Moves on k-graphs preserving Morita equivalence

Elizabeth Gillaspy
University of Montana

joint with C. Eckhardt, K. Fieldhouse, D. Gent, I. Gonzales, and D. Pask

BIRS: Topology and Measure in Dynamics and Operator Algebras
10 September 2019

Theorem (Eilers-Restorff-Ruiz-Sørensen [ERRS16])

Let E, F be directed graphs with finitely many vertices. $C^*(E)$ and $C^*(F)$ are stably equivalent if and only if one can convert E into F by a finite sequence of the moves

(S), (O), (I), (R), (C), (P)

and their inverses.

Background: Geometric classification of graph C^*-algebras

Theorem (Eilers-Restorff-Ruiz-Sørensen [ERRS16])

Let E, F be directed graphs with finitely many vertices. $C^*(E)$ and $C^*(F)$ are stably equivalent if and only if one can convert E into F by a finite sequence of the moves

$$(S),\ (O),\ (I),\ (R),\ (C),\ (P)$$

and their inverses.

Built on work of Bates-Pask [BP04], Drinen (thesis) and Crisp-Gow [CG06] on moves for graph algebras; Rørdam [Rø95] on classification of Cuntz-Krieger algebras; Boyle-Huang [BH] from dynamical systems.

Background: Geometric classification of graph C^*-algebras

Theorem (Eilers-Restorff-Ruiz-Sørensen [ERRS16])

Let E, F be directed graphs with finitely many vertices. $C^*(E)$ and $C^*(F)$ are stably equivalent if and only if one can convert E into F by a finite sequence of the moves

(S), (O), (I), (R), (C), (P)

and their inverses.

Built on work of Bates-Pask [BP04], Drinen (thesis) and Crisp-Gow [CG06] on moves for graph algebras; Rørdam [Rø95] on classification of Cuntz-Krieger algebras; Boyle-Huang [BH] from dynamical systems.

ERRS also obtained a classification, up to isomorphism, of such graph C^*-algebras by using ordered filtered K-theory.

Background: Geometric classification of graph C^*-algebras

Theorem (Eilers-Restorff-Ruiz-Sørensen [ERRS16])

Let E, F be directed graphs with finitely many vertices. $C^*(E)$ and $C^*(F)$ are stably equivalent if and only if one can convert E into F by a finite sequence of the moves

(S), (O), (I), (R), (C), (P)

and their inverses.

Built on work of Bates-Pask [BP04], Drinen (thesis) and Crisp-Gow [CG06] on moves for graph algebras; Rørdam [Rø95] on classification of Cuntz-Krieger algebras; Boyle-Huang [BH] from dynamical systems.

ERRS also obtained a classification, up to isomorphism, of such graph C^*-algebras by using ordered filtered K-theory.

Our work constitutes a first step in developing such classification results for higher-rank graphs.

Moves for k-graphs

So far, we have identified four types of moves on k-graphs which preserve Morita equivalence:

- Sink deletion

Moves for k-graphs

So far, we have identified four types of moves on k-graphs which preserve Morita equivalence:

- Sink deletion (S)

Moves for k-graphs

So far, we have identified four types of moves on k-graphs which preserve Morita equivalence:

- Sink deletion (S)
- Insplitting

Moves for k-graphs

So far, we have identified four types of moves on k-graphs which preserve Morita equivalence:

- Sink deletion (S)
- Insplitting (O)

Moves for k-graphs

So far, we have identified four types of moves on k-graphs which preserve Morita equivalence:

- Sink deletion (S)
- Insplitting (O)
- Reduction

Moves for k-graphs

So far, we have identified four types of moves on k-graphs which preserve Morita equivalence:

- Sink deletion (S)
- Insplitting (O)
- Reduction (R)

Moves for k-graphs

So far, we have identified four types of moves on k-graphs which preserve Morita equivalence:

- Sink deletion (S)
- Insplitting (O)
- Reduction (R)
- Delay

Moves for k-graphs

So far, we have identified four types of moves on k-graphs which preserve Morita equivalence:

- Sink deletion (S)
- Insplitting (O)
- Reduction (R)
- Delay

Delay for directed graphs was considered by Drinen and Bates-Pask; related to flow equivalence for shifts of finite type. (Parry-Sullivan)

Moves for k-graphs

So far, we have identified four types of moves on k-graphs which preserve Morita equivalence:

- Sink deletion (S)
- Insplitting (O)
- Reduction (R)
- Delay

Delay for directed graphs was considered by Drinen and Bates-Pask; related to flow equivalence for shifts of finite type. (Parry-Sullivan)

For directed graphs, reduction and delay are (intuitively but not exactly) inverses.

