Proceedings of the Edinburgh Mathematical Society (2018) **61**, 215–249
doi:10.1017/S0013091517000189

VISUALIZING AUTOMORPHISMS OF GRAPH ALGEBRAS

JAMES EMIL AVERY[1]*, RUNE JOHANSEN[2], WOJCIECH SZYMAŃSKI[3]

[1]*Niels Bohr Institute, University of Copenhagen, Blegdamsvej 17, 2100 København Ø, Denmark* (avery@nbi.ku.dk)
[2]*Department of Mathematics, University of Copenhagen, Universitetsparken 5, 2100 København Ø, Denmark* (rune@cyx.dk)
[3]*Department of Mathematics and Computer Science, University of Southern Denmark, Campusvej 55, 5230 Odense M, Denmark* (szymanski@imada.sdu.dk)

(Received 16 August 2016)

Abstract Many fundamental properties of graph C^*-algebras may be determined directly from the structure of the underlying graph, and because of this, they have been celebrated as C^*-algebras that can be seen. This paper shows how permutative endomorphisms of graph C^*-algebras can be represented by labelled directed multigraphs that give visual representations of the endomorphisms and facilitate computations. This formalism provides a useful calculus for permutative automorphisms and allows efficient exhaustive construction of such automorphisms.

Keywords: graph algebra; permutative endomorphism; automorphism; textile system

2010 *Mathematics subject classification:* Primary 46L40
Secondary 37B10

1. Introduction

This paper introduces the permutation graph as an efficient representation of permutative automorphisms of graph C^*-algebras, Leavitt path algebras, and shift spaces associated with a directed multigraph. This formalism makes practical the systematic study of permutative automorphisms in two ways. First, there is an efficient calculus for permutation graphs which makes it possible to determine whether a given permutative endomorphism is an automorphism, to efficiently compute compositions and inverses, and to find the orders of automorphisms, even in many cases of infinite order. Second, the permutation graph formalism makes it possible to efficiently carry out exhaustive searches for such automorphisms either by hand or using a computer program; in concrete cases, the computations can be completed many millions of times faster than previously possible.

* Corresponding author.

© 2018 The Edinburgh Mathematical Society

https://doi.org/10.1017/S0013091517000189 Published online by Cambridge University Press

These two developments have made it possible to generate a database of such automorphisms and their properties for a systematic collection of small graphs. In this paper, the permutation graph formalism will be introduced and applied to a number of concrete questions, including exhaustive construction of permutative automorphisms, while the database will be described and published elsewhere.

In [**14**], Cuntz introduced the Weyl group of the simple purely infinite C^*-algebras $\mathcal{O}_n$. It arises as the quotient of the normalizer of a maximal abelian subgroup of the C^*-algebra automorphism group. An important subgroup of the Weyl group corresponds to those automorphisms that globally preserve the canonical uniformly hyperfinite subalgebra of $\mathcal{O}_n$. Cuntz raised the question of how one determines the structure of this subgroup, and this was answered in [**7**]. In [**6**], this program was taken one step further by expanding it to a much wider class of graph C^*-algebras. In this paper, it is shown how permutative endomorphisms can be represented by labelled directed multigraphs, and how properties of such a permutation graph can be used to determine whether the corresponding permutative endomorphism is an automorphism.

Permutative endomorphisms of graph C^*-algebras, and especially of the Cuntz algebras $\mathcal{O}_n$, have already received considerable attention and have been studied in several different contexts. In particular, localized endomorphisms (a class of endomorphisms that includes permutative ones) of the Cuntz algebras were investigated in the framework of the Jones index theory in [**9**–**11**, **18**, **20**, **26**]. Similar investigations of localized endomorphisms of the Cuntz–Krieger algebras were carried out in [**21**]. Voiculescu's entropy and related properties of permutative endomorphisms of the Cuntz algebras were studied in [**30**, **31**]. Such endomorphisms have a role in the approach to wavelet theory via representations of the Cuntz algebras taken in [**5**]. An intriguing connection between permutative automorphisms of the Cuntz algebra $\mathcal{O}_n$ and automorphisms of the full two-sided n-shift was found in [**7**]. An interesting combinatorial approach to permutative endomorphisms of $\mathcal{O}_n$ was presented in [**22**].

In § 2, background information, definitions, and terminology are given, followed by a definition of permutation graphs and an examination of their properties in § 3. Here, a new proof is given of the result from [**6**], yielding automorphism conditions for permutative endomorphisms that are formulated directly in terms of their permutation graphs. A connection is shown between permutation graphs and Nasu's textile systems, allowing the computational methods of textile systems to be used in the analysis of permutative endomorphisms. In § 4, an algorithm is presented that serves to exhaustively construct all permutation graphs corresponding to permutative automorphisms. In § 5, inner equivalence of permutative automorphisms is linked to the behaviour of the corresponding permutation graphs as dynamical systems. An ordering of the vertices and edges of a permutation graph is introduced and used to construct representations of classes of automorphisms that are inner equivalent through permutative unitaries. These representations are textile systems, allowing the machinery developed in [**28**] to be applied to computations involving such equivalence classes. In § 6, an algorithm is presented that allows efficient exhaustive construction of inner equivalence classes of permutative automorphisms. The methods of §§ 4–6 have been implemented in a set of computer programs, which are applied to a particular graph C^*-algebra in § 7 and to certain Cuntz algebras in § 8. The connection to the theory of textile systems, which this paper introduces, paves the way for an efficient search for permutative automorphisms using computer programs.

https://doi.org/10.1017/S0013091517000189 Published online by Cambridge University Press

First, the algorithms presented in §§ 4 and 6 drastically reduce the number of cases one needs to examine in an exhaustive search for (equivalence classes of) permutative automorphisms. Second, the conditions introduced in § 3 – guaranteeing that a permutative endomorphism is an automorphism – can be efficiently tested using an algorithm that will be presented in a forthcoming paper [**4**]. Finally, the first and second authors have developed a number of practical algorithms that allow the efficient investigation of the order of a given permutative automorphism [**3**]. A forthcoming paper will apply these techniques to an investigation of concrete permutative automorphisms such as the ones identified in §§ 7 and 8, and another forthcoming paper will detail the computer programs we have developed based on these methods. These programs can be downloaded from the web page

`http://math.kvante.org/permutative-endomorphisms/`

Although we study endomorphisms of graph C^*-algebras, which are analytic in nature, our approach relies only on discrete and combinatorial properties, and only small modifications are needed to make the results of this paper applicable to purely algebraic objects. In particular, Proposition 3.8 and Theorem 3.13 also apply to permutative endomorphisms of Leavitt path algebras [**1**, **2**, **32**]. Similarly, the algorithms presented in §§ 4 and 6 can be used to construct permutative automorphisms of Leavitt path algebras. Furthermore, the action of permutative endomorphisms on the graph C^*-algebra restricts to the action on the subgroup of permutative unitaries inside the unitary group. This restriction gives rise to interesting endomorphisms and automorphisms of certain locally finite groups (cf. comments in [**13**, § 3.3] about permutative automorphisms of the infinite symmetric group with diagonal embeddings). A more detailed account of this aspect of the theory of endomorphisms of graph algebras will be given elsewhere.

2. Background and notation

2.1. Graphs, graph algebras, and permutative endomorphisms

Let $E = (E^0, E^1, \mathsf{r}, \mathsf{s})$ be a *directed multigraph*, where E^0 and E^1 are countable sets of vertices and edges, respectively, while $\mathsf{s}, \mathsf{r} : E^1 \to E^0$ denote the source and range maps. For notational convenience – and in accordance with the tradition in the graph algebra literature – we will deviate from the graph theoretical terminology and simply call these objects *graphs* hereafter. When multiple graphs are considered at the same time, the notation s_E and r_E will be used to disambiguate to which graph the maps belong. If an edge $e \in E^1$ has $\mathsf{s}(e) = u$ and $\mathsf{r}(e) = v$, we write $e : u \to v$. A *path* μ of *length* l is a sequence of edges $\mu_1 \cdots \mu_l$ for which $\mathsf{r}(\mu_i) = \mathsf{s}(\mu_{i+1})$. If $\mathsf{s}(\mu_i) = v_i$ and $\mathsf{r}(\mu_i) = v_{i+1}$ for vertices $v_1, \ldots, v_{l+1}$, we write $\mu : v_1 \to \cdots \to v_{l+1}$.

For each $l \in \mathbb{N}$, E^l will denote the set of paths in E of length l, and E^* will denote the set of all finite paths. The range and source maps are extended to E^* in the natural way. For $u, v \in E^0$, let $E^l_{u\to v} = \{\alpha \in E^l \mid \mathsf{s}(\alpha) = u, \mathsf{r}(\alpha) = v\}$, $E^l_{u\to *} = \{\alpha \in E^l \mid \mathsf{s}(\alpha) = u\}$, and $E^l_{*\to v} = \{\alpha \in E^l \mid \mathsf{r}(\alpha) = v\}$. A *sink* is a vertex that emits no edges, and a *source* is a vertex that receives no edges. A *cycle* is a path μ with $\mathsf{r}(\mu) = \mathsf{s}(\mu)$, and a *loop* is a cycle of length 1. A cycle $\mu \in E^k$ is said to have an *exit* if there exits $1 \leq i \leq k$ such that $\mathsf{s}(\mu_i)$ emits at least two edges.

https://doi.org/10.1017/S0013091517000189 Published online by Cambridge University Press

The following definition of the graph C^*-algebra corresponding to an arbitrary countable graph was given in [16]. Graph algebras provide a natural generalization of the Cuntz–Krieger algebras [15] and are the subject of widespread investigation by specialists in the theory of operator algebras, symbolic dynamics, non-commutative geometry and quantum groups. The *graph C^*-algebra* [16, 23, 24] of E, denoted $C^*(E)$, is the universal C^*-algebra generated by a collection of mutually orthogonal projections $\{P_v \mid v \in E^0\}$ and partial isometries $\{S_e \mid e \in E^1\}$ satisfying the relations

- $S_e^* S_e = P_{\mathsf{r}(e)}$ and $S_e^* S_f = 0$ when $e \neq f$,
- $S_e S_e^* \leq P_{\mathsf{s}(e)}$,
- $P_v = \sum_{\mathsf{s}(e)=v} S_e S_e^*$ when v emits a non-zero finite number of edges.

For a general introduction to graph C^*-algebras, see [29]. An *endomorphism* of $C^*(E)$ is a unital $*$-homomorphism from $C^*(E)$ into itself. The *Leavitt path algebra* of E is analogously defined as the universal algebraic object satisfying the relations given above, [1, Definition 1.3]. It is not a $*$-algebra, however, so a partial isometry S_e^* must be added to the list of generators for each $e \in E^1$. Also note that since $S_e S_e^*$ and $P_{\mathsf{s}(e)}$ are projections in a C^*-algebra, the inequality $S_e S_e^* \leq P_{\mathsf{s}(e)}$ is equivalent to the algebraic identity $S_e S_e^* P_{\mathsf{s}(e)} = S_e S_e^*$.

For $\mu \in E^k$, let $S_\mu = S_{\mu_1} \cdots S_{\mu_k}$ be the corresponding non-zero partial isometry in $C^*(E)$. The final projection of S_μ is $P_\mu = S_\mu S_\mu^*$, and the initial projection is $P_{\mathsf{r}(\mu)}$. The final projections in $\{P_\mu \mid \mu \in E^*\}$ commute, and the C^*-algebra generated by this set is called the *diagonal subalgebra*, denoted $\mathcal{D}_E$. If every cycle in E has an exit, then $\mathcal{D}_E$ is a maximal abelian subalgebra (MASA) in $C^*(E)$ as shown in [19, Theorem 5.2] and [27, Theorem 3.7]. From now on we make the following assumption.

Standing Assumption. all graphs considered hereafter will be assumed to be finite, to have no sinks or sources and to have an exit from every cycle.

Given $k \in \mathbb{N}$, a permutation $\tau \in \mathrm{Perm}(E^k)$ is said to be *endpoint-fixing* if $\tau(E^k_{u \to v}) = E^k_{u \to v}$ for all $u, v \in E^0$. If $\tau \in \mathrm{Perm}(E^k)$ is endpoint-fixing, then $U_\tau = \sum_{\alpha \in E^k} S_{\tau(\alpha)} S_\alpha^*$ defines a unitary in $C^*(E)$ for which $U_\tau S_\alpha = S_{\tau(\alpha)}$ for all $\alpha \in E^k$. The universality of $C^*(E)$ guarantees that an endomorphism $\lambda_\tau : C^*(E) \to C^*(E)$ can be defined by $\lambda_\tau(S_e) = U_\tau S_e$. Such an endomorphism is said to be a *permutative endomorphism* at *level* k. Note that this part of the construction is not limited to unitaries arising from permutations: it can be carried out for any unitary in the multiplier algebra that commutes with the vertex projections. This is examined in [6]. By the gauge invariant uniqueness theorem, a permutative endomorphism is automatically injective [6, Proposition 2.1].

2.2. Labelled graphs and shift spaces

Given a finite set $\mathcal{A}$, a *labelled* graph with alphabet $\mathcal{A}$ is a pair $(E, \mathcal{L})$ consisting of a graph E and a surjective *labelling map* $\mathcal{L} : E^1 \to \mathcal{A}$. The labelling map naturally extends to a map $\mathcal{L} : E^* \to \mathcal{A}^*$. A labelled graph $(E, \mathcal{L})$ is said to be *left-resolving* if for all $e_1, e_2 \in E^1$, $\mathsf{r}(e_1) = \mathsf{r}(e_2)$ and $\mathcal{L}(e_1) = \mathcal{L}(e_2)$ implies $e_1 = e_2$. A labelled graph $(E, \mathcal{L})$ is said to be *right-resolving* if for all $e_1, e_2 \in E^1$, $\mathsf{s}(e_1) = \mathsf{s}(e_2)$ and $\mathcal{L}(e_1) = \mathcal{L}(e_2)$ implies

https://doi.org/10.1017/S0013091517000189 Published online by Cambridge University Press

$e_1 = e_2$. A labelled graph $(E, \mathcal{L})$ is said to be *left-synchronizing with delay* $m \in \mathbb{N}$ if $\mathsf{s}(\alpha) = \mathsf{s}(\beta)$ whenever $\alpha, \beta \in E^m$ and $\mathcal{L}(\alpha) = \mathcal{L}(\beta)$. In the case where no two parallel edges have the same label, the labelled graph is left-synchronizing if and only if it is *right-closing* (see [**25**, Definition 5.1.4] for a definition of right-closing labelled graphs). Analogously, $(E, \mathcal{L})$ is said to be *right-synchronizing* if there exists m such that any two paths with the same label and length greater than or equal to m must have the same range.

If $e \in E^1$ has $\mathsf{s}(e) = u$, $\mathsf{r}(e) = v$, and $\mathcal{L}(e) = a$, we write $e : u \xrightarrow{a} v$. Similarly, the statement that $\mu : v_1 \to \cdots \to v_n$ is a path over E with labels $\mathcal{L}(\mu_i) = a_i$ may be written as $\mu : v_1 \xrightarrow{a_1} \cdots \xrightarrow{a_{n-1}} v_n$.

