973), 462–474.

School of Mathematics and Statistics, Newcastle University, Newcastle upon Tyne, NE1 7RU, UK

*E-mail address*: `c.barrett2@newcastle.ac.uk`

School of Mathematics and Statistics, Newcastle University, Newcastle upon Tyne, NE1 7RU, UK

*E-mail address*: `evgenios.kakariadis@ncl.ac.uk`