Higher-rank graphs

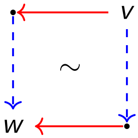
A k-graph Λ consists of:

- An edge-colored directed graph, with k colors of edges;

Higher-rank graphs

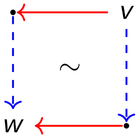
A k-graph Λ consists of:

- An edge-colored directed graph, with k colors of edges; plus
- A factorization rule: for any two colors (red, blue), any red-blue path from v to w is equivalent to a unique blue-red path.



A k-graph Λ consists of:

- An edge-colored directed graph, with k colors of edges; plus
- A factorization rule: for any two colors (red, blue), any red-blue path from v to w is equivalent to a unique blue-red path.

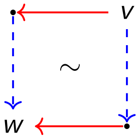


- Any 3-color path needs to determine a well-defined "cube."

Higher-rank graphs

A k-graph Λ consists of:

- An edge-colored directed graph, with k colors of edges; plus
- A factorization rule: for any two colors (red, blue), any red-blue path from v to w is equivalent to a unique blue-red path.



- Any 3-color path needs to determine a well-defined "cube."

Higher-rank graphs (k-graphs) were introduced by Kumjian & Pask in 2000 to give examples of combinatorial, computable C^*-algebras, more general than graph C^*-algebras.

Definition of $C^*(\Lambda)$

$C^*(\Lambda)$ is a universal C^*-algebra, generated by projections associated to the vertices in Λ and partial isometries associated to the edges, such that multiplication of these generators corresponds to concatenating the given edges in Λ.

Definition of $C^*(\Lambda)$

$C^*(\Lambda)$ is a universal C^*-algebra, generated by projections associated to the vertices in Λ and partial isometries associated to the edges, such that multiplication of these generators corresponds to concatenating the given edges in Λ.

Precisely: Given a row-finite source-free higher-rank graph Λ, $C^*(\Lambda)$ is the universal C^*-algebra generated by a family of projections $\{p_v : v \text{ a vertex in } \Lambda\}$ and partial isometries $\{s_e : e \text{ an edge in } \Lambda\}$ satisfying the Cuntz-Krieger relations:

Definition of $C^*(\Lambda)$

$C^*(\Lambda)$ is a universal C^*-algebra, generated by projections associated to the vertices in Λ and partial isometries associated to the edges, such that multiplication of these generators corresponds to concatenating the given edges in Λ.

Precisely: Given a row-finite source-free higher-rank graph Λ, $C^*(\Lambda)$ is the universal C^*-algebra generated by a family of projections $\{p_v : v \text{ a vertex in } \Lambda\}$ and partial isometries $\{s_e : e \text{ an edge in } \Lambda\}$ satisfying the Cuntz-Krieger relations:

(CK1) $p_v p_w = \delta_{v,w} p_v$

Definition of $C^*(\Lambda)$

$C^*(\Lambda)$ is a universal C^*-algebra, generated by projections associated to the vertices in Λ and partial isometries associated to the edges, such that multiplication of these generators corresponds to concatenating the given edges in Λ.

Precisely: Given a row-finite source-free higher-rank graph Λ, $C^*(\Lambda)$ is the universal C^*-algebra generated by a family of projections $\{p_v : v \text{ a vertex in } \Lambda\}$ and partial isometries $\{s_e : e \text{ an edge in } \Lambda\}$ satisfying the Cuntz-Krieger relations:

(CK1) $p_v p_w = \delta_{v,w} p_v$

(CK2) If $ef \sim f'e'$ then $s_e s_f = s_{f'} s_{e'}$

Definition of $C^*(\Lambda)$

$C^*(\Lambda)$ is a universal C^*-algebra, generated by projections associated to the vertices in Λ and partial isometries associated to the edges, such that multiplication of these generators corresponds to concatenating the given edges in Λ.