For a graph E, the collection of one-sided infinite paths $\mathsf{X}_E^+ = \{e_1 e_2 \cdots \mid e_i \in E^1, \mathsf{r}(e_i) = \mathsf{s}(e_{i+1})\}$ is a *one-sided shift space* equipped with the shift map $\sigma : \mathsf{X}_E^+ \to \mathsf{X}_E^+$ defined by $\sigma(e_1 e_2 e_3 \cdots) = e_2 e_3 \cdots$. Similarly, the collection of bi-infinite paths $\mathsf{X}_E = \{\cdots e_1 e_2 \cdots \mid e_i \in E^1, \mathsf{r}(e_i) = \mathsf{s}(e_{i+1})\}$ is a *two-sided shift space* equipped with the shift map $\sigma : \mathsf{X}_E \to \mathsf{X}_E$ defined by $\sigma(x)_i = x_{i+1}$. X_E is called the *edge shift* of E. For a thorough introduction to the theory of shift spaces, see [**25**]. The shift space *presented* by a labelled graph $(E, \mathcal{L})$ is $\mathsf{X}_{(E,\mathcal{L})} = \{\cdots \mathcal{L}(e_1)\mathcal{L}(e_2) \cdots \mid e_i \in E^1, \mathsf{r}(e_i) = \mathsf{s}(e_{i+1})\}$. The *language* of such a shift space is $\mathcal{L}(E^*)$. The *sliding block code induced by the labelling* is the map $\phi_{\mathcal{L}} : \mathsf{X}_E \to \mathsf{X}_{(E,\mathcal{L})}$ defined by $\phi_{\mathcal{L}}(\cdots e_1 e_2 \cdots) = \cdots \mathcal{L}(e_1)\mathcal{L}(e_2) \ldots$.

It is well known (see e.g. [**33**, Theorem 3.7]) that the Gelfand spectrum of $\mathcal{D}_E$ can be identified with X_E^+ via the identification of $z^+ \in \mathsf{X}_E^+$ with the map $\phi_{z^+} : \mathcal{D}_E \to \mathbb{C}$ as

$$\phi_{z^+}(P_\mu) = \begin{cases} 1, & \mu \text{ is a prefix of } z \\ 0, & \text{otherwise.} \end{cases}$$

Hence, an endomorphism of $C^*(E)$ that preserves $\mathcal{D}_E$ induces an endomorphism of X_E^+.

3. Permutation graphs

3.1. Definition and basic properties

Let E be a finite graph without sinks or sources where every cycle has an exit.

Definition 3.1. Let $k \in \mathbb{N}$ and let $\tau \in \mathrm{Perm}(E^k)$ be an endpoint-fixing permutation. Define a graph $E_\tau = (E_\tau^0, E_\tau^1, \mathsf{s}_\tau, \mathsf{r}_\tau)$ by

$$E_\tau^0 = E^{k-1} \quad \mathsf{s}_\tau(\mu) = \tau(\mu)_1 \cdots \tau(\mu)_{k-1}$$
$$E_\tau^1 = E^k \quad \mathsf{r}_\tau(\mu) = \mu_2 \cdots \mu_k.$$

Define $\mathcal{L}_1, \mathcal{L}_2 : E_\tau^1 \to E^1$ by

$$\mathcal{L}_1(\mu) = \mu_1 \quad \text{and} \quad \mathcal{L}_2(\mu) = \tau(\mu)_k,$$

and define a labelling $\mathcal{L}_\tau : E_\tau^1 \to E^1 \times E^1$ by $\mathcal{L}_\tau(\mu) = [\mathcal{L}_1(\mu), \mathcal{L}_2(\mu)]$. The *permutation graph* of E and τ is defined as the labelled graph $(E_\tau, \mathcal{L}_\tau)$. This will be said to be a permutation graph of E at *level* k.

https://doi.org/10.1017/S0013091517000189 Published online by Cambridge University Press

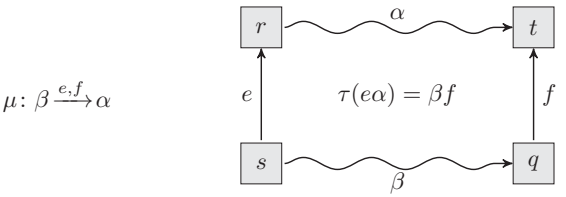


Figure 1. Illustration of the relationship between paths over E and labelled edges in E_τ. For each $e \in E^1$ and $\alpha \in E^{k-1}$ for which $\mathsf{r}_E(e) = \mathsf{s}_E(\alpha)$, the relation $\tau(e\alpha) = \beta f$ is encoded in E_τ as the labelled edge $\mu : \beta \xrightarrow{e,f} \alpha$. The respective sources and ranges in E must be of the form $e : s \to r$, $\alpha \in E^{k-1}_{r\to t}$, $\beta \in E^{k-1}_{s\to q}$, and $f : q \to t$ for appropriate $s, r, q, t \in E^0$, as shown in the diagram.

A permutation is exactly determined by the corresponding permutation graph, since the permutation graph $(E_\tau, \mathcal{L}_\tau)$ contains an edge labelled $[e, f]$ from $\beta \in E^{k-1}$ to $\alpha \in E^{k-1}$ if and only if $\tau(e\alpha) = (\beta f)$. This is illustrated in Figure 1. The permutation graph defined above is introduced as a key tool for investigations of permutative endomorphisms. It constitutes a significant improvement to the graphical devices constructed ad hoc in [**6**, **12**], since it captures the crucial information about the corresponding endomorphism in a form that is visually appealing and easy to work with.

Lemma 3.2. *Let E and τ be as in Definition 3.1 above.*

(1) *Let $e \in E^1$. For each edge $\mu : \beta \xrightarrow{e,f} \alpha$ in E_τ, $\mathsf{s}_E(\beta) = \mathsf{s}_E(e)$ and $\mathsf{s}_E(\alpha) = \mathsf{r}_E(e)$.*

(2) *For every edge $e : s \to r$ in E^1 and $\alpha \in E^{k-1}_{r\to *}$ there is a unique edge $\mu : \beta \xrightarrow{e,f} \alpha$ of $(E_\tau, \mathcal{L}_\tau)$.*

(3) *Let $f \in E^1$. For each edge $\mu : \beta \xrightarrow{e,f} \alpha$ in E_τ, $\mathsf{r}_E(\beta) = \mathsf{s}_E(f)$ and $\mathsf{r}_E(\alpha) = \mathsf{r}_E(f)$.*

(4) *For every edge $f : q \to t$ in E^1 and $\beta \in E^{k-1}_{*\to q}$ there is a unique edge $\mu : \beta \xrightarrow{e,f} \alpha$ of $(E_\tau, \mathcal{L}_\tau)$.*

Proof. The statements (1) and (3) follow from the requirement that τ is endpoint-fixing as seen in Figure 1. Bijectivity of τ entails that e, α uniquely determines β, f, and vice versa, whereby (2) and (4) follow. □

The following corollary is an immediate consequence of Lemma 3.2 (1) and (3).

Corollary 3.3. *Let E and τ be as in Definition 3.1 above. Then $(E_\tau, \mathcal{L}_1)$ is left-resolving and $(E_\tau, \mathcal{L}_2)$ is right-resolving.*

In [**12**], similar conditions were developed to deal specifically with the permutative endomorphisms of the Cuntz algebras $\mathcal{O}_n$. In that case, the conditions were phrased in terms of certain labelled trees carrying the same information as the permutation graph. It was shown that it was sufficient to consider certain configurations of these labelled trees when searching for permutative endomorphisms; the conditions given above generalize

https://doi.org/10.1017/S0013091517000189 Published online by Cambridge University Press

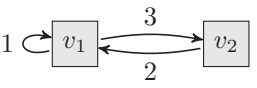


Figure 2. The graph E considered in Example 3.6.

those results. It is straightforward to check that the properties given in Lemma 3.2 are also sufficient to characterize permutation graphs.

Proposition 3.4. *Let $(F, \mathcal{L})$ be a labelled graph with vertex set $F^0 = E^{k-1}$ and labels $\mathcal{L}(F^1) \subseteq E^1 \times E^1$. Then $(F, \mathcal{L})$ is the permutation graph of some endpoint-fixing permutation of E^k if and only if $(F, \mathcal{L})$ satisfies the conditions given in Lemma 3.2.*

The following proposition shows that $(E_\tau, \mathcal{L}_\tau)$ gives two alternative presentations of the edge shift presented by E.

Proposition 3.5. *The two-sided shift spaces X_E, $\mathsf{X}_{(E_\tau, \mathcal{L}_1)}$, and $\mathsf{X}_{(E_\tau, \mathcal{L}_2)}$ presented by E, $(E_\tau, \mathcal{L}_1)$ and $(E_\tau, \mathcal{L}_2)$, respectively, are identical.*

Proof. It is sufficient to prove that the three labelled graphs have the same language. As any other labelling map, $\mathcal{L}_1$ can be extended naturally and considered as a map from E_τ^* to E^* and, using this, the language of $\mathsf{X}_{(E_\tau, \mathcal{L}_1)}$ is $\mathcal{L}_1(E_\tau^*)$. By Lemma 3.2 (2), $\mathcal{L}_1(E_\tau^1) = E^1$. Let $n \geq 1$ be given and assume that $\mathcal{L}_1(E_\tau^n) = E^n$. Let $e \in E^0$. By Lemma 3.2 (1) and (2), $ew \in E^{n+1}$ if and only if $ew \in \mathcal{L}_1(E_\tau^{n+1})$. By induction, $\mathcal{L}_1(E_\tau^*) = E^*$, and an analogous argument proves that $\mathcal{L}_2(E_\tau^*) = E^*$. □

The presentations $(E_\tau, \mathcal{L}_1)$, and $(E_\tau, \mathcal{L}_2)$ do not in general correspond to any standard presentation of X_E known to the authors. However, if the permutation is the identity, then the permutation graph is simply a higher block presentation of the original edge shift (see e.g. [**25**]).

Example 3.6. Consider the graph E shown in Figure 2, and let

$$\tau = (111, 132, 321)(113, 323) \in \mathrm{Perm}(E^3).$$

This is the endpoint-fixing permutation considered in [**6**, Example 6.5]. Its permutation graph $(E_\tau, \mathcal{L}_\tau)$ is shown in Figure 3. It is easy to check that E, $(E_\tau, \mathcal{L}_1)$, and $(E_\tau, \mathcal{L}_2)$ all present the same two-sided shift space, as required by Proposition 3.5.

3.2. Images via permutation graphs

Now, it will be shown how permutation graphs can be used in direct computations involving permutative endomorphisms.

Lemma 3.7. *Let $\tau \in \mathrm{Perm}(E^k)$ be endpoint-fixing, $e \in E^1$ and consider $\{\mu_1, \ldots, \mu_n\} = \mathcal{L}_1^{-1}(e) \subseteq E_\tau^1$. For each i, let $\beta_i = \mathsf{s}_\tau(\mu_i)$, $\alpha_i = \mathsf{r}_\tau(\mu_i)$, and $f_i = \mathcal{L}_2(\mu_i)$, i.e. $\mu_i : \beta_i \xrightarrow{e, f_i} \alpha_i$. Then*

$$\lambda_\tau(S_e) = \sum_{i=1}^{n} S_{\beta_i} S_{f_i} S_{\alpha_i}^*.$$

https://doi.org/10.1017/S0013091517000189 Published online by Cambridge University Press

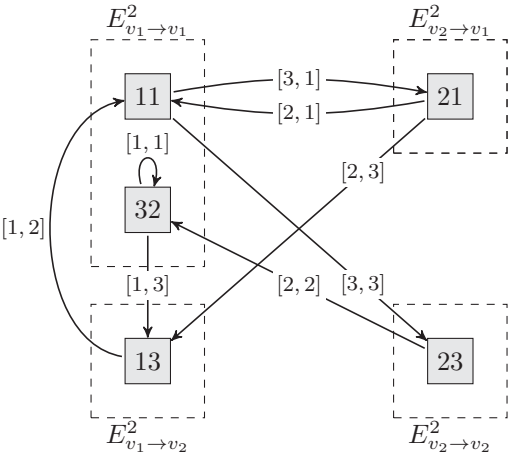


Figure 3. The permutation graph $(E_\tau, \mathcal{L}_\tau)$ considered in Example 3.6. The dashed boxes contain vertices corresponding to paths in E^2 with the specified sources and ranges.

Proof. By the definition of λ_τ and the construction of the permutation graph

$$\lambda_\tau(S_e) = \Big(\sum_{\gamma\in E^k} S_{\tau(\gamma)}S^*_\gamma\Big)S_e = \sum_{(e\alpha)\in E^k} S_{\tau(e\alpha)}S^*_\alpha = \sum_{i=1}^{n} S_{\beta_i}S_{f_i}S^*_{\alpha_i}. \qquad \square$$

This allows the permutation graph to be used as a visually intuitive tool and for direct calculations. To find the image of S_e, one simply needs to identify the edges in the permutation graph for which the first label is e and construct the sum given above.

It is straightforward to extend Lemma 3.7 to an arbitrary path $e_1\cdots e_n$ by applying the result to individual generators S_{e_i} and by using the relations of $C^*(E)$ to achieve cancellations. This leads to the following proposition.

Proposition 3.8. *Let $\tau \in \mathrm{Perm}(E^k)$ be endpoint-fixing. Let $l \in \mathbb{N}$, and let $\gamma \in E^l$. Let $\{A_1, \ldots, A_n\} = \mathcal{L}_1^{-1}(\gamma) \subseteq E^l_\tau$. For each i, let $\beta_i = \mathsf{s}_\tau(A_i)$, $\alpha_i = \mathsf{r}_\tau(A_i)$, and $\delta_i = \mathcal{L}_2(A_i)$. Then*

$$\lambda_\tau(S_\gamma) = \sum_{i=1}^{n} S_{\beta_i}S_{\delta_i}S^*_{\alpha_i}. \tag{3.1}$$

3.3. Permutation graphs as textile systems

A *textile system*, $T = (\Gamma, G, p, q)$, consists of graphs G and $\Gamma = (\Gamma^0, \Gamma^1, \mathsf{s}_\Gamma, r_\Gamma)$ and a pair of graph homomorphisms $p, q : \Gamma \to G$ such that for each edge $a \in \Gamma^1$, the quadruple $(\mathsf{s}_\Gamma(a), \mathsf{r}_\Gamma(a), p(a), q(a))$ uniquely determines a [**28**, p. 14]. Textile systems were introduced by Nasu to investigate endomorphisms and automorphisms of shift spaces, and a powerful set of tools has been developed for this purpose. A textile system $T = (\Gamma, G, p, q)$ is said to be *in the standard form* if Γ and G have no sinks or sources and the graph homomorphisms p, q are onto [**28**, p. 18].

https://doi.org/10.1017/S0013091517000189 Published online by Cambridge University Press

Standing Assumption. from now on, all textile systems will be assumed to be in the standard form.

For a textile system $T = (\Gamma, G, p, q)$, define maps $\phi_1, \phi_2 : \mathsf{X}_\Gamma \to \mathsf{X}_G$ by

$$\phi_1(\dots a_{-1}a_0a_1\dots) = \dots p(a_{-1})p(a_0)p(a_1)\dots$$
$$\phi_2(\dots a_{-1}a_0a_1\dots) = \dots q(a_{-1})q(a_0)q(a_1)\dots$$

Define $X_0 = \mathsf{X}_G$ and $Z_0 = \mathsf{X}_\Gamma$. For each $l \in \mathbb{N}$, define

$$X_l = \phi_1(Z_{l-1}) \cap \phi_2(Z_{l-1})$$
$$Z_l = \phi_1^{-1}(X_l) \cap \phi_2^{-1}(X_l),$$

and let

$$\mathsf{X}_T = \bigcap_{l=0}^{\infty} X_k \quad \text{and} \quad \mathsf{Z}_T = \bigcap_{l=0}^{\infty} Z_k.$$

The textile system T is said to be *non-degenerate* if $\mathsf{X}_T = \mathsf{X}_G$ (from which it follows that $\mathsf{Z}_T = \mathsf{X}_\Gamma$) [**28**, p. 19]. For a non-degenerate textile system, the maps $\phi_1, \phi_2 : \mathsf{X}_\Gamma \to \mathsf{X}_G$ are surjective. In all cases, the restrictions $\phi_1|_{\mathsf{Z}_T}, \phi_2|_{\mathsf{Z}_T} : \mathsf{Z}_T \to \mathsf{X}_T$ have image X_T.

A non-degenerate textile system, T, is said to be *one-sided one-to-one* if ϕ_1 is injective and *two-sided one-to-one* if both ϕ_1 and ϕ_2 are injective [**28**, p. 19]. If T is one-sided one-to-one, then the map $\phi_T = \phi_2 \circ \phi_1^{-1} : \mathsf{X}_G \to \mathsf{X}_G$ is a shift space endomorphism, and this endomorphism is said to be *coded by* T. If T is two-sided one-to-one, then this is an automorphism of shift spaces.