Precisely: Given a row-finite source-free higher-rank graph Λ, $C^*(\Lambda)$ is the universal C^*-algebra generated by a family of projections $\{p_v : v \text{ a vertex in } \Lambda\}$ and partial isometries $\{s_e : e \text{ an edge in } \Lambda\}$ satisfying the Cuntz-Krieger relations:

(CK1) $p_v p_w = \delta_{v,w} p_v$

(CK2) If $ef \sim f'e'$ then $s_e s_f = s_{f'} s_{e'}$

(CK3) $s_e^* s_e = p_{s(e)}$

$C^*(\Lambda)$ is a universal C^*-algebra, generated by projections associated to the vertices in Λ and partial isometries associated to the edges, such that multiplication of these generators corresponds to concatenating the given edges in Λ.

Precisely: Given a row-finite source-free higher-rank graph Λ, $C^*(\Lambda)$ is the universal C^*-algebra generated by a family of projections $\{p_v : v \text{ a vertex in } \Lambda\}$ and partial isometries $\{s_e : e \text{ an edge in } \Lambda\}$ satisfying the Cuntz-Krieger relations:

(CK1) $p_v p_w = \delta_{v,w} p_v$

(CK2) If $ef \sim f'e'$ then $s_e s_f = s_{f'} s_{e'}$

(CK3) $s_e^* s_e = p_{s(e)}$

(CK4) For any vertex v and any color i, $p_v = \sum_{e:d(e)=i, r(e)=v} s_e s_e^*$.

Universality

Recall: $C^*(\Lambda)$ is the universal C^*-algebra generated by $\{p_v, s_e\}$ satisfying the Cuntz-Krieger relations.

Universality

Recall: $C^*(\Lambda)$ is the universal C^*-algebra generated by $\{p_v, s_e\}$ satisfying the Cuntz-Krieger relations.

That is, if we have any family of projections and partial isometries $\{Q_v, T_e\}$ satisfying (CK1)-(CK4) – a Cuntz-Krieger family – then there is a surjective $*$-homomorphism $\pi : C^*(\Lambda) \to C^*(\{Q_v, T_e\})$.

Universality

Recall: $C^*(\Lambda)$ is the universal C^*-algebra generated by $\{p_v, s_e\}$ satisfying the Cuntz-Krieger relations.

That is, if we have any family of projections and partial isometries $\{Q_v, T_e\}$ satisfying (CK1)-(CK4) – a Cuntz-Krieger family – then there is a surjective $*$-homomorphism $\pi : C^*(\Lambda) \to C^*(\{Q_v, T_e\})$.

Theorem (Kumjian-Pask; "Gauge-invariant uniqueness theorem")

Let Λ be a k-graph. There is a continuous action α of $\mathbb{T}^k$ on $C^*(\Lambda)$, satisfying

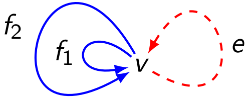
$$\alpha_z(s_e) = z_i s_e$$

if e is an edge of color i. If $\pi(p_v) \neq 0$ for all v, and there is also an action β of $\mathbb{T}^k$ on $C^*(\{Q_v, T_e\})$ such that

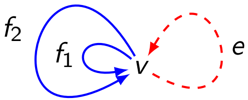
$$\pi \circ \alpha = \beta \circ \pi$$

then π is an isomorphism.

Example and notation

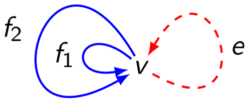


Example and notation



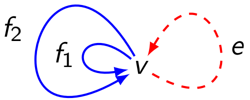
Factorization rule: $f_1 e = e f_2, \quad f_2 e = e f_1.$

Example and notation



Factorization rule: $f_1 e = e f_2, \quad f_2 e = e f_1$. $C^*(\Lambda) \cong \mathcal{O}_2 \rtimes \mathbb{Z}$

Example and notation

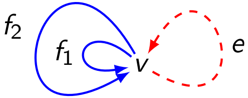


Factorization rule: $f_1 e = ef_2, \quad f_2 e = ef_1$. $C^*(\Lambda) \cong \mathcal{O}_2 \rtimes \mathbb{Z}$

For $n = (n_1, \ldots, n_k) \in \mathbb{N}^k$ we write

$$\Lambda^n = \{\lambda \in \Lambda : \lambda \text{ has } n_i \text{ edges of color } i\}.$$

Example and notation



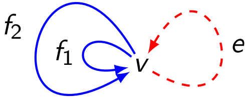
Factorization rule: $f_1 e = e f_2, \quad f_2 e = e f_1$. $C^*(\Lambda) \cong \mathcal{O}_2 \rtimes \mathbb{Z}$

For $n = (n_1, \ldots, n_k) \in \mathbb{N}^k$ we write

$$\Lambda^n = \{\lambda \in \Lambda : \lambda \text{ has } n_i \text{ edges of color } i\}.$$

The factorization rule means this is well defined.

Example and notation



Factorization rule: $f_1 e = e f_2, \quad f_2 e = e f_1$. $C^*(\Lambda) \cong \mathcal{O}_2 \rtimes \mathbb{Z}$

For $n = (n_1, \ldots, n_k) \in \mathbb{N}^k$ we write

$$\Lambda^n = \{\lambda \in \Lambda : \lambda \text{ has } n_i \text{ edges of color } i\}.$$

The factorization rule means this is well defined.

Note that Λ^0 is the vertices of Λ.

Sink deletion

Definition

A sink in a k-graph Λ is a vertex which emits no edges of color i, for some i.

Sink deletion

Definition

A <u>sink</u> in a k-graph Λ is a vertex which emits no edges of color i, for some i.

Definition

If $v, w \in \Lambda^0$ we say $v \geq w$ if there exists a path $\lambda \in \Lambda$ with $s(\lambda) = v, r(\lambda) = w$.

Sink deletion

Definition

A <u>sink</u> in a k-graph Λ is a vertex which emits no edges of color i, for some i.

Definition

If $v, w \in \Lambda^0$ we say $v \geq w$ if there exists a path $\lambda \in \Lambda$ with $s(\lambda) = v, r(\lambda) = w$.

Proposition (Eckhardt-Fieldhouse-Gent-G-Gonzales-Pask)

If v is a sink in Λ^0, then deleting v, all vertices w such that $w \geq v$, and all incident edges results in a k-graph Λ_S such that $C^*(\Lambda_S) \sim_{ME} C^*(\Lambda)$.

Example of sink deletion

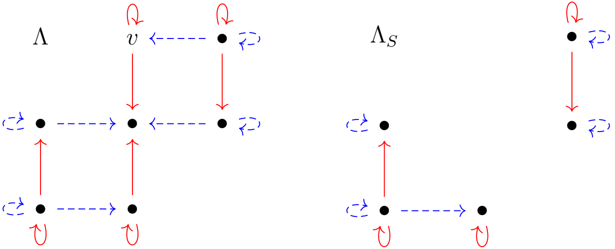


FIGURE 1. The 1-skeletons for k-graphs Λ and the resulting Λ_S with the sink v deleted.

Example of sink deletion

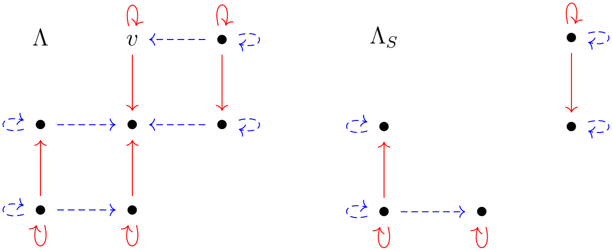


FIGURE 1. The 1-skeletons for k-graphs Λ and the resulting Λ_S with the sink v deleted.

In this case, $C^*(\Lambda) \sim_{ME} C(\mathbb{T}^2) \oplus C(\mathbb{T}^2)$, which we see if we delete the three remaining sinks.

Insplitting

Intuitively, to in-split at a vertex, we make two copies of the vertex; divide up the incoming edges; and duplicate the outgoing edges.

Insplitting

Intuitively, to in-split at a vertex, we make two copies of the vertex; divide up the incoming edges; and duplicate the outgoing edges.

With k-graphs, the factorization rule means we have to be careful about how we divide up the incoming edges.

Insplitting

Intuitively, to in-split at a vertex, we make two copies of the vertex; divide up the incoming edges; and duplicate the outgoing edges. With k-graphs, the factorization rule means we have to be careful about how we divide up the incoming edges.

Definition

Fix $v \in \Lambda^0$. Partition the edges with range v into two non-empty sets $\mathcal{E}_1$ and $\mathcal{E}_2$ satisfying the *pairing condition*: if $f, g \in r^{-1}(v)$ and there exist edges a_1, a_2 such that $fa_1 \sim ga_2$ in Λ, then f and g are contained in the same set.
We define Λ_I by $\Lambda_I^0 = \Lambda^0 \backslash \{v\} \cup \{v_1, v_2\}$; importing from Λ all edges not incident on v; edges in $\mathcal{E}_i$ have range v_i; and making two copies e_1, e_2, of all edges e with source v.