A graph homomorphism $h : \Gamma \to G$ is said to be *right-resolving* if for each $\beta \in \Gamma^0$ and $e \in G^1$ with $\mathsf{s}_G(e) = h(\beta)$ there exists a unique $\mu \in \Gamma^1$ such that $\mathsf{s}_\Gamma(\mu) = \beta$ and $h(\mu) = e$ [**28**, p. 40]. Note that (Γ, h) is a right-resolving labelled graph with alphabet G^1 when h is a right-resolving graph homomorphism (but the converse is not always true). Left-resolving graph homomorphisms are defined analogously. A textile system is said to be *LR* if p is a left-resolving graph homomorphism and q is a right-resolving graph homomorphism [**28**, p. 40]. It is straightforward to check that an LR textile system is always non-degenerate [**28**, Fact 3.3].

Given a permutation graph $(E_\tau, \mathcal{L}_\tau)$ over a graph E, define maps $p_\tau, q_\tau : E_\tau \to E$ by

$$p_\tau(\mu) = \mathcal{L}_1(\mu) \quad p_\tau(\beta) = \mathsf{s}_E(\beta)$$
$$q_\tau(\mu) = \mathcal{L}_2(\mu) \quad q_\tau(\beta) = \mathsf{r}_E(\beta)$$

for all $\mu \in E_\tau^1$ and $\beta \in E_\tau^0 = E^{k-1}$, and let $T_\tau = (E_\tau, E, p_\tau, q_\tau)$. Lemma 3.2 implies that p_τ, q_τ are graph homomorphisms and that $(\mathsf{s}_\tau(\mu), \mathsf{r}_\tau(\mu), p_\tau(\mu), q_\tau(\mu))$ uniquely determines each $\mu \in E_\tau^1$, so T_τ is a textile system over E. Furthermore, Lemma 3.2 implies that this textile system is LR, and hence non-degenerate. The connection between the shift space endomorphism ϕ_{T_τ} encoded by T_τ when T_τ is one-sided one-to-one and the endomorphism λ_τ of $C^*(E)$ will be explored in the following sections.

https://doi.org/10.1017/S0013091517000189 Published online by Cambridge University Press

3.4. Automorphism criteria

Let λ_τ be a permutative endomorphism of $C^*(E)$ at level k. Combinatorial conditions for the invertibility of λ_τ have been developed in [6]. These conditions can be reformulated in terms of the permutation graph, but here an independent formulation is given, together with a new proof of the most interesting implication.

Remark 3.9. To make the connection to [6] explicit, this remark gives a short discussion of the terminology used in that paper. In the notation of [6], define for each $e \in E^1$ the map $f_e : E^{k-1}_{\mathsf{r}(e)\to *} \to E^{k-1}_{\mathsf{s}(e)\to *}$ by $f_e(\alpha) = \mathsf{s}_\tau(e\alpha)$. The graph $(E_\tau, \mathcal{L}_1)$ has an edge labelled e from $\beta \in E^{k-1}$ to $\alpha \in E^{k-1}$ if and only if $f_e(\alpha) = \beta$. Hence, reversing the edges on $(E_\tau, \mathcal{L}_1)$ will yield the graph considered in [6]. As an example of this, notice how the arrows of the permutation graph considered in Example 3.6 have been reversed compared with the corresponding graph from [6, Example 6.5].

The following lemma will help connect the theory of permutative endomorphisms to the theory of textile systems. An analogous result holds for right-resolving graphs.

Lemma 3.10. *Let $(G, \mathcal{L}_1)$ be a finite left-resolving labelled graph. The following are equivalent.*

(1) *The one-block code induced by the labelling of $(G, \mathcal{L}_1)$ is invertible.*

(2) *$(G, \mathcal{L}_1)$ is left-synchronizing.*

(3) *No distinct cycles admitted by $(G, \mathcal{L}_1)$ have the same label.*

Proof. (1) $\implies$ (3): If there are two distinct cycles with the same label μ, then the block code induced by the labelling is not injective, since the bi-infinite sequence $\ldots\mu\mu\mu\ldots$ has two presentations. (3) $\implies$ (2): Assume that $(G, \mathcal{L}_1)$ is not left synchronizing. This implies that there exists a label sequence μ with $|\mu| > (G^0)^2$ such that there are $x, y \in G^*$ with $\mathcal{L}_1(x) = \mu = \mathcal{L}_1(y)$ and $s(x) \neq s(y)$. Let $u_i = r(x_i)$ and $v_i = r(y_i)$. Because $|\mu| > (G^0)^2$, there exist $1 \leq i < j \leq |\mu|$ such that $(u_i, v_i) = (u_j, v_j)$. Let $\nu = \mu_{i+1}\cdots\mu_j$. Then the cycles $x_{i+1}\ldots x_j$ and $y_{i+1}\ldots y_j$ are both labelled ν. If these paths were identical, then $s(x)$ would be equal to $s(y)$ because the graph is left-resolving.

(2) $\implies$ (1): Choose n such that any word of length n synchronizes. Because the graph is left-resolving, any sequence of labels of length $n+1$ will have a unique first edge. This defines a sliding block inverse to the map induced by the labelling. □

A permutation graph $(E_\tau, \mathcal{L}_\tau)$ is said to be *synchronizing in the first label* if $(E_\tau, \mathcal{L}_1)$ is left-synchronizing. Similarly, a permutation graph is said to be *synchronizing in the second label* if $(E_\tau, \mathcal{L}_2)$ is right-synchronizing. In [6], equivalent conditions were denoted property (b) and property (d), respectively.

We recall from [6, Proposition 2.1] that every permutative endomorphism λ_τ is automatically injective. Indeed, $\lambda_\tau(P_v) = U_\tau P_v U_\tau^* \neq 0$ for each $v \in E^0$. Also, $\gamma_t(\lambda_\tau(S_e)) = \gamma_t(U_\tau S_e) = tU_\tau S_e = \lambda_\tau(\gamma_t(S_e))$ for all $e \in E^1$ and $t \in U(1)$. Hence, λ_τ commutes with every gauge automorphism γ_t of $C^*(E)$, whereby λ_τ is injective by [29, Theorem 2.2].

https://doi.org/10.1017/S0013091517000189 Published online by Cambridge University Press

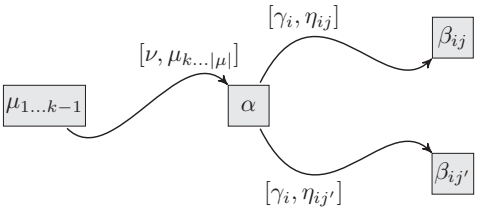


Figure 4. Construction of the paths considered in the proof of Lemma 3.12.

Proposition 3.11. *If a permutation graph $(E_\tau, \mathcal{L}_\tau)$ is synchronizing in the first or second label with delay n, then any path $A \in E_\tau^m$ with $m \geq n$, is uniquely determined by $\mathcal{L}_\tau(A)$.*

Proof. Assume $(E_\tau, \mathcal{L}_\tau)$ is synchronizing in the first label with delay n, and consider a length m path $A = e_1 \cdots e_m \in E_\tau^m$ with $m \geq n$. By synchronization, $\mathcal{L}_\tau(A)$ determines $\mathsf{s}(e_1)$. Because $(E_\tau, \mathcal{L}_2)$ is right-resolving by Corollary 3.3, $\mathsf{s}(e_i)$ together with $\mathcal{L}_2(e_i)$ determines e_i for $1 \leq i \leq m$, and since $\mathsf{r}(e_i) = \mathsf{s}(e_{i+1})$ for $i \leq m-1$, we recover all of A. The case when $(E_\tau, \mathcal{L}_\tau)$ is synchronizing in the second label is proved symmetrically. □

Lemma 3.12. *Let $(E_\tau, \mathcal{L}_\tau)$ be a permutation graph. The following are equivalent.*

(1) *The labelled graph $(E_\tau, \mathcal{L}_\tau)$ is synchronizing in the first label.*

(2) *T_τ is one-sided one-to-one.*

(3) *The endomorphism $\lambda_\tau : C^*(E) \to C^*(E)$ restricts to an automorphism of $\mathcal{D}_E$.*

Proof. (1) $\iff$ (2): By Lemma 3.10, $(E_\tau, \mathcal{L}_1)$ is left-synchronizing if and only if the sliding block code induced by the labelling is invertible.

(1) $\implies$ (3): Let τ be an endpoint-fixing permutation of E^k, and assume that $(E_\tau, \mathcal{L}_1)$ is left-synchronizing with delay l. The permutative endomorphism λ_τ is automatically injective, so it is sufficient to prove surjectivity. To this end, it is enough to find preimages under λ_τ of all P_μ with $|\mu| \geq k$, since the linear span of such projections is dense in $\mathcal{D}_E$. Let $\mu \in E^*$ with $|\mu| \geq k$ be given. The aim of the following is to use Proposition 3.8 to construct an element $x \in \mathcal{D}_E$ for which $\lambda_\tau(x) = P_\mu$. This process is illustrated in Figure 4. By Lemma 3.2, the vertex $\mu_{1\ldots k-1} \in E_\tau^0 = E^{k-1}$ emits a unique path with second label $\mu_{k\ldots|\mu|}$. Let ν be the first label of this path (i.e. its image under $\mathcal{L}_1$) and let $\alpha \in E_\tau^0 = E^{k-1}$ be the range.

In order to use Proposition 3.8, it is necessary to find paths in the permutation graph with the required first and second labels. By Lemma 3.2, there is a unique path in E_τ^* with source α and second label $\eta \in E^*$ precisely when $\mathsf{s}_E(\eta) = \mathsf{r}_E(\alpha)$. Consider the paths in E_τ^* with source α and length l. Let $\{\gamma_i\}$ be the set of first labels for all such paths. Left-synchronization implies that any path in E_τ^* with first label γ_i must start at α. This is illustrated in Figure 4 for a single γ_i. For each i, there may be more than one path with first label γ_i, but the second label uniquely identifies the path by Lemma 3.2. Let $\{\eta_{ij}\}$ be the set of second labels of paths with first label γ_i, and let β_{ij} be the range of the unique path with second label η_{ij} in E_τ. Because $(E_\tau, \mathcal{L}_1)$ is left-resolving, $\beta_{ij} \neq \beta_{ij'}$ whenever $j \neq j'$.

https://doi.org/10.1017/S0013091517000189 Published online by Cambridge University Press

The left-synchronization and left-resolvancy of $(E_\tau, \mathcal{L}_1)$ implies that any path with first label $\nu\gamma_i$ must have $\mu_{k\ldots|\mu|}$ as a prefix of the second label. By Proposition 3.8,

$$\lambda_\tau\left(\sum_i S_\nu S_{\gamma_i} S_{\gamma_i}^* S_\nu^*\right) = S_{\mu_{1\ldots k-1}} S_{\mu_{k\ldots|\mu|}} \sum_{i,j,j'} (S_{\eta_{ij}} S_{\beta_{ij}}^* S_{\beta_{ij'}} S_{\eta_{ij'}}^*) S_{\mu_{k\ldots|\mu|}}^* S_{\mu_{1\ldots k-1}}^*$$
$$= S_\mu \sum_{ij} (S_{\eta_{ij}} S_{\eta_{ij}}^*) S_\mu^* = S_\mu S_\mu^*.$$

As μ was arbitrary, this shows that λ_τ restricts to a surjection of $\mathcal{D}_E$.

(3) $\implies$ (1): This is a restatement of a result proved in [**6**, Lemma 6.1] under the identification given in Remark 3.9, since existence of the partial order $\leq$ described therein is equivalent to the graph $(E_\tau, \mathcal{L}_\tau)$ being synchronizing in the first label. □

Theorem 3.13. *The following are equivalent.*

(1) $(E_\tau, \mathcal{L}_\tau)$ *is synchronizing in both the first and the second label.*

(2) T_τ *is two-sided one-to-one.*

(3) λ_τ *is an automorphism of* $C^*(E)$.

Furthermore, such an automorphism has a permutative inverse.

Proof. (1) $\iff$ (2): As in the proof of Lemma 3.12, this follows from Lemma 3.10.

(1) $\implies$ (3): λ_τ is automatically injective, as described above in Lemma 3.12. To prove surjectivity, choose l such that $(E_\tau, \mathcal{L}_1)$ and $(E_\tau, \mathcal{L}_2)$ are respectively left- and right-synchronizing, both with delay l. It suffices to find preimages under λ_τ of elements of the form $S_\mu S_\nu^*$ with $|\mu|, |\nu| \geq k$, since their linear span is dense in $C^*(E)$. Let $\mu, \nu \in E^*$ have the same endpoint $\mathsf{r}(\mu) = \mathsf{r}(\nu)$, and $|\mu|, |\nu| \geq k$. As in the proof of Lemma 3.12, the aim is to use Proposition 3.8 to construct an element x, such that $\lambda_\tau(x) = S_\mu S_\nu^*$. This process is illustrated in Figure 5. Such elements will then be used to construct a permutative unitary U_π for which λ_π is the inverse of λ_τ.

Consider the unique path in E_τ^* with source node $\mu_{1\ldots k-1}$ and second label $\mu_{k\ldots|\mu|}$. Let $\bar{\mu}$ be the first label of this path, and let $\alpha \in E_\tau^0 = E^{k-1}$ be the range. Similarly, consider the unique path in E_τ^* with source $\nu_{1\ldots k-1}$ and second label $\nu_{k\ldots|\nu|}$. Let $\bar{\nu}$ be the first label of this path, and let $\beta \in E_\tau^0 = E^{k-1}$ be the range.

By Lemma 3.2, for each $\gamma \in E^*$ with $\mathsf{s}_E(\gamma) = \mathsf{r}_E(\mu)$, there is precisely one path in E_τ^* with second label γ and source α. Similarly, there is precisely one path in E_τ^* with second label γ and source β. Let $\{\gamma_i\}$ be the paths in E^l with $\mathsf{s}_E(\gamma) = \mathsf{r}_E(\mu) = \mathsf{r}_E(\nu)$. Given such a γ_i, let $\bar{\gamma}_i^1$ be the first label of the unique path in E_τ^* with source α and second label γ_i. Similarly, let $\bar{\gamma}_i^2$ be the first label of the unique path in E_τ^* with source β and second label γ_i. Right-synchronization of $(E_\tau, \mathcal{L}_2)$ implies that these two paths have the same range ζ_i. This is shown in Figure 5 for a single path γ_i.

Left-synchronization of $(E_\tau, \mathcal{L}_1)$ implies that any path in E_τ^* with first label $\bar{\gamma}_i^1$ must start at α and right-resolvancy of $(E_\tau, \mathcal{L}_1)$ means that all these paths have different second labels. Let $\{\gamma_i\} \cup \{\eta_{ij}\}$ be the set of second labels of these paths, and let δ_{ij} be the range of the path with second label η_{ij}. This is illustrated by a dashed path in Figure 5 for a

https://doi.org/10.1017/S0013091517000189 Published online by Cambridge University Press

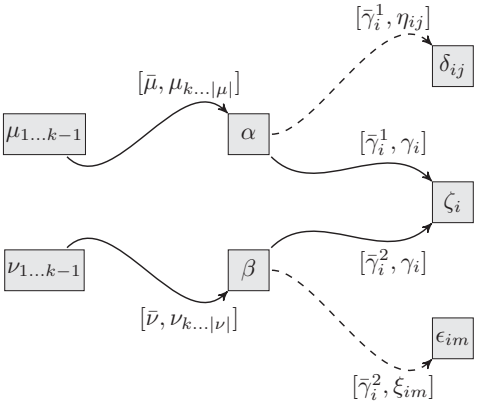


Figure 5. Construction of the paths considered in the proof of Theorem 3.13 for a given γ_i.

single value of j. Similarly, any path in E_τ^* with first label $\bar{\gamma}_i^2$ must start at β, and all these paths have unique second labels. Let $\{\gamma_i\} \cup \{\xi_{im}\}$ be the the set of second labels of these paths, and let ϵ_{im} be the range of the path with second label ξ_{im}. As above, this is illustrated by a dashed path in Figure 5 for a single value of m.