Insplitting

Intuitively, to in-split at a vertex, we make two copies of the vertex; divide up the incoming edges; and duplicate the outgoing edges. With k-graphs, the factorization rule means we have to be careful about how we divide up the incoming edges.

Definition

Fix $v \in \Lambda^0$. Partition the edges with range v into two non-empty sets $\mathcal{E}_1$ and $\mathcal{E}_2$ satisfying the *pairing condition*: if $f, g \in r^{-1}(v)$ and there exist edges a_1, a_2 such that $fa_1 \sim ga_2$ in Λ, then f and g are contained in the same set.

We define Λ_I by $\Lambda_I^0 = \Lambda^0 \backslash \{v\} \cup \{v_1, v_2\}$; importing from Λ all edges not incident on v; edges in $\mathcal{E}_i$ have range v_i; and making two copies e_1, e_2, of all edges e with source v. In Λ_I, if e, f are not "duplicated" edges, we have $ef \sim_I f'e'$ iff $ef \sim f'e'$ in Λ. We define $e_i f :\sim_I f'e'$ if $s(e_i) = v_i, f \in \mathcal{E}_i$, and $ef \sim f'e'$ in Λ.

Insplitting

Intuitively, to in-split at a vertex, we make two copies of the vertex; divide up the incoming edges; and duplicate the outgoing edges. With k-graphs, the factorization rule means we have to be careful about how we divide up the incoming edges.

Definition

Fix $v \in \Lambda^0$. Partition the edges with range v into two non-empty sets $\mathcal{E}_1$ and $\mathcal{E}_2$ satisfying the *pairing condition*: if $f, g \in r^{-1}(v)$ and there exist edges a_1, a_2 such that $fa_1 \sim ga_2$ in Λ, then f and g are contained in the same set.
We define Λ_I by $\Lambda_I^0 = \Lambda^0 \backslash \{v\} \cup \{v_1, v_2\}$; importing from Λ all edges not incident on v; edges in $\mathcal{E}_i$ have range v_i; and making two copies e_1, e_2, of all edges e with source v. In Λ_I, if e, f are not "duplicated" edges, we have $ef \sim_I f'e'$ iff $ef \sim f'e'$ in Λ. We define $e_i f :\sim_I f'e'$ if $s(e_i) = v_i, f \in \mathcal{E}_i$, and $ef \sim f'e'$ in Λ.

The pairing condition ensures that we can define the factorization in Λ_I by importing the factorization in Λ.

Insplitting examples

Not all k-graphs can be insplit at all vertices; this depends on the factorization rule as well as on the underlying directed graph.

Insplitting examples

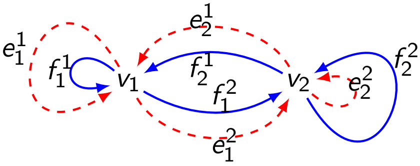
Not all k-graphs can be insplit at all vertices; this depends on the factorization rule as well as on the underlying directed graph.

If $e^i f^j = f^j e^i$, then Λ cannot be insplit.

Insplitting examples

Not all k-graphs can be insplit at all vertices; this depends on the factorization rule as well as on the underlying directed graph.

If $e^i f^j = f^j e^i$, then Λ cannot be insplit. If $e^i f^j = f^i e^j$, then take $\mathcal{E}_1 = \{e^1, f^1\}, \mathcal{E}_2 = \{e^2, f^2\}$. Then in Λ_I, we have $e^i_j f^j_k \sim_I f^i_j e^j_k$.



Λ_I

Insplitting and isomorphism

Theorem (Eckhardt-Fieldhouse-Gent-G-Gonzales-Pask)

If Λ is a row-finite, source-free k-graph then $C^*(\Lambda_I) \cong C^*(\Lambda)$.

Insplitting and isomorphism

Theorem (Eckhardt-Fieldhouse-Gent-G-Gonzales-Pask)

If Λ is a row-finite, source-free k-graph then $C^*(\Lambda_I) \cong C^*(\Lambda)$.