For each γ_i with $\mathsf{s}_E(\gamma_i) = \mathsf{r}_E(\mu) = \mathsf{r}_E(\nu)$ and $|\gamma_i| = l$, use Lemma 3.12 to choose $q_{\mu\gamma_i}$ and $q_{\nu\gamma_i}$ such that $\lambda_\tau(q_{\mu\gamma_i}) = P_{\mu\gamma_i}$ and $\lambda_\tau(q_{\nu\gamma_i}) = P_{\nu\gamma_i}$. By Proposition 3.8,

$$\begin{aligned}\lambda_\tau\Big(\sum_i q_{\mu\gamma_i} S_{\bar{\mu}} S_{\bar{\gamma}_i^1} S_{\bar{\gamma}_i^2}^* S_{\bar{\nu}}^* q_{\nu\gamma_i}\Big) &= \sum_i P_{\mu\gamma_i} S_\mu \Big(S_{\gamma_i} S_{\zeta_i}^* + \sum_j S_{\eta_{ij}} S_{\delta_{ij}}^*\Big) \\ &\quad \times \Big(S_{\zeta_i} S_{\gamma_i}^* + \sum_m S_{\epsilon_{im}} S_{\xi_{im}}^*\Big) S_\nu^* P_{\nu\gamma_i} \\ &= \sum_i S_{\mu\gamma_i} S_{\nu\gamma_i}^* = S_\mu S_\nu^*.\end{aligned}$$

By the above, there exist $\eta_j, \omega_j \in E^*$ such that $\lambda_\tau(U_\pi) = U_\tau^*$, where $U_\pi = \sum_j S_{\eta_j} S_{\omega_j}^*$. Since λ_τ is an automorphism of $C^*(E)$ that commutes with the gauge-action, which fixes the unitary U_τ^*, U_π is again a unitary fixed by the gauge-action. Consequently, U_π is a permutative unitary, and we have $\lambda_\pi = \lambda_\tau^{-1}$, [**6**, p. 7].

(3) $\implies$ (1): This is a restatement of a result proved in [**6**, Theorem 6.4] under the identification given in Remark 3.9. Indeed, the partial order $\leq$ described in [**6**, Lemma 6.3] exists if and only if the graph $(E_\tau, \mathcal{L}_\tau)$ is synchronizing in the second label. Further, since $\lambda_\tau(\mathcal{D}_E) \subseteq \mathcal{D}_E$, if λ_τ is an automorphism of $C^*(E)$, then it restricts to an automorphism of MASA $\mathcal{D}_E$. Thus, by Lemma 3.12, $(E_\tau, \mathcal{L}_\tau)$ is synchronizing in the first label. □

Note how this result shows that for a permutative endomorphism λ_τ of $C^*(E)$, invertibility of the induced endomorphism of the shift space X_E can be lifted and used to constructively find an inverse to λ_τ.

Example 3.14. Consider again the graph E and the permutation τ from Example 3.6. It is simple to check that for any path A in E_τ^3, the label $\mathcal{L}_1(A)$ uniquely determines the

https://doi.org/10.1017/S0013091517000189 Published online by Cambridge University Press

source $\mathsf{s}_\tau(A)$, so the permutation graph is synchronizing in the first label. On the other hand, $(E_\tau, \mathcal{L}_2)$ contains a cycle labelled 11 and a loop labelled 1, so by Lemma 3.10, the permutation graph is not synchronizing in the second label. Hence, λ_τ is not an automorphism. This was also observed in [**6**] without reference to the permutation graph.

Remark 3.15. As in Example 3.14, Lemmas 3.10, 3.12 and Theorem 3.13 are useful in general for automorphism testing. A forthcoming paper will present an efficient algorithm that decides whether a permutative endomorphism λ_τ is an automorphism, by way of its permutation graph [**4**]. This is used in the computer program discussed in § 7 together with the algorithm from § 6 to perform exhaustive searches for permutative automorphisms at higher levels than otherwise possible.

3.5. Induced endomorphism of the one- and two-sided shift

A permutative endomorphism that restricts to an automorphism of $\mathcal{D}_E$, i.e. its permutation graph is synchronizing in the first label, induces an endomorphism ϕ_τ^+ on the one-sided edge shift X_E^+ through the identification of $z^+ \in \mathsf{X}_E^+$ with the map $\phi_{z^+} : \mathcal{D}_E \to \mathbb{C}$. This in turn induces an endomorphism ϕ_τ on the two-sided edge shift X_E.

Proposition 3.16. *Let λ_τ be a permutative endomorphism at level k for which the permutation graph is synchronizing in the first label, and let ϕ_τ^+ be the induced endomorphism of the one-sided edge shift X_E^+. Given $x^+ \in \mathsf{X}_E^+$, let $\beta \in E^{k-1}$ be the source of the unique infinite path in E_τ^* with first label x^+, and let y^+ be the second label of this path. Then $\phi_\tau^+(x^+) = \beta y^+$.*

Proof. Synchronization in the first label guarantees that there is a unique right-infinite path in the permutation graph with first label x^+. The result follows from Proposition 3.8 and the identification of $z^+ \in \mathsf{X}_E^+$ with the map $\phi_{z^+} : \mathcal{D}_E \to \mathbb{C}$. □

As described in § 3.3, the textile system T_τ corresponding to a permutation graph codes an endomorphism ϕ_{T_τ} of X_E. As an immediate consequence of the previous result, this endomorphism can be linked to the natural endomorphism of X_E induced by λ_τ.

Corollary 3.17. *Let λ_τ be a permutative endomorphism at level k for which the permutation graph is synchronizing in the first label, let ϕ_τ be the induced endomorphism of the two-sided edge shift X_E, and let ϕ_{T_τ} be the endomorphism coded by T_τ. Then $\phi_\tau(x) = \sigma^{-k+1} \circ \phi_{T_\tau}$.*

3.6. Composition of permutation graphs

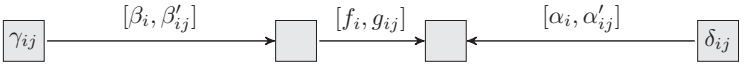
Let λ_τ and λ_π be permutative endomorphisms at level l and k, respectively. Given $e \in E^1$, consider all $\alpha_i, \beta_i \in E^{k-1}$ and $f_i \in E^1$ for which E_τ contains an edge $\beta_i \xrightarrow{e,f_i} \alpha_i$. By Lemma 3.7, $\lambda_\pi(S_e) = \sum_i S_{\beta_i} S_{f_i} S_{\alpha_i}$.

https://doi.org/10.1017/S0013091517000189 Published online by Cambridge University Press

For each i, let j enumerate the pairs of paths A_{ij}, B_{ij} over $(E_\tau, \mathcal{L}_\tau)$ for which

- the first label of A_{ij} is α_i,
- the first label of B_{ij} is $\beta_i f_i$, and
- A_{ij} and B_{ij} have the same range.

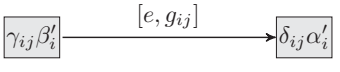
For each j, let δ_{ij} be the source of A_{ij}, let γ_{ij} be the source of B_{ij}, let α'_{ij} be the second label of A_{ij}, and let $\beta'_{ij}g_{ij}$ be the second label of B_{ij}. Such a pair of paths is illustrated here:



By Proposition 3.8,

$$\lambda_\tau(\lambda_\pi(S_e)) = \sum_{i,j} S_{\gamma_{ij}} S_{\beta'_{ij}} S_{g_{ij}} S^*_{\alpha'_{ij}} S^*_{\delta_{ij}},$$

so the permutation graph of $\lambda_\tau \circ \lambda_\pi$ will have vertices and edges as illustrated here:



Note that this has the form of a permutation graph for a permutation at level $k + l - 1$. The corresponding permutation is determined by this permutation graph, but it is not easily computed directly from τ and π without going through this process. The following result summarizes this discussion.

Proposition 3.18. *Let λ_τ and λ_π be permutative endomorphisms. Then the permutation graph of $\lambda_\tau \circ \lambda_\pi$ is the labelled graph constructed above.*

4. Finding permutative automorphisms

This section describes a recursive algorithm that, given a graph E and a level k, finds all level-k permutative automorphisms of $C^*(E)$. Only a high-level description is given here; a detailed description of the algorithm will be published separately. The overall structure follows two simple mutually recursive functions Φ_1 and Φ_0, defined in Equation (4.1), that build all valid extensions to a partial graph $\mathcal{G}$ edge by edge. It will be shown below that $\Phi_1(0, \mathcal{G}_0)$ precisely constructs the set of permutation graphs of level k when $\mathcal{G}_0$ is the empty labelled graph with vertex set E^{k-1}. For each $r \in E^0$, write $E^{k-1}_{r\to *} = \{\alpha^n_r \mid 0 \le n \le |E^{k-1}_{r\to *}| - 1\}$ and $E^1 = \{e_m \mid 0 \le m \le |E^1| - 1\}$. Let $r_m = \mathsf{r}_E(e_m)$

https://doi.org/10.1017/S0013091517000189 Published online by Cambridge University Press

for each m, and let $\mathcal{G} = (G, \mathcal{L})$ denote a partially completed permutation graph.

$$\begin{aligned} \Phi_1(|E^1|, \mathcal{G}) &= \{\mathcal{G}\} \\ \Phi_1(m, \mathcal{G}) &= \Phi_0(m, 0, \mathcal{G}) \\ \Phi_0(m, |E^{k-1}_{r\to *}|, \mathcal{G}) &= \Phi_1(m+1, \mathcal{G}) \\ \Phi_0(m, n, \mathcal{G}) &= \biguplus_{\substack{\mu:\beta \xrightarrow{e_m, f} \alpha^n_{r_m} \\ \mathsf{valid}(\mathcal{G}\oplus\mu)}} \Phi_0(m, n+1, \mathcal{G} \oplus \mu). \end{aligned} \tag{4.1}$$

In the above, '$\uplus$' denotes disjoint set union, and $\mathcal{G} \oplus \mu$ is the labelled graph resulting from extending $\mathcal{G}$ with the labelled edge μ. Each edge μ is added only if $\mathcal{G} \oplus \mu$ satisfies a predicate $\mathsf{valid}\,(\cdot)$ defined below, which guarantees that the construction only results in permutation graphs for automorphisms. By Lemma 3.2, each vertex $\alpha \in E^{k-1}_{r\to *}$ must receive exactly one edge with first label e for each $e : s \to r$ in E^1. In the algorithm defined above, the graphs are constructed one label $e_m : s_m \to r_m$ at a time, placing for each destination vertex $\alpha^n_{r_m}$ the e_m-labelled edge incident to it. For each β and f for which adding the edge $\mu : \beta \xrightarrow{e_m, f} \alpha^n_{r_m}$ results in a valid subgraph of a permutation graph, the recursion proceeds. Dead ends result in the empty set, but if all labels are successfully completed, the singleton set containing the completed graph is returned. At each recursion level, the result is the disjoint union of the results from the levels below.

The predicate $\mathsf{valid}\,(\cdot)$ ensures that it is exactly the permutative automorphisms that are constructed. It guarantees that only edges that satisfy two concurrent conditions are placed: the resulting graph must be a subgraph of a permutation graph and, by Theorem 3.13, it must also synchronize in both labels in order to complete to an automorphism. The following proposition, which is an easy corollary to Proposition 3.4, characterizes the subgraphs of permutation graphs.

Proposition 4.1. *Let $(G, \mathcal{L})$ be a labelled graph with node set E^{k-1} and label alphabet $E^1 \times E^1$. $(G, \mathcal{L})$ is a subgraph of a permutation graph $(E_\tau, \mathcal{L}_\tau)$ for some endpoint-fixing permutation $\tau \in \mathrm{Perm}(E^k)$ if and only if*

(1) *$(G, \mathcal{L})$ satisfies parts (1) and (3) of Lemma 3.2,*

(2) *for every edge $e : s \to r$ in E^1 and $\alpha \in E^{k-1}_{r\to *}$, there is at most one edge $\mu : \beta \xrightarrow{e,f} \alpha$ in $(G, \mathcal{L})$ with $f \in E^1$ and $\beta \in E^{k-1}$, and*

(3) *for every edge $f : q \to t$ in E^1 and $\beta \in E^{k-1}_{*\to q}$, there is at most one edge $\mu : \beta \xrightarrow{e,f} \alpha$ in $(G, \mathcal{L})$ with $e \in E^1$ and $\alpha \in E^{k-1}$.*

This yields a simple test for whether a partially completed graph is a subgraph of some permutation graph that corresponds to an automorphism.

Definition 4.2. Let $\mathcal{G} = (G, \mathcal{L})$ be a labelled graph with node set E^{k-1} and label alphabet $E^1 \times E^1$. $\mathsf{valid}\,(\mathcal{G})$ is true if $\mathcal{G}$ synchronizes in both labels and satisfies the conditions of Proposition 4.1.

https://doi.org/10.1017/S0013091517000189 Published online by Cambridge University Press

The correctness of the recursive construction of all permutative automorphisms depends on the following property of the $\mathsf{valid}(\cdot)$ predicate.

Definition 4.3. A predicate p on the set of labelled graphs is said to be inherited if $\mathsf{p}(H,\mathcal{L}) \implies \mathsf{p}(G,\mathcal{L})$ for every labelled graph $(H,\mathcal{L})$ and subgraph G of H.

That is, the property is invariant throughout an edge-by-edge construction of the labelled graph. As soon as a labelled edge is placed that makes the partially constructed graph invalid, the entire search tree below it may be safely discarded, as it can be contained in no valid completed graph. Conversely, any valid completed graph is reached from the empty graph (or any other subgraph) by a sequence of locally valid edge placements, the ordering of which is unimportant.

It is easy to verify that Properties (1)–(3) of Proposition 4.1 are inherited properties. Left- and right-synchronization are inherited properties by way of Lemma 3.10, since the set of cycles over H is a subset of the cycles over G when H is a subgraph of G.

Theorem 4.4. *Let $\mathcal{G}_0$ be the labelled graph with vertex set E^{k-1} and no edges. Then $\Phi_1(0,\mathcal{G}_0)$ is the set of permutation graphs corresponding to all level k permutative automorphisms.*

Proof. Assume that $\mathcal{G} \in \Phi_1(0,\mathcal{G}_0)$. It is immediately apparent from Equation (4.1) that the total number of edges in $\mathcal{G} = (G,\mathcal{L})$ must be

$$\left|G^1\right| = \sum_{\substack{e:s\to r \\ e\in E^1}} \left|E^{k-1}_{r\to *}\right| = \sum_{\substack{f:q\to t \\ f\in E^1}} \left|E^{k-1}_{*\to q}\right|.$$

Proposition 4.1(2) ensures that for each $e : s \to r$, every $\alpha \in E^{k-1}_{r\to *}$ receives at most one edge with first label e, and the full edge count can only be achieved if each α receives exactly one such edge, fulfilling Lemma 3.2(2). Similarly, Proposition 4.1(3) and the second equality above ensure that Lemma 3.2(4) is satisfied. Thus, $\mathcal{G}$ satisfies all properties of Lemma 3.2 and, by Proposition 3.4, it is a permutation graph. Since it also synchronizes in both labels, it represents a permutative automorphism by Theorem 3.13.

Conversely, let $(E_\tau,\mathcal{L}_\tau)$ be a permutative automorphism at level k. By Proposition 4.1 and the fact that $\mathsf{valid}(\cdot)$ is inherited, the edge set E^1_τ traversed in any order constitutes a sequence of locally valid edge placements that incrementally extends $\mathcal{G}_0$ until $(E_\tau,\mathcal{L}_\tau)$ is completed. Hence, each time the last line in Equation (4.1) is reached, there is a valid edge placement to pick from E^1_τ. Since the procedure terminates after $\sum_{e\in E}\left|E^{k-1}_{\mathrm{r}_E(e)\to *}\right| = \left|E^1_\tau\right|$ steps, all the edges of E^1_τ are eventually placed, and thus $(E_\tau,\mathcal{L}_\tau) \in \Phi_1(0,\mathcal{G}_0)$.

Consequently, the procedure constructs exactly the set of all level-k permutative automorphisms. □

From the recursion structure in Equation (4.1), it is apparent that each permutation graph is built at most once. Because all level-k permutative automorphisms are reached, each is constructed exactly once.

It is instructive to note that the permutation graph property (Lemma 3.2) and the automorphism property (Theorem 3.13) are separate tests: simply omitting the synchronization test in the above procedure would instead yield the set of all permutative

https://doi.org/10.1017/S0013091517000189 Published online by Cambridge University Press

endomorphisms at level k. While this set becomes too large to practically compute even for small k, other properties than automorphism may be filtered for, as long as they can be expressed in an inherited form as defined in Definition 4.3.

5. Inner equivalence, shift space equivalence, and ordering

One is most often not interested in every automorphism, but only in those that are sufficiently different to warrant distinct consideration. Commonly, endomorphisms are considered up to inner equivalence, i.e. modulo conjugation by a unitary. The present section introduces *shift space equivalence*, which groups endomorphisms that have identical properties as dynamical systems.

Definition 5.1 (Shift space equivalence). The labelled graphs $(E, \mathcal{L}_E)$ and $(F, \mathcal{L}_F)$ are shift space equivalent if they present the same shift space $\mathsf{X}_{(E,\mathcal{L}_E)} = \mathsf{X}_{(F,\mathcal{L}_E)}$.

Definition 5.1 induces an equivalence relation on $\mathrm{Perm}(E^k)$ and on the corresponding permutative endomorphisms. Two level-k endpoint-fixing permutations τ and τ' are said to be shift space equivalent when their respective permutation graphs are equivalent, i.e. when $\mathsf{X}_{(E_\tau,\mathcal{L}_\tau)} = \mathsf{X}_{(E_{\tau'},\mathcal{L}_{\tau'})}$. For permutative endomorphisms that are synchronizing in the first label, it will be shown that shift space equivalence is the same as inner equivalence via a permutative unitary.

5.1. Permuting permutations

Let $\tau \in \mathrm{Perm}(E^k)$ be endpoint-fixing. Given an endpoint-fixing permutation $\pi \in \mathrm{Perm}(E^{k-1})$, the aim of the following is to rigorously construct a new permutation $g_\pi(\tau)$, the permutation graph of which is the graph obtained from $(E_\tau, \mathcal{L}_\tau)$, by permuting the vertices through π while maintaining the remaining structure. Later, it will be shown that when π ranges over the endpoint-fixing permutations in $\mathrm{Perm}(E^{k-1})$, this construction exactly reaches all of the level-k permutations that are shift space equivalent to τ whenever $(E_\tau, \mathcal{L}_\tau)$ is synchronizing in the first label.

Let $\tau \in \mathrm{Perm}(E^k)$ and $\pi \in \mathrm{Perm}(E^{k-1})$ be endpoint-fixing. Given $e \in E^1$ and $\alpha \in E^{k-1}$ with $e\alpha \in E^k$, consider the unique $\beta \in E^{k-1}$ and $f \in E^1$ for which $\tau(e\alpha) = \beta f$. Define $g_\pi(\tau) : E^k \to E^k$ by

$$g_\pi(\tau)(e\pi(\alpha)) = \pi(\beta)f. \tag{5.1}$$

The paths used in this definition are sketched in Figure 6.

Lemma 5.2. *For each endpoint-fixing $\tau \in \mathrm{Perm}(E^k)$ and $\pi \in \mathrm{Perm}(E^{k-1})$, $g_\pi(\tau)$ is also an endpoint-fixing permutation of E^k, and the edges of $(E_{g_\pi(\tau)}, \mathcal{L}_{g_\pi(\tau)})$ are $\mu_\pi : \pi(\beta) \xrightarrow{e,f} \pi(\alpha)$ for each edge $\mu : \beta \xrightarrow{e,f} \alpha$ of $(E_\tau, \mathcal{L}_\tau)$.*

Proof. Consider Figures 1 and 6, and notice that $g_\pi(\tau)$ is an endpoint-fixing permutation since both π and τ are endpoint-fixing. □

In this way, $(E_\tau, \mathcal{L}_\tau)$ and $(E_{g_\pi(\tau)}, \mathcal{L}_{g_\pi(\tau)})$ are constructed to have the same structure, and this immediately yields the following.

https://doi.org/10.1017/S0013091517000189 Published online by Cambridge University Press

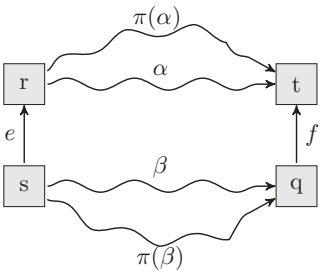


Figure 6. Paths and edges in E used in the construction of $g_\pi(\tau)$ and in the proof of Lemma 5.2. Compare with Figure 1.

Corollary 5.3. *For endpoint-fixing $\tau \in \mathrm{Perm}(E^k)$ and $\pi \in \mathrm{Perm}(E^{k-1})$, $(E_\tau, \mathcal{L}_\tau)$ is synchronizing in the first/second label if and only if $(E_{g_\pi(\tau)}, \mathcal{L}_{g_\pi(\tau)})$ is synchronizing in the first/second label.*

The following proposition shows that the construction given above captures shift space equivalence of permutative endomorphisms which are synchronizing in the first label, and that shift space equivalence of such endomorphisms is the same as inner equivalence through a permutative unitary U_π with $\pi \in \mathrm{Perm}(E^{k-1})$. This relation to inner equivalence has already been considered elsewhere. In particular, [**12**, § 4.2] contains a discussion of the action of such inner automorphisms in the context of Cuntz algebras.

Proposition 5.4. *Let $\tau, \tau' \in \mathrm{Perm}(E^k)$ be endpoint-fixing, and assume that their permutation graphs are synchronizing in the first label. The following are equivalent.*

(1) *τ and τ' are shift space equivalent, i.e. $\mathsf{X}_{(E_\tau, \mathcal{L}_\tau)} = \mathsf{X}_{(E_{\tau'}, \mathcal{L}_{\tau'})}$.*

(2) *There exists $\pi \in \mathrm{Perm}(E^{k-1})$ such that $\tau' = g_\pi(\tau)$.*

(3) *There exists an endpoint-fixing $\pi \in \mathrm{Perm}(E^{k-1})$ such that $\mathrm{Ad}(U_\pi) \circ \lambda_\tau = \lambda_{\tau'}$.*

(4) *The shift space automorphisms coded by the textile systems are equal: $\phi_{T_\tau} = \phi_{T_{\tau'}}$.*

(5) *The shift space automorphisms induced by λ_τ and λ'_τ are equal: $\phi_\tau = \phi_{\tau'}$.*

Proof. (2) ⇒ (1): This follows from Lemma 5.2.

(1) ⇒ (2): By synchronization in the first label and Proposition 3.11, there is an $n \in \mathbb{N}$ such that any path of length n in $(E_\tau, \mathcal{L}_\tau)$ or $(E_{\tau'}, \mathcal{L}_{\tau'})$ is uniquely determined by its label. In particular, its source vertex is determined, whereby each vertex in E^0_τ or $E^0_{\tau'}$ can be identified with the set of labels for length n paths that it emits. Since the languages presented by $(E_\tau, \mathcal{L}_\tau)$ and $(E_{\tau'}, \mathcal{L}_{\tau'})$ are equal by assumption, this induces a bijection π from $E^{k-1} = E^0_\tau$ to $E^{k-1} = E^0_{\tau'}$. By construction, π is endpoint-fixing and $\tau' = g_\pi(\tau)$.

https://doi.org/10.1017/S0013091517000189 Published online by Cambridge University Press

(2) ⇔ (3): Let $e \in E^1$ be given and use Lemma 3.7 to obtain $\alpha_i, \beta_i \in E^{k-1}$ and $f_i \in E^1$ such that $\lambda_\tau(S_e) = \sum_i S_{\beta_i} S_{f_i} S_{\alpha_i}$. Then, using Lemma 5.2,

$$\begin{aligned}(\mathrm{Ad}(U_\pi) \circ \lambda_\tau)(S_e) &= \left(\sum_\alpha S_{\pi(\alpha)} S_\alpha^*\right)\left(\sum_i S_{\beta_i} S_{f_i} S_{\alpha_i}^*\right)\left(\sum_\alpha S_\alpha S_{\pi(\alpha)}^*\right) \\ &= \sum_i S_{\pi(\beta_i)} S_{f_i} S_{\pi(\alpha_i)}^* \\ &= \lambda_{g_\pi(\tau)}(S_e).\end{aligned}$$

(1) ⇔ (4): Let $x \in \mathsf{X}_E$. By definition, $\phi_{T_\tau}(x)$ is the unique $y \in \mathsf{X}_E$ for which $(x_i, y_i)_{i\in\mathbb{Z}} \in \mathsf{X}_{(E_\tau, \mathcal{L}_\tau)}$. Hence, $\phi_{T_\tau} = \phi_{T_{\tau'}}$ if and only if $\mathsf{X}_{(E_\tau, \mathcal{L}_\tau)} = \mathsf{X}_{(E_{\tau'}, \mathcal{L}_{\tau'})}$. Finally, (4) and (5) are equivalent by Corollary 3.17. □

5.2. Vertex ordering and shift space equivalence

Proposition 5.4(2) shows that the shift space equivalence relation is a refinement of the inner equivalence relation, corresponding to inner equivalence through permutative unitaries. Thus, every class at level k has the same size, namely the number of endpoint-fixing permutations at level $(k-1)$. This number grows combinatorially in k, which prompts two questions: how can one recognize when endpoint-fixing permutations $\tau, \tau' \in \mathrm{Perm}(E^k)$ belong to the same shift space equivalence class without testing every endpoint-fixing $\pi \in \mathrm{Perm}(E^{k-1})$; and is it possible to define a succinct representation of the equivalence classes that can be constructed directly, without considering their many individual elements?

Assume that there exists a strict total ordering – i.e. an enumeration – of the vertices in a permutation graph, for which the position in the ordering of any given vertex depends only on the labelled paths emitted and/or received by this vertex. Such an ordering is determined by the language of the permutation graph. Assuming that $\mathsf{X}_{(E_\tau, \mathcal{L}_\tau)} = \mathsf{X}_{(E_{\tau'}, \mathcal{L}_{\tau'})}$, this allows direct construction of π from Proposition 5.4 by pairing the vertices that have the same number in the ordering. In fact, replacing each vertex in a permutation graph with its number in the ordering will be shown to yield a canonical representation for an entire shift space equivalence class. The resulting structure is called an *ordered permutation graph*, and a precise definition will be given below.

From a computational viewpoint, it is greatly desirable if the ordered permutation graphs can be constructed directly, such that the equivalence classes can be studied without first considering the many individual members; this motivates the definition of the concrete ordering introduced below. The aim is to order the vertices of a permutation graph in a way that facilitates recursive construction of collections of shift space equivalence classes using an algorithm structurally similar to the one introduced in § 4. Since different equivalence classes have different languages and hence different orderings, this requires that the ordering can be built incrementally alongside the ordered permutation graph. The concrete ordering given below makes it easy to construct ordered permutation graphs by adding each edge *in order*, building at the same time the ordered permutation graph and its corresponding total ordering. An algorithm carrying out this construction will be detailed in § 6.

(i) The order of minimal emitted sequences.

https://doi.org/10.1017/S0013091517000189 Published online by Cambridge University Press

Begin by ordering the vertices of E, and introduce a strict total order on the edges of E for which $e \leq f$ for $e, f \in E^1$ if $\mathsf{s}_E(e) < \mathsf{s}_E(f)$ or $\mathsf{s}_E(e) = \mathsf{s}_E(f)$ and $\mathsf{r}_E(e) \leq \mathsf{r}_E(f)$. These properties uniquely determine the ordering if and only if E has no parallel edges, but in general the edge ordering involves an arbitrary choice. This ordering on the vertices and edges of E is then used to construct a special family of orderings of the vertices and edges of the permutation graphs E_τ by the following construction.

Definition 5.5 (min. emitted sequence). Let $\tau \in \mathrm{Perm}(E^k)$ be endpoint-fixing. Given $\alpha \in E^0_\tau$, define $M_\tau(\alpha) = e_1\mathsf{r}_E(f_1)e_2\mathsf{r}_E(f_2)\cdots$ to be the lexicographically minimal sequence for which there exists an infinite path

$$A : \alpha \xrightarrow{e_1,f_1} \alpha_1 \xrightarrow{e_2,f_2} \alpha_2 \xrightarrow{e_3,f_3} \cdots$$

over $(E_\tau, \mathcal{L}_\tau)$. Such a path is called a *minimal emitted path*, and the sequence $M_\tau(\alpha)$ is called the *minimal emitted sequence* of α. Lexicographic order of the minimal emitted sequences defines a total preorder on E^0_τ, the *preorder of minimal emitted sequences*, given by $\alpha \leq \beta$ if and only if $M_\tau(\alpha) \leq M_\tau(\beta)$.

This preorder is lifted to permutation graph edges in the obvious way.

Definition 5.6 (edge ordering). Given a labelled edge $\mu : \beta \xrightarrow{e,f} \alpha$ of $(E_\tau, \mathcal{L}_\tau)$, let $M^1_\tau(\mu) = e\mathsf{r}_E(f)M_\tau(\alpha)$. Define a total preorder on the edges E^1_τ by $\mu \leq \nu$ if and only if $M^1_\tau(\mu) \leq M^1_\tau(\nu)$ lexicographically.

Lemma 5.7. *If $(E_\tau, \mathcal{L}_\tau)$ is synchronizing in the first label, then the preorders given in Definitions 5.5 and 5.6 are strict and total orders (i.e. enumerations of the vertices and edges, respectively.)*

Proof. Let $(E_\tau, \mathcal{L}_\tau)$ be synchronizing in the first label. Then there exists $m \in \mathbb{N}$ such that each $\alpha \in E^0_\tau$ is uniquely determined by the first $2m$ elements of $M_\tau(\alpha)$. An analogous argument proves the other statement. □

Notice that, rather than defining one global ordering on E^{k-1} and E^k, each order of minimal emitted sequences orders the permutation graph vertices and edges according to a *particular* permutation τ, yielding different orderings for different τ.

Lemma 5.8. *Let $\tau, \tau' \in Perm(E^k)$ be endpoint-fixing permutations that synchronize in the first label. If $\tau' = g_\pi(\tau)$, then $M_{\tau'} = M_\tau \circ \pi^{-1}$.*

Proof. By Lemma 5.2, there is exactly one labelled edge $\pi(\alpha) \xrightarrow{e,f} \pi(\beta)$ of $(E_{\tau'}, \mathcal{L}_{\tau'})$ for each labelled edge $\alpha \xrightarrow{e,f} \beta$ of $(E_\tau, \mathcal{L}_\tau)$. This lifts to a label-preserving bijection between the paths over $(E_\tau, \mathcal{L}_\tau)$ and the paths over $(E_{\tau'}, \mathcal{L}_{\tau'})$. Since the minimal emitted sequences depend only on path labels, this implies that $M_\tau(\alpha) = M_{\tau'}(\pi(\alpha))$ for every $\alpha \in E^0_\tau$. □

https://doi.org/10.1017/S0013091517000189 Published online by Cambridge University Press

Let $(E_\tau, \mathcal{L}_\tau)$ be a permutation graph of level k that is synchronizing in the first label. For $u, v \in E^0$ let $O_{u\to v} = \{o^i_{u\to v} \mid 0 \le i < |E^{k-1}_{u\to v}|\}$. Define

$$O^0 = \bigcup_{u,v\in E^0} O_{u\to v} \quad \text{and} \quad O^1 = \{\mu_0, \ldots, \mu_{|E^k|-1}\}. \tag{5.2}$$

Since $(E_\tau, \mathcal{L}_\tau)$ is synchronizing in the first label, the orderings of the vertices and edges of E_τ induced by the minimal emitted sequences are strict and total by Lemma 5.7. Hence, it is possible to define bijections $\omega^0_\tau : E^0_\tau \to O^0$ and $\omega^1_\tau : O^1 \to E^1_\tau$ that number the permutation graph vertices and edges according to the ordering. Note that the edges are numbered globally while the vertices are ordered within each class such that the i^{th} element of $E^{k-1}_{u\to v}$ is mapped to $o^i_{u\to v} \in O_{u\to v} \subseteq O^0$.