Idea: Define a Cuntz-Krieger Λ-family $\{Q_w, T_e\}$ in $C^*(\Lambda_I)$:

$$Q_w = \begin{cases} p_w, & w \neq v \\ p_{v_1} + p_{v_2}, & w = v \end{cases} \qquad T_e = \begin{cases} s_e, & s(e) \neq v \\ s_{e_1} + s_{e_2}, & s(e) = v. \end{cases}$$

Insplitting and isomorphism

Theorem (Eckhardt-Fieldhouse-Gent-G-Gonzales-Pask)

If Λ is a row-finite, source-free k-graph then $C^*(\Lambda_I) \cong C^*(\Lambda)$.

Idea: Define a Cuntz-Krieger Λ-family $\{Q_w, T_e\}$ in $C^*(\Lambda_I)$:

$$Q_w = \begin{cases} p_w, & w \neq v \\ p_{v_1} + p_{v_2}, & w = v \end{cases} \qquad T_e = \begin{cases} s_e, & s(e) \neq v \\ s_{e_1} + s_{e_2}, & s(e) = v. \end{cases}$$

Then show we get an onto map $\psi : C^*(\Lambda) \to C^*(\Lambda_I)$.

Insplitting and isomorphism

Theorem (Eckhardt-Fieldhouse-Gent-G-Gonzales-Pask)

If Λ is a row-finite, source-free k-graph then $C^*(\Lambda_I) \cong C^*(\Lambda)$.

Idea: Define a Cuntz-Krieger Λ-family $\{Q_w, T_e\}$ in $C^*(\Lambda_I)$:

$$Q_w = \begin{cases} p_w, & w \neq v \\ p_{v_1} + p_{v_2}, & w = v \end{cases} \qquad T_e = \begin{cases} s_e, & s(e) \neq v \\ s_{e_1} + s_{e_2}, & s(e) = v. \end{cases}$$

Then show we get an onto map $\psi : C^*(\Lambda) \to C^*(\Lambda_I)$. Use the gauge-invariant uniqueness theorem to prove that $C^*(\Lambda_I) \cong C^*(\Lambda)$.

Delay

For directed graphs, we delay at an edge e by breaking it into two edges.

Delay

For directed graphs, we delay at an edge e by breaking it into two edges. In k-graphs, this introduces complications.

Delay

For directed graphs, we delay at an edge e by breaking it into two edges. In k-graphs, this introduces complications.
Suppose we want to delay at an edge e in Λ. Set $\mathcal{E}^1 = \{e\}$.

Delay

For directed graphs, we delay at an edge e by breaking it into two edges. In k-graphs, this introduces complications.

Suppose we want to delay at an edge e in Λ. Set $\mathcal{E}^1 = \{e\}$.

For each edge $e \in \mathcal{E}^1$, if $ef \sim f'e'$, add e' to $\mathcal{E}^1$.

Delay

For directed graphs, we delay at an edge e by breaking it into two edges. In k-graphs, this introduces complications.
Suppose we want to delay at an edge e in Λ. Set $\mathcal{E}^1 = \{e\}$.
For each edge $e \in \mathcal{E}^1$, if $ef \sim f'e'$, add e' to $\mathcal{E}^1$. Wash rinse repeat.

Delay

For directed graphs, we delay at an edge e by breaking it into two edges. In k-graphs, this introduces complications.

Suppose we want to delay at an edge e in Λ. Set $\mathcal{E}^1 = \{e\}$.

For each edge $e \in \mathcal{E}^1$, if $ef \sim f'e'$, add e' to $\mathcal{E}^1$. Wash rinse repeat.

Note that all edges in $\mathcal{E}^1$ are the same color, black say.

In Λ_D, we will delay at all edges in $\mathcal{E}^1$

Delay

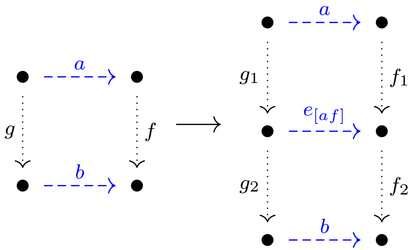
For directed graphs, we delay at an edge e by breaking it into two edges. In k-graphs, this introduces complications.

Suppose we want to delay at an edge e in Λ. Set $\mathcal{E}^1 = \{e\}$.

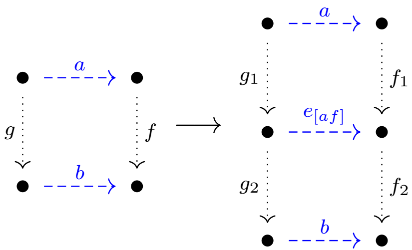
For each edge $e \in \mathcal{E}^1$, if $ef \sim f'e'$, add e' to $\mathcal{E}^1$. Wash rinse repeat.

Note that all edges in $\mathcal{E}^1$ are the same color, black say.

In Λ_D, we will delay at all edges in $\mathcal{E}^1$ – for each $e \in \mathcal{E}_1$, we add a vertex v_e, and replace e with e_1, e_2.

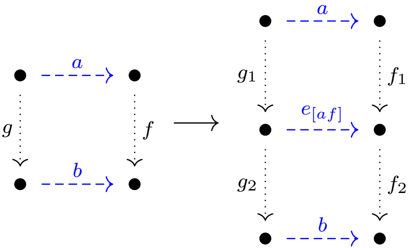


Delay



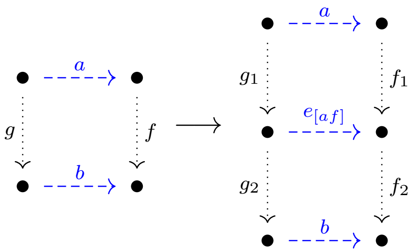
Many commuting squares α in Λ have now become rectangles;

Delay



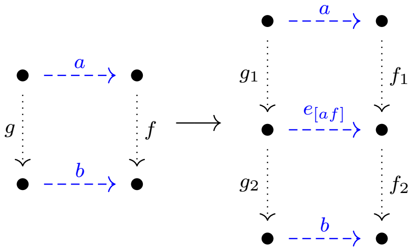
Many commuting squares α in Λ have now become rectangles; add an edge e_α for each.

Delay



Many commuting squares α in Λ have now become rectangles; add an edge e_α for each. If α had black and blue edges, then e_α is blue.

Delay



Many commuting squares α in Λ have now become rectangles; add an edge e_α for each. If α had black and blue edges, then e_α is blue.

The factorization in Λ_D essentially comes from the factorization in Λ, but there are lots of cases to check.

Delay: theorem

Theorem (Eckhardt-Fieldhouse-Gent-G-Gonzales-Pask)

If Λ is a row-finite source-free k-graph, then so is Λ_D.

Delay: theorem

Theorem (Eckhardt-Fieldhouse-Gent-G-Gonzales-Pask)

If Λ is a row-finite source-free k-graph, then so is Λ_D. Moreover, $C^*(\Lambda_D) \sim_{ME} C^*(\Lambda)$.

Delay: theorem

Theorem (Eckhardt-Fieldhouse-Gent-G-Gonzales-Pask)

If Λ is a row-finite source-free k-graph, then so is Λ_D. Moreover, $C^*(\Lambda_D) \sim_{ME} C^*(\Lambda)$.

Proof.

If $C^*(\Lambda_D) = C^*(\{p_v, s_e\})$, define

$$q_v = p_v \ \forall \ v \in \Lambda^0; \qquad t_e = \begin{cases} s_e, & e \notin \mathcal{E}^1 \\ s_{e_2} s_{e_1}, & e \in \mathcal{E}^1 \end{cases}.$$

Delay: theorem

Theorem (Eckhardt-Fieldhouse-Gent-G-Gonzales-Pask)

If Λ is a row-finite source-free k-graph, then so is Λ_D. Moreover, $C^*(\Lambda_D) \sim_{ME} C^*(\Lambda)$.

Proof.

If $C^*(\Lambda_D) = C^*(\{p_v, s_e\})$, define

$$q_v = p_v \ \forall \ v \in \Lambda^0; \qquad t_e = \begin{cases} s_e, & e \notin \mathcal{E}^1 \\ s_{e_2} s_{e_1}, & e \in \mathcal{E}^1 \end{cases}.$$

Then $\{q_v, t_e\}$ is a Cuntz-Krieger Λ-family in $C^*(\Lambda_D)$;

Delay: theorem

Theorem (Eckhardt-Fieldhouse-Gent-G-Gonzales-Pask)

If Λ is a row-finite source-free k-graph, then so is Λ_D. Moreover, $C^*(\Lambda_D) \sim_{ME} C^*(\Lambda)$.

Proof.

If $C^*(\Lambda_D) = C^*(\{p_v, s_e\})$, define

$$q_v = p_v \ \forall\ v \in \Lambda^0; \qquad t_e = \begin{cases} s_e, & e \notin \mathcal{E}^1 \\ s_{e_2} s_{e_1}, & e \in \mathcal{E}^1 \end{cases}.$$

Then $\{q_v, t_e\}$ is a Cuntz-Krieger Λ-family in $C^*(\Lambda_D)$; the gauge-invariant uniqueness theorem tells us $C^*(\Lambda) \cong C^*(\{q_v, t_e\})$.