Definition 5.9 (ordered permutation graph). Given an endpoint-fixing permutation $\tau \in \mathrm{Perm}(E^k)$ that synchronizes in the first label, define its *ordered permutation graph* as the labelled graph $(O_\tau, \mathcal{L}_{O_\tau})$, where

$$O^0_\tau = O^0 \quad \mathsf{s}_{O_\tau} = \omega^0_\tau \circ \mathsf{s}_\tau \circ \omega^1_\tau$$
$$O^1_\tau = O^1 \quad \mathsf{r}_{O_\tau} = \omega^0_\tau \circ \mathsf{r}_\tau \circ \omega^1_\tau,$$

and $\mathcal{L}_{O_\tau} = \mathcal{L}_\tau \circ \omega^1_\tau$.

In other words, $\omega^0_\tau(\alpha) \xrightarrow{e,f} \omega^0_\tau(\beta)$ is a labelled edge in $(O_\tau, \mathcal{L}_{O_\tau})$ precisely when $\alpha \xrightarrow{e,f} \beta$ is a labelled edge in $(E_\tau, \mathcal{L}_\tau)$. In particular, the ordered permutation graph $(O_\tau, \mathcal{L}_{O_\tau})$ presents the same shift space as $(E_\tau, \mathcal{L}_\tau)$. In fact, all mutually shift space equivalent permutations at the same level yield the same ordered permutation graph, as shown in the following proposition.

Proposition 5.10. *If $\tau, \tau' \in \mathrm{Perm}(E^k)$ are endpoint-fixing permutations for which $(E_\tau, \mathcal{L}_\tau)$ and $(E_{\tau'}, \mathcal{L}_{\tau'})$ are synchronizing in the first label, then the following is equivalent to the statements given in Proposition 5.4:*

(1) $(O_\tau, \mathcal{L}_{O_\tau}) = (O_{\tau'}, \mathcal{L}_{O_{\tau'}})$.

Proof. Proposition 5.4(2) ⇒ (1): Assume that $\tau' = g_\pi(\tau)$ for some $\pi \in \mathrm{Perm}(E^{k-1})$. Then by Lemma 5.2, there is precisely one edge $\pi(\alpha) \xrightarrow{e,f} \pi(\beta)$ in $(E_{\tau'}, \mathcal{L}_{\tau'})$ for each edge $\alpha \xrightarrow{e,f} \beta$ of $(E_\tau, \mathcal{L}_\tau)$. By Definition 5.9, the edges of $(O_\tau, \mathcal{L}_{O_\tau})$ and $(O_{\tau'}, \mathcal{L}_{O_{\tau'}})$ are then, respectively, $\omega^0_\tau(\alpha) \xrightarrow{e,f} \omega^0_\tau(\beta)$ and $\omega^0_{\tau'}(\pi(\alpha)) \xrightarrow{e,f} \omega^0_{\tau'}(\pi(\beta))$ for each of the edges $\alpha \xrightarrow{e,f} \beta$ in $(E_\tau, \mathcal{L}_\tau)$. But $\omega^0_{\tau'} \circ \pi = \omega^0_\tau$ by Lemma 5.8 and Definition 5.9, whereby $(O_\tau, \mathcal{L}_{O_\tau}) = (O_{\tau'}, \mathcal{L}_{O_{\tau'}})$.

(6) ⇒ Proposition (5.4)(1): Whenever τ is synchronizing in the first label, ω^1_τ lifts to a label-preserving bijection between the paths over $(E_\tau, \mathcal{L}_\tau)$ and the paths over $(O_\tau, \mathcal{L}_{O_\tau})$, so the language of the shift space presented by a permutation graph is the same as the language of the corresponding ordered permutation graph. The result follows directly. □

Remark 5.11. A construction analogous to the one presented in § 3.6 allows the ordered permutation graph of $\lambda_\tau \circ \lambda_\eta$ to be computed directly from $(O_\tau, \mathcal{L}_{O_\tau})$ and

https://doi.org/10.1017/S0013091517000189 Published online by Cambridge University Press

$(O_\eta, \mathcal{L}_{O_\eta})$. This gives a straightforward way to find $\lambda_\tau \circ \lambda_\eta$ up to adjunction by a permutative unitary when λ_τ and λ_η are known up to adjunction by permutative unitaries. In this setting, the construction is simply a composition of the corresponding textile systems [**28**, p. 18], and this can be calculated as a matrix multiplication in an appropriate semiring. Tools from the theory of textile systems can hence be applied when studying products of permutative automorphisms, and efficient algorithms for this purpose will be examined in a forthcoming paper [**3**]. In this paper, tools for automatic detection of automorphism order are presented as well.

6. Finding outer permutative automorphisms

Because the number of equivalent automorphisms in a shift space equivalence class at level k is the same as the number of all endpoint-fixing permutations at level $(k-1)$, it is infeasible in practice to first construct all automorphisms before picking a canonical one from each class. The great benefit of the ordering introduced in the previous section is that it facilitates not only direct construction of a particular equivalence class, but also direct construction of collections of them.

The following details the recursive construction of all shift space equivalence classes for the permutative automorphism at level k. The structure is similar to that of Equation (4.1), except that a third recursion layer is added to keep track of the numbering within each vertex class $O_{u\to v}$. In addition, edges are now placed in increasing order of minimal emitted sequences, incrementally defining the ordering while the graph is being built. To achieve this, each edge μ added to a partially completed ordered permutation graph $\mathcal{G}$ is required to satisfy an extra ordering condition compared with Definition 4.2.

Definition 6.1. Let $\mathcal{G}$ be a labelled graph with vertex set O^0 and edge set contained in O^1. Let $e : s \to r$ and $f : q \to t$ be edges in E^1 and $\mu : o^i_{s\to q} \xrightarrow{e,f} o^j_{r\to t}$. The predicate $\mathsf{valid}'(\mathcal{G}, \mu)$ is true if

(1) in $\mathcal{G}$, $i \le \min\left\{j \,\middle|\, \mathsf{outdeg}(o^j_{s\to q}) = 0\right\}$, where $\mathsf{outdeg}(v)$ is the outdegree of v,

(2) $\mathcal{G} \oplus \mu$ synchronizes in both labels,

(3) $\mathcal{G} \oplus \mu$ obeys Proposition 4.1 when replacing $E^{k-1}_{u\to v}$ everywhere by $O_{u\to v}$.

Condition (1) is sketched in Figure 7.

In Equation (6.1), which is analogous to Equation (4.1), E^0 is enumerated as $\{v_0, \dots, v_{|E^0|-1}\}$, E^1 as $\{e_0, \dots, e_{|E^1|-1}\}$, and $O_{u\to v}$ is defined as in Equation (5.2). For each $0 \le m \le |E^1| - 1$, let r_m and s_m denote respectively the range and the source of e_m. Define three mutually recursive functions Ψ_1, Ψ_r, and Ψ_s by

$$\begin{aligned} \Psi_1(|E^1|, \mathcal{G}) &= \{\mathcal{G}\} \\ \Psi_1(m, \mathcal{G}) &= \Psi_r(m, 0, \mathcal{G}) \\ \Psi_r(m, |E^0|, \mathcal{G}) &= \Psi_1(m+1, \mathcal{G}) \end{aligned}$$

https://doi.org/10.1017/S0013091517000189 Published online by Cambridge University Press

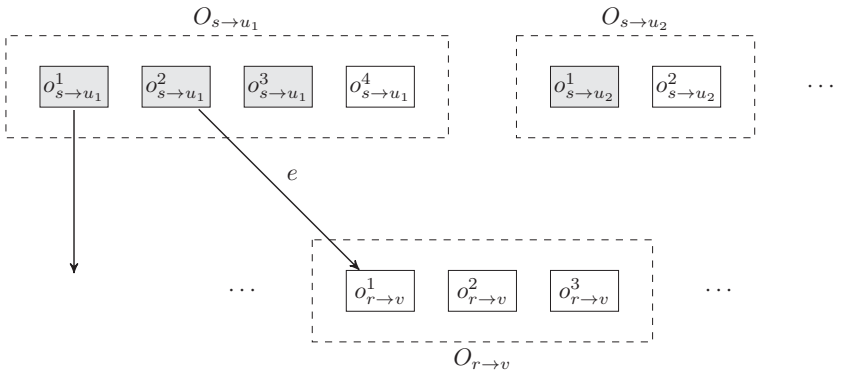


Figure 7. Restriction on the placement of edges caused by Definition 6.1(1). Previously placed edges are as shown. The edge to be placed has first label $e : s \to r$ and range $o^2_{r\to v}$. The legal sources according to Definition 6.1(1) are marked with gray.

$$\begin{aligned}
\Psi_r(m,n,\mathcal{G}) &= \Psi_s(m,n,0,\mathcal{G}) \\
\Psi_s(m,n,|E^{k-1}_{r_m\to v_n}|,\mathcal{G}) &= \Psi_r(m,n+1,\mathcal{G}) \\
\Psi_s(m,n,i,\mathcal{G}) &= \biguplus_{\substack{\mu_j:\beta \xrightarrow{e_m,f} o^i_{r_m\to v_n} \\ \mathsf{valid}'(\mathcal{G},\mu_j)}} \Psi_s(m,n,i+1,\mathcal{G}\oplus\mu_j), \quad j = \left|G^1\right|
\end{aligned} \tag{6.1}$$

The algorithm proceeds as follows. For each $e_m : s_m \to r_m$ in order, together with each range $o^i_{r_m\to v_n} \in O_{r_m\to v_n}$ in order, all possible ways of placing edges of the form $\mu_j : \beta \xrightarrow{e_m,f} o^i_{r_m\to v_n}$ that satisfy Definition 6.1 are tried, taking the disjoint union of the results from the recursion level below. It will be shown in the following that when $\mathcal{G}_0$ is the empty labelled graph with vertex set O^0, $\Psi_1(0,\mathcal{G}_0)$ is exactly the set of ordered permutation graphs of every shift space equivalence class for permutative automorphisms at level k, and that each equivalence class is constructed exactly once.

Lemma 6.2. *Let $(G,\mathcal{L}) \in \Psi_1(0,\mathcal{G}_0)$ be a labelled graph constructed by the algorithm presented above, and let $\mu_i, \mu_{i'} \in G^1$ with labels $[e,f]$ and $[e',f']$, respectively. Then $i < i'$ if and only if one of the following three conditions holds.*

(1) $e < e'$.

(2) $e = e'$ *and* $\mathsf{r}_E(f) < \mathsf{r}_E(f')$.

(3) $e = e'$, $\mathsf{r}_E(f) = \mathsf{r}_E(f')$ *and* $j < j'$ *when* $\mathsf{r}_G(\mu_i) = o^j_{\mathsf{r}_E(e)\to\mathsf{r}_E(f)}$ *and* $\mathsf{r}_G(\mu_{i'}) = o^{j'}_{\mathsf{r}_E(e)\to\mathsf{r}_E(f)}$.

Proof. It is easy to see from the structure of Equation (4.1) that the triplets (m,n,i) are visited in ascending lexicographic order, whereby edges are placed ordered first by first label e_m, then by $\mathsf{r}_E(f) = v_n$, and lastly by the position i within $O_{r_m\to v_n}$. In case

https://doi.org/10.1017/S0013091517000189 Published online by Cambridge University Press

$e\mathsf{r}_E(f) = e'\mathsf{r}_E(f')$, it follows that μ_i was placed before $\mu_{i'}$ if and only if the range of μ_i has a lower number than the range of $\mu_{i'}$ in $O_{\mathsf{r}_E(e)\to\mathsf{r}_E(f)}$. □

For a labelled graph $\mathcal{G} = (G, \mathcal{L})$ constructed by the algorithm presented above, i.e. for $\mathcal{G} \in \Psi_1(0, \mathcal{G}_0)$, the i^{th} edge $\mu_i \in G^1$ is said to be a *first edge* if there is no $j < i$ such that $\mathsf{s}_G(\mu_j) = \mathsf{s}_G(\mu_i)$. That is, μ_i was the first edge with source $\mathsf{s}_G(\mu_i)$ added in the algorithmic construction. There is precisely one first edge associated with each vertex.

Lemma 6.3. *Let $(G, \mathcal{L}) \in \Psi_1(0, \mathcal{G}_0)$, and let $\mu_i, \mu_{i'} \in G^1$ be first edges with labels $[e, f]$ and $[e', f']$, respectively. Assume that $\mathsf{s}_E(e) = \mathsf{s}_E(e')$ and $\mathsf{s}_E(f) = \mathsf{s}_E(f')$. Let $\mathsf{s}_G(\mu_i) = o^j_{\mathsf{s}(e)\to\mathsf{s}(f)}$ and $\mathsf{s}_G(\mu_{i'}) = o^{j'}_{\mathsf{s}(e)\to\mathsf{s}(f)}$. Then $i < i'$ if and only if $j < j'$.*

Proof. This follows immediately from Definition 6.1(1). □

Lemma 6.4. *Let $(G, \mathcal{L}) \in \Psi_1(0, \mathcal{G}_0)$, let $o \in O^0$, and let $M = \mu_{i_1}\mu_{i_2}\cdots$ be the unique path with source o consisting only of first edges. Then M is the unique minimal emitted path of o.*

Proof. Let $o \in O^0$. By Definition 6.1(2), $\mathcal{G}$ is synchronizing in the first label, so the minimal emitted path of o is unique by Lemma 5.7. Let $M' = \mu_{i'_1}\mu_{i'_2}\cdots$ be this minimal emitted path.

Consider $\mu = \mu_{i_1}$ and $\mu' = \mu_{i'_1}$. Let $\mathcal{L}(\mu) = [e, f]$ and let $\mathcal{L}(\mu') = [e', f']$. Since M' is the minimal emitted path of o, $e' \le e$, and μ was the first edge added with source o, Lemma 6.2 implies $e \le e'$ and hence $e = e'$. Similarly, $\mathsf{r}_E(f') \le \mathsf{r}_E(f)$ because M' is a minimal emitted path, and Lemma 6.2 implies $\mathsf{r}_E(f) \le \mathsf{r}_E(f')$, whereby $\mathsf{r}_E(f') = \mathsf{r}_E(f)$.

Furthermore, Lemma 6.2 implies that the number of $\mathsf{r}_G(\mu)$ in $O_{\mathsf{r}(e)\to\mathsf{r}(f)}$ is lower than or equal to the number of $\mathsf{r}_G(\mu')$. Hence, Lemma 6.3 implies that the first edge with source $\mathsf{r}_G(\mu)$ was added before the first edge with source $\mathsf{r}_G(\mu')$, i.e. $i_2 \le i'_2$. Repeated application of this argument proves that for any n, $\mathcal{L}_1(\mu_{i_1}\mu_{i_2}\cdots\mu_{i_n}) = \mathcal{L}_1(\mu_{i'_1}\mu_{i'_2}\cdots\mu_{i'_n})$. Since $\mathcal{G}$ is synchronizing in the first label, this implies that $M = M'$. □

Lemma 6.5. *Let $(G, \mathcal{L}) \in \Psi_1(0, \mathcal{G}_0)$, let $o, o' \in O^0$, and let their minimal emitted paths be $\mu_{i_1}\mu_{i_2}\ldots$ and $\mu_{i'_1}\mu_{i'_2}\ldots$, respectively. Then $o < o'$ in the order of minimal emitted sequences if and only if $i_1 < i'_1$.*

Proof. Assume $o \ne o'$. For each i, let $[e_i, f_i]$ and $[e'_j, f'_j]$ be the labels of μ_{i_j} and $\mu_{i'_j}$, respectively. Note that the edges μ_{i_j} and $\mu_{i'_j}$ are all first edges by Lemma 6.4. By Definition 6.1(2), $\mathcal{G}$ is synchronizing in the first label, so by Lemma 5.7, the two minimal emitted sequences are different. Choose the minimal $l \in \mathbb{N}$ for which $e_l\mathsf{r}_E(f_l) \ne e'_l\mathsf{r}_E(f'_l)$. If $l = 1$, then Lemma 6.2 trivially implies the result, so assume that $l > 1$.

Assume that $i_1 < i'_1$. Since $e_1\mathsf{r}_E(f_1) = e'_1\mathsf{r}_E(f'_1)$, Lemma 6.2 implies that $\mathsf{r}_G(\mu)$ has a strictly lower number in $O_{\mathsf{r}_E(e)\to\mathsf{r}_E(f)}$ than $\mathsf{r}_G(\mu')$. Hence, Lemma 6.3 implies that $i_2 < i'_2$. Repeated application of this argument leads to $i_l < i'_l$. Since $e_l\mathsf{r}_E(f_l) \ne e'_l\mathsf{r}_E(f'_l)$, it follows that Lemma 6.2 yields $e_l\mathsf{r}_E(f_l) < e'_l\mathsf{r}_E(f'_l)$.

For the converse implication, assume that the minimal emitted sequence of o is strictly smaller than the minimal emitted sequence of o'. Then $e_l\mathsf{r}_E(f_l) < e'_l\mathsf{r}_E(f'_l)$, so $i_l < i'_l$ by Lemma 6.2. Since $e_j\mathsf{r}_E(f_j) = e'_j\mathsf{r}_E(f'_j)$ for all $j < l$, Lemmas 6.2 and 6.3 yield $i_1 < i'_1$. □

https://doi.org/10.1017/S0013091517000189 Published online by Cambridge University Press

Corollary 6.6. *Let $(G, \mathcal{L}) \in \Psi_1(0, \mathcal{G}_0)$. The following hold.*

(1) *For $o = o^j_{u\to v}, o' = o^{j'}_{u\to v} \in O_{u\to v}$, $o < o'$ in the order of minimal emitted sequences if and only if $j < j'$.*

(2) *For $\mu_i, \mu'_i \in G^1$, $\mu_i < \mu_{i'}$ in the order of minimal emitted sequences if and only if $i < i'$.*

Proof.

(1) Let μ_i and $\mu_{i'}$ be the first edges emitted by o and o', respectively. By Lemma 6.4 and Lemma 6.5, $o < o'$ if and only if $i < i'$. Since μ_i and $\mu_{i'}$ are first edges, the result then follows from Lemma 6.3.

(2) This follows from Lemma 6.2 and the previous statement. □

Theorem 6.7. *Let $\mathcal{G}_0$ be the labelled graph with vertex set O^0 and no edges, then $\Psi_1(0, \mathcal{G}_0)$ is the set of ordered permutation graphs corresponding to the shift space equivalence classes for every permutative automorphism at level k. Furthermore, each ordered permutation graph is constructed only once.*

Proof. Let $\mathcal{G} \in \Psi_1(0, \mathcal{G}_0)$ be a labelled graph constructed by the algorithm. The aim is to show that $\mathcal{G}$ is an ordered permutation graph. By an argument analogous to the one used in the proof of Theorem 4.4, Definition 6.1(3) implies that any assignment of the elements of $E^{k-1}_{u\to v}$ to $O_{u\to v}$ will map $\mathcal{G}$ to a permutation graph. Furthermore, by Definition 6.1(2), the resulting permutation graph is synchronizing in both labels and hence corresponds to an automorphism. By Corollary 6.6, the vertices and edges of $\mathcal{G}$ are ordered correctly, and consequently $\mathcal{G}$ is also an ordered permutation graph.

Next, let $\mathcal{G} = (G, \mathcal{L})$ be an ordered permutation graph, corresponding to the shift space equivalence class for a permutative automorphism. The aim is to show that $\mathcal{G} \in \Psi_1(0, \mathcal{G}_0)$. Given $0 \le j \le |G^1|$, let $\mathcal{G}_j$ denote the subgraph of $\mathcal{G}$ obtained by including precisely the first j edges of G^1 in the order of minimal emitted sequences. Note that for $j > 0$, the j edges in $\mathcal{G}_j$ are $\mu_0, \ldots, \mu_{j-1}$. The following proves by induction that for each j, the subgraph $\mathcal{G}_j$ is constructed by an intermediate step of the algorithm. Note that the algorithm starts from $\mathcal{G}_0$.

For the induction step, assume that $\mathcal{G}_j$ has been constructed at an intermediate step. Let $\mu_j \in G^1$ with $\mu_j : o^l_{s\to u} \xrightarrow{e_m, f} o^i_{r\to v_n}$. Consider another edge $\mu_{j'} \in G^1$ for which $\mu_{j'} : o^{l'}_{s'\to u'} \xrightarrow{e_{m'}, f'} o^{i'}_{r'\to v_{n'}}$. Since $\mathcal{G}$ is an ordered permutation graph, $j' < j$ if and only if $(n', m', i') < (n, m, i)$ lexicographically. Specifically, the edges in $\mathcal{G}_j$ are precisely the $\mu_{j'} \in G^1$ for which $(n', m', i') < (n, m, i)$. Hence, the next step in the algorithm, following the construction of $\mathcal{G}_j$ will add an edge with the same first label and range as μ_j. The specific edge μ_j can be added by the algorithm if and only if $(\mathcal{G}_j, \mu_j)$ satisfies Definition 6.1. Since $\mathcal{G}$ is an ordered permutation graph, $(\mathcal{G}_j, \mu_j)$ satisfies Definition 6.1(2) and (3).

It remains to be shown that $(\mathcal{G}_j, \mu_j)$ satisfies Definition 6.1(1), i.e. that $o^l_{s\to u}$ is a valid source in the construction. Assume there exists a $o^{l'}_{s\to u}$ with $l' < l$ (the condition is trivially satisfied if no such l' exists). $\mathcal{G}$ is an ordered permutation graph, so $o^{l'}_{s\to u} < o^l_{s\to u}$ in the order of minimal emitted sequences. Hence, there is an edge $\mu_{j'} \in G^1$ with source

https://doi.org/10.1017/S0013091517000189 Published online by Cambridge University Press

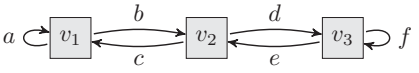


Figure 8. The graph E considered in § 7.

$o^{l'}_{s\to u}$ and $j' < j$, and $o^{l'}_{s\to u}$ emits at least one edge in $\mathcal{G}_j$. Consequently, $(\mathcal{G}_j, \mu_j)$ satisfies Definition 6.1(1), whereby $\mathcal{G}_j \oplus \mu_j = \mathcal{G}_{j+1}$ can be constructed by the next step of the algorithm. Because $\mathcal{G} = \mathcal{G}_{|G^1|}$, the procedure terminates, and the desired result follows.

Finally, let $\mathcal{G}$ be an ordered permutation graph. To see that the algorithm does not construct $\mathcal{G}$ twice, note that it places edges in ascending order. Given a subgraph $\mathcal{G}_j$ as above, there is precisely one way to place the next edge to obtain $\mathcal{G}_{j+1}$. □

7. Example: constructing automorphisms

Our study of permutative automorphisms of graph algebras is partially motivated by the desire to investigate their crossed products. In particular, composition of a suitable permutative automorphism with a graph automorphism may result in an outer action of the integer group on the corresponding graph algebra, whose crossed product is a Kirchberg algebra. Since graph automorphisms may act non-trivially on the K-theory, the resulting crossed product may have torsion in its K_1-group. Thus, actions of this type lead to very explicit and accessible models of very general Kirchberg algebras. We intend to employ these models to structural investigations of these algebras. As this work falls far outside the scope of the present paper, the results will be reported elsewhere. However, the following will show how the methods developed here can be used to detect automorphisms that are relevant for such an investigation.

Consider the graph E shown in Figure 8. The only non-trivial graph automorphism of E acts non-trivially on the K_1-group of the associated graph C^*-algebra, so one may hope to construct a crossed product of $C^*(E)$ as outlined above. In this section, the techniques developed in the previous sections will be used to investigate the permutative endomorphisms and automorphisms of $C^*(E)$. For each k, there are clearly $3 \cdot 2^k$ paths of length k in E. For small values of k, Table 1 summarizes the number of paths, permutative endomorphisms, permutative automorphisms, and classes of permutative automorphisms equivalent up to adjunction by permutative unitaries as described in Proposition 5.10. The numbers in the first three columns are easily computed, while the last two columns require finding all the automorphisms. Notice the rapid combinatorial growth with k, and note also that the quotient of the numbers in columns 4 and 5 is the number of permutative endomorphisms of the previous level, as can be understood from Proposition 5.4.

At levels $k = 1$ and $k = 2$, it is possible to construct the ordered permutation graphs by hand using the algorithms and tools given above. At level $k = 3$, there are too many permutative endomorphisms to check by hand, but it would be possible, albeit time consuming, to test all the permutative endomorphisms individually using a direct brute-force computer program. Using the algorithm from § 6, the search is completed instantaneously. At level $k = 4$, there are so many endpoint-fixing permutations that the problem cannot be handled by brute force. Even assuming that the automorphism condition can be tested in a microsecond, considering all permutative endomorphisms individually would take 35 million years. However, combining the methods presented in this paper, it is possible to

https://doi.org/10.1017/S0013091517000189 Published online by Cambridge University Press

Table 1. *Number of paths of length k, permutative endomorphisms, permutative automorphisms, and order classes of permutative automorphisms for small values of k.*

k	Paths	Endomorphisms	Automorphisms	Classes
1	6	1	1	1
2	12	$(2! \cdot 1! \cdot 1!)^3 = 8$	2	2
3	24	$(2! \cdot 3! \cdot 3!)^3 = 373,248$	32	4
4	48	$(5! \cdot 5! \cdot 6!)^3 \approx 1.1 \cdot 10^{21}$	454,989,312	1219
5	96	$(10! \cdot 11! \cdot 11!)^3 \approx 1.9 \cdot 10^{65}$	?	?

reduce the problem to a tractable one. Constructing representations of the equivalence classes directly using the methods of § 5 reduces the number of graphs to consider by a factor 373,248 from $1.1 \cdot 10^{21}$ to $3.0 \cdot 10^{15}$, so – under the same assumption as above – investigating each class would take about 100 years. However, the pruning of the search tree built into the algorithm presented in § 6 means that we are able to complete the exhaustive search for automorphism classes in a few minutes, and this investigation reveals that there are 1219 distinct equivalence classes. There is, however, no way to avoid the combinatorial growth of the problem. At level $k = 5$, the number of permutative endomorphism classes is $6 \cdot 10^{28}$ times greater than at level $k = 4$; hence, we expect the number of automorphism classes to be so large that it is impossible to generate all of them. For level $k = 5$ and beyond, we are thus forced to restrict our attention to interesting subsets. The following sections give detailed discussions of the automorphisms found in the levels $k = 1$ through $k = 5$.

Levels $k = 1$ and $k = 2$

Clearly, the identity is the only endpoint-fixing permutation at level $k = 1$, and it is straightforward to check by hand that it induces an automorphism. At level $k = 2$, there are precisely two permutative automorphisms. They are given by the following two endpoint-fixing permutations of E^2:

$$\tau_{2,0} = \mathrm{Id}, \quad \tau_{2,1} = (de, cb).$$

Level $k = 3$

At this level, there are four different equivalence classes of permutative automorphisms. The following endpoint-fixing permutations of E^3 are representatives of these classes:

$$\begin{aligned}
\tau_{3,0} &= \mathrm{Id} \\
\tau_{3,1} &= (dfe, cab) \\
\tau_{3,2} &= (bde, bcb)(ede, ecb) \\
\tau_{3,3} &= (bde, bcb)(dfe, cab)(ede, ecb)
\end{aligned}$$

Each class contains eight permutative automorphisms, since every endpoint-fixing permutation from the previous level permutes the labels of the permutation graph, resulting in a presentation of a permutative automorphism that, as described in Proposition 5.10,

https://doi.org/10.1017/S0013091517000189 Published online by Cambridge University Press

is inner equivalent to the original one. Note that $\lambda_{\tau_{3,1}}$ is inner equivalent to $\lambda_{\tau_{2,1}}$, while $\lambda_{\tau_{3,2}}$ and $\lambda_{\tau 3,3}$ do not occur at lower levels. In general, each permutative automorphism at level k gives rise to an inner equivalent permutative automorphism at level $k+1$.

Level $k = 4$

At this level, there are 1219 classes of permutative automorphisms, which is too many to list. As an example, consider the permutation

$$\tau_4 = (abde, bdfe, bcab)(ecab, edfe).$$

This gives a permutative automorphism that has infinite order, unlike the permutative automorphisms at levels 1 through 3.

Level $k = 5$

As mentioned above, it is not feasible to find all permutative automorphisms at level $k = 5$ using the methods developed in this paper. However, by restricting to automorphisms with certain properties, it is possible to find interesting automorphisms. Define

$$T_1 = S_a + S_b + S_d \quad \text{and} \quad T_2 = S_c + S_e + S_f.$$

Then it is straightforward to check that T_1, T_2 generate a unital copy of $\mathcal{O}_2$ inside $C^*(E)$. The aim of the following will be to identify permutative automorphisms of $C^*(E)$ that fix T_2, and that send T_1 to a sum of words in T_1, T_2, T_1^*, T_2^*. Such a permutative automorphism of $C^*(E)$ will naturally induce an automorphism of $\mathcal{O}_2$. However, even with this restriction, there are too many possibilities to check by brute force. To counter this, we restrict to permutations that fix any path (of length 5) starting with c, e or f (i.e. the induced automorphism of $\mathcal{O}_2$ fixes all paths starting with the second edge). This restricts the problem to an investigation of $5! \cdot 10! = 435,456,000$ permutations, a number that is sufficiently small to be handled by brute force. An exhaustive search yields 12 distinct $\mathcal{O}_2$-preserving permutative automorphisms that fix all paths starting with c, e or f. As an example, two of these are:

$$\tau_5 = (abcbc, abdec, bdfec)(abdfe, bdffe, bdede)(abdff, bdfff, bdedf)$$

$$\tau_5' = (aaabc, abdec, abcbc, bdfec)(aabde, abdfe, bdede, bdffe)(aabdf, abdff, bdedf, bdfff).$$

ϕ_{τ_5} can be shown to be of infinite order, while $\phi_{\tau_5'}$ has order 60. General tools for examining the order of permutative automorphisms will be described in the forthcoming paper mentioned in Remark 5.11 [**3**].

8. Example: automorphisms of $\mathcal{O}_n$

Permutative automorphisms of $\mathcal{O}_n$ have already been investigated experimentally in [**8**, **12**], and it is straightforward to use the algorithms presented in §§ 4 and 6 to verify the numbers of permutative automorphisms of $\mathcal{O}_n$ and classes of such automorphisms found in these two papers. It is worth noting that the algorithms presented in the present paper yield significant improvements over the specialized approaches used in the

https://doi.org/10.1017/S0013091517000189 Published online by Cambridge University Press

Table 2. *Numbers of order classes of permutative automorphisms of $\mathcal{O}_n$ at various levels. Some of these results were first discovered in [8, 12]; new results are emphasized in bold.*

	$k=2$	$k=3$	$k=4$	$k=5$
$\mathcal{O}_2$	$1 \cdot 2!$	$1 \cdot 2!$	$7 \cdot 2!$	$\mathbf{419 \cdot 2!}$
$\mathcal{O}_3$	$16 \cdot 3!$	$\mathbf{254, 104 \cdot 3!}$	–	–
$\mathcal{O}_4$	$10,020 \cdot 4!$	–	–	–
$\mathcal{O}_5$	$\mathbf{245, 046, 336 \cdot 5!}$	–	–	–

previous investigations. First, the methods presented here can be applied to a wide range of interesting graph algebras beyond the Cuntz algebras. Second, they are much faster than the previously used methods. For instance, the original identification of the automorphism classes at level 2 in $\mathcal{O}_4$ took approximately 70 days of computation on a server [12], while our methods were able to deliver the same result after two seconds of computation, an improvement by a factor of more than two million. The following sections exemplify how the approach presented in this paper facilitates the study of permutative automorphisms.

8.1. Cataloguing automorphisms

Thanks to the drastic improvement in the speed of the computations achieved through the techniques developed in this paper, it has been possible to extend the investigation of the Cuntz algebra automorphisms from [8, 12] to larger values of n and higher levels, in spite of the violent combinatorial growth of the problem. The numbers of classes of permutative automorphisms of $\mathcal{O}_n$ at various levels are shown in Table 2. Previously unknown results were obtained for $\mathcal{O}_2$ level 5, $\mathcal{O}_3$ level 3, and $\mathcal{O}_5$ level 2. These are emphasized in bold in the table.

The efficient algorithms developed in this paper allowed the previously known results to be reproduced using a few seconds of computation, while the new results required substantial computing power. Cataloguing the automorphisms of $\mathcal{O}_5$ level 2 required approximately 6 CPU-weeks of computation, so this result can now be obtained using a desktop computer, while the identification of all automorphisms of $\mathcal{O}_3$ level 3 and $\mathcal{O}_2$ level 5 required massive parallel computations. In the case of $\mathcal{O}_3$ level 3, the result was obtained after 10 CPU-years of computation, which was completed in about 2 weeks on 250 CPUs. In the case of $\mathcal{O}_2$ level 5, the computation requires approximately 100 CPU-years. This computation has been carried out on a variable number of CPUs ranging from 100 to 400 over 6 months. The full database of permutative automorphism classes will be published elsewhere, together with the programs used to generate and analyse the automorphisms.

8.2. Subgroups

The shift space automorphisms induced by the outer automorphisms at a given level do not usually form a group, since the composition of two permutative automorphisms generally results in a permutative automorphism belonging to a higher level. However, some products do remain within the original level and, for each level, it is possible to

https://doi.org/10.1017/S0013091517000189 Published online by Cambridge University Press

Table 3. *Maximal shift space automorphism groups at level 5 in* $\mathcal{O}_2$.

Order	Group	Multiplicity
12	D_{12}	4
2	C_2	9

Table 4. *Maximal shift space automorphism groups at level 3 in* $\mathcal{O}_3$.

Order	Group	Multiplicity
256	$C_2 \times C_2 \times C_2 \times C_2 \times C_2 \times C_2 \times C_2 \times C_2$	3
128	$C_2 \times C_2 \times C_2 \times C_2 \times D_8$	6
96	$C_2 \times C_2 \times S_4$	6
64	$C_2 \times C_2 \times C_2 \times D_8$	6
12	D_{12}	24
8	D_8	6
6	S_3	10
4	$C_2 \times C_2$	24

identify the groups of shift space automorphisms that are fully contained in it. Similarly, one can find the subgroups contained in any given finite set of shift space automorphisms, for example, products up to a particular length of elements taken from the set of shift space automorphisms at a given level.

Such an investigation has been carried out, using programs developed for this purpose to compute all maximal subgroups and their group tables, and using `GAP`[**17**] to identify the groups that were found. Tables 3–5 list the maximal groups contained in the $419 \cdot 2! = 838$ shift space automorphisms at level 5 in $\mathcal{O}_2$, the $254,104 \cdot 3! = 1,524,624$ shift space automorphisms at level 3 in $\mathcal{O}_3$, and the $10,020 \cdot 4! = 240,480$ shift space automorphisms at level 2 in $\mathcal{O}_4$, respectively. The listed groups are maximal in the sense that they are not subgroups of any larger groups contained in the given set of shift space automorphisms.

Table 5. *Maximal shift space automorphism groups at level 2 in* $\mathcal{O}_4$.

Order	Group	Multiplicity
128	$(C_2 \times C_2 \times C_2 \times C_2 \times C_2 \times C_2) \rtimes C_2$	3
96	$C_2 \times C_2 \times S_4$	6
64	$C_2 \times C_2 \times C_2 \times D_8$	12
64	$C_2 \times ((C_2 \times C_2 \times C_2 \times C_2) \rtimes C_2)$	6
54	$(C_3 \times C_3 \times C_3) \rtimes C_2$	4
36	$S_3 \times S_3$	8
24	S_4	7

https://doi.org/10.1017/S0013091517000189 Published online by Cambridge University Press

Table 6. *Permutations that induce distinct outer permutative automorphisms at level 2 in $\mathcal{O}_3$, and the orders of the 96 shift space automorphisms. The first column lists the 16 permutations that induce distinct automorphisms of $\mathcal{O}_3$ up to inner equivalence and graph automorphism. Each row lists the orders of the 6 shift space automorphisms obtained by composing ϕ_τ with the shift space automorphisms induced by graph automorphisms, i.e., the 6 permutations of the edges in the graph for $\mathcal{O}_3$. The orders are computed using the techniques discussed in Remark 5.11.*

Permutation τ	Orders of induced automorphisms					
	ϕ_τ	$\phi_{(a,b)}\phi_\tau$	$\phi_{(a,c)}\phi_\tau$	$\phi_{(b,c)}\phi_\tau$	$\phi_{(a,b,c)}\phi_\tau$	$\phi_{(a,c,b)}\phi_\tau$
Id	1	2	2	2	3	3
(ca, cb)	2	2	∞	∞	∞	∞
$(bb, bc)(cb, cc)$	2	∞	∞	2	∞	∞
$(bb, cb, ca)(bc, cc)$	∞	∞	∞	∞	∞	∞
$(bb, cb)(bc, cc)$	2	∞	∞	2	∞	∞
$(bb, cc)(bc, cb)$	2	∞	∞	2	∞	∞
(ba, bc)	2	∞	2	∞	∞	∞
$(ba, cc, bc)(bb, cb)$	∞	∞	∞	∞	∞	∞
(ac, bc, ba)	∞	∞	∞	∞	∞	∞
$(ac, cc, bc, ca, ba)(bb, cb)$	∞	∞	∞	∞	∞	∞
$(ac, bc, bb)(cb, cc)$	∞	∞	∞	∞	∞	∞
$(ac, cc, bb)(ba, ca)(bc, cb)$	∞	∞	∞	∞	∞	∞
(ac, bc)	2	2	∞	∞	∞	∞
$(ac, bc)(ca, cb)$	2	2	∞	∞	∞	∞
$(ac, cc, bc)(ba, ca)(bb, cb)$	∞	∞	∞	2	∞	2
$(ac, cc, bc)(ba, ca, bb, cb)$	∞	∞	∞	2	∞	2

8.3. Orders

As an example of the results achieved in this way, consider the permutative automorphisms at level 2 in $\mathcal{O}_3$. In this case, there are 96 outer permutative automorphisms, but the entire collection can be reconstructed from the 16 permutations listed in Table 6 by composing with the graph automorphisms of the original graph, i.e. the $3! = 6$ permutations of the three edges. As in the previous example, the techniques mentioned in Remark 5.11 were used to investigate the orders of these permutative automorphisms, and the results of this investigation are listed in Table 6. The exhaustive search for outer permutative automorphisms at level 2 in $\mathcal{O}_3$ required 2ms of computation time, and the automorphism orders took 60ms in total to calculate.

8.4. Structural classification

For $\mathcal{O}_n$ and $1 \leq i \leq n$, the *i-subgraph* of an (ordered) permutation graph over $\mathcal{O}_n$ will denote the subgraph induced by all edges with first label i. In [**12**], it was shown (up to the reformulation mentioned in Remark 3.9) that each i-subgraph is a directed tree with a loop attached to the root. Since automorphisms with the same ordered permutation graphs must have the same i-trees, the structures of these trees are invariant under

https://doi.org/10.1017/S0013091517000189 Published online by Cambridge University Press

Table 7. *Number of ordered permutation graphs at level* 3 *in* $\mathcal{O}_3$ *subdivided by ID type triples.*

ID type triple	# outer automorphisms
A A A	$12,748 \cdot 3!$
A B B	$124,740 \cdot 3!$
A C D	$19,440 \cdot 3!$
A E E	$86,412 \cdot 3!$
A F G	$5676 \cdot 3!$
A H H	$5088 \cdot 3!$

shift space equivalence. In [**8**, Table 1], the possible trees occurring in permutative automorphisms at level 3 in $\mathcal{O}_3$ were given ID types $A, \ldots, H$ based on the multiplicities of indegrees. This was then used to classify permutative endomorphisms by the unordered triple consisting of the ID types of the 1-, 2-, and 3-subgraphs. As discussed in the previous section and seen in Table 2, the permutative automorphisms at level 3 in $\mathcal{O}_3$ have not previously been constructed. Hence, the classification given in [**8**] is incomplete as it was not possible at the time to distinguish automorphisms from endomorphisms which are only surjective on the diagonal. Having computed all the possible ordered permutation graphs of $\mathcal{O}_3$ at level 3, it is straightforward for us to complete this classification. The results are given in Table 7.

Acknowledgements. This work was supported by VILLUM FONDEN through the experimental mathematics network at the University of Copenhagen, by the Danish National Research Foundation through the Centre for Symmetry and Deformation (DNRF92), and by the Danish Natural Science Research Council (FNU). J.A. was partially supported by the FNU Sapere Aude grant 'Complexity via Logic and Algebra' (COLA). The authors are grateful to Mike Boyle and Brendan Berg for valuable comments and suggestions concerning the theory of textile systems. The parallel calculations were performed in part on the University of Copenhagen High Performance Computing facilities, Denmark, and in part on the HPC cluster at New Zealand Institute for Advanced Study, Massey University, New Zealand.

References

1. G. Abrams and G. Aranda Pino, The Leavitt path algebra of a graph, *J. Algebra* **293**(2) (2005), 319–334.
2. P. Ara, M. A. Moreno and E. Pardo, Nonstable K-theory for graph algebras, *Algebr. Represent. Theory* **10**(2) (2007), 157–178.
3. J. E. Avery and R. Johansen, Efficient computations with automorphisms of shift spaces, in preparation.
4. J. E. Avery, R. Johansen and J. G. Simonsen, Invertibility of one-block codes, in preparation.
5. O. Bratteli and P. E. T. Jorgensen, Iterated function systems and permutation representations of the Cuntz algebra, *Mem. Amer. Math. Soc.* **139**(663) (1999), x+89.
6. R. Conti, J. H. Hong and W. Szymański, Endomorphisms of graph algebras, *J. Funct. Anal.* **263**(9) (2012), 2529–2554.

https://doi.org/10.1017/S0013091517000189 Published online by Cambridge University Press

7. R. Conti, J. H. Hong and W. Szymański, The restricted Weyl group of the Cuntz algebra and shift endomorphisms, *J. Reine Angew. Math.* **667** (2012), 177–191.
8. R. Conti, J. Kimberley and W. Szymański, More localized automorphisms of the Cuntz algebras, *Proc. Edinb. Math. Soc. (2)* **53**(3) (2010), 619–631.
9. R. Conti and C. Pinzari, Remarks on the index of endomorphisms of Cuntz algebras, *J. Funct. Anal.* **142**(2) (1996), 369–405.
10. R. Conti, M. Rørdam and W. Szymański, Endomorphisms of $\mathcal{O}_n$ which preserve the canonical UHF-subalgebra, *J. Funct. Anal.* **259**(3) (2010), 602–617.
11. R. Conti and W. Szymański, Computing the Jones index of quadratic permutation endomorphisms of $\mathcal{O}_2$, *J. Math. Phys.* **50**(1) (2009), 012705.
12. R. Conti and W. Szymański, Labeled trees and localized automorphisms of the Cuntz algebras, *Trans. Amer. Math. Soc.* **363**(11) (2011), 5847–5870.
13. R. Conti and W. Szymański, Automorphisms of the Cuntz algebras, in *Progress in operator algebras, noncommutative geometry, and their applications*, Proceedings of the 4th Annual Meeting of the European Noncommutative Geometry Network, Bucharest, 2011, pp. 1–15 (Theta, Bucharest, 2012).
14. J. Cuntz, Automorphisms of certain simple C^*-algebras, in *Quantum fields—algebras, processes* Proceedings of the Symposium Bielefeld Encounters in Physics and Mathematics, Volume II, pp. 187–196 (Springer, Vienna, 1980).
15. J. Cuntz and W. Krieger, A class of C^*-algebras and topological Markov chains, *Invent. Math.* **56**(3) (1980), 251–268.
16. N. J. Fowler, M. Laca and I. Raeburn, The C^*-algebras of infinite graphs, *Proc. Amer. Math. Soc.* **128**(8) (2000), 2319–2327.
17. The GAP Group, *GAP – Groups, Algorithms, and Programming, Version 4.7.7.* (2015).
18. T. Hayashi, On normalizers of C^*-subalgebras in the Cuntz algebra $\mathcal{O}_n$, *J. Operator Theory* **69**(2) (2013), 525–533.
19. A. Hopenwasser, J. R. Peters and S. C. Power, Subalgebras of graph C^*-algebras, *New York J. Math.* **11** (2005), 351–386 (electronic).
20. M. Izumi, Subalgebras of infinite C^*-algebras with finite Watatani indices. I. Cuntz algebras, *Comm. Math. Phys.* **155**(1) (1993), 157–182.
21. M. Izumi, Subalgebras of infinite C^*-algebras with finite Watatani indices. II. Cuntz-Krieger algebras, *Duke Math. J.* **91**(3) (1998), 409–461.
22. K. Kawamura, Polynomial endomorphisms of the Cuntz algebras arising from permutations. I. General theory, *Lett. Math. Phys.* **71**(2) (2005), 149–158.
23. A. Kumjian, D. Pask and I. Raeburn, Cuntz-Krieger algebras of directed graphs, *Pacific J. Math.* **184**(1) (1998), 161–174.
24. A. Kumjian, D. Pask, I. Raeburn and J. Renault, Graphs, groupoids, and Cuntz-Krieger algebras, *J. Funct. Anal.* **144**(2) (1997), 505–541.
25. D. Lind and B. Marcus, *An introduction to symbolic dynamics and coding* (Cambridge University Press, Cambridge, 1995).
26. R. Longo, A duality for Hopf algebras and for subfactors. I, *Comm. Math. Phys.* **159**(1) (1994), 133–150.
27. G. Nagy and S. Reznikoff, Abelian core of graph algebras, *J. Lond. Math. Soc. (2)* **85**(3) (2012), 889–908.
28. M. Nasu, Textile systems for endomorphisms and automorphisms of the shift, *Mem. Amer. Math. Soc.* **114**(546) (1995), viii+215.
29. I. Raeburn, *Graph algebras*, CBMS Regional Conference Series in Mathematics, Volume 103 (CBMS, Washington, DC, 2005).
30. A. Skalski, Noncommutative topological entropy of endomorphisms of Cuntz algebras II, *Publ. Res. Inst. Math. Sci.* **47**(4) (2011), 887–896.

https://doi.org/10.1017/S0013091517000189 Published online by Cambridge University Press

31. A. Skalski and J. Zacharias, Noncommutative topological entropy of endomorphisms of Cuntz algebras, *Lett. Math. Phys.* **86**(2–3) (2008), 115–134.
32. M. Tomforde, Leavitt path algebras with coefficients in a commutative ring, *J. Pure Appl. Algebra* **215**(4) (2011), 471–484.
33. S. B. G. Webster, The path space of a directed graph, *Proc. Amer. Math. Soc.* **142**(1) (2014), 213–225.

https://doi.org/10.1017/S0013091517000189 Published online by Cambridge University Press