Delay: theorem

Theorem (Eckhardt-Fieldhouse-Gent-G-Gonzales-Pask)

If Λ is a row-finite source-free k-graph, then so is Λ_D. Moreover, $C^*(\Lambda_D) \sim_{ME} C^*(\Lambda)$.

Proof.

If $C^*(\Lambda_D) = C^*(\{p_v, s_e\})$, define

$$q_v = p_v \ \forall \ v \in \Lambda^0; \qquad t_e = \begin{cases} s_e, & e \notin \mathcal{E}^1 \\ s_{e_2} s_{e_1}, & e \in \mathcal{E}^1 \end{cases}.$$

Then $\{q_v, t_e\}$ is a Cuntz-Krieger Λ-family in $C^*(\Lambda_D)$; the gauge-invariant uniqueness theorem tells us $C^*(\Lambda) \cong C^*(\{q_v, t_e\})$. If $p = \sum_{v \in \Lambda^0} p_v$, then $C^*(\{q_v, t_e\}) \cong pC^*(\Lambda_D)p$ is a full corner of $C^*(\Lambda_D)$. □

Reduction

For directed graphs, if all edges with range v have the same source, and $s^{-1}(v) = \{e\}$ with $r(e) \neq v$, we can reduce at v:

Reduction

For directed graphs, if all edges with range v have the same source, and $s^{-1}(v) = \{e\}$ with $r(e) \neq v$, we can reduce at v: basically, delete e.

Reduction

For directed graphs, if all edges with range v have the same source, and $s^{-1}(v) = \{e\}$ with $r(e) \neq v$, we can reduce at v: basically, delete e.

For k-graphs, if

- $r^{-1}(v)$ and $s^{-1}(v)$ both contain a single edge of each color;

Reduction

For directed graphs, if all edges with range v have the same source, and $s^{-1}(v) = \{e\}$ with $r(e) \neq v$, we can reduce at v: basically, delete e.

For k-graphs, if

- $r^{-1}(v)$ and $s^{-1}(v)$ both contain a single edge of each color;
- $|s(r^{-1}(v))| = |r(s^{-1}(v))| = 1$;

Reduction

For directed graphs, if all edges with range v have the same source, and $s^{-1}(v) = \{e\}$ with $r(e) \neq v$, we can reduce at v: basically, delete e.

For k-graphs, if

- $r^{-1}(v)$ and $s^{-1}(v)$ both contain a single edge of each color;
- $|s(r^{-1}(v))| = |r(s^{-1}(v))| = 1$;
- If $s(e) = v$ and $ef = f'e'$ then $s(f') = e$;

Reduction

For directed graphs, if all edges with range v have the same source, and $s^{-1}(v) = \{e\}$ with $r(e) \neq v$, we can reduce at v: basically, delete e.

For k-graphs, if

- $r^{-1}(v)$ and $s^{-1}(v)$ both contain a single edge of each color;
- $|s(r^{-1}(v))| = |r(s^{-1}(v))| = 1$;
- If $s(e) = v$ and $ef = f'e'$ then $s(f') = e$;

then we can reduce at v.

Reduction: theorem

Theorem (Eckhardt-Fieldhouse-Gent-G-Gonzales-Pask)

The k-graph Λ_R resulting from reducing at v satisfies $C^*(\Lambda_R) \sim_{ME} C^*(\Lambda)$.

Reduction: theorem

Theorem (Eckhardt-Fieldhouse-Gent-G-Gonzales-Pask)

The k-graph Λ_R resulting from reducing at v satisfies $C^*(\Lambda_R) \sim_{ME} C^*(\Lambda)$.

Proof.

Pick an edge $f \in s^{-1}(v)$; define a Cuntz-Krieger Λ_R-family in $C^*(\Lambda) = C^*\{p_v, s_e\})$ by

$$q_v = p_v; \qquad t_e = \begin{cases} s_f s_e, & r(e) = v \\ s_f, & r(e) \neq v \end{cases}.$$

Reduction: theorem

Theorem (Eckhardt-Fieldhouse-Gent-G-Gonzales-Pask)

The k-graph Λ_R resulting from reducing at v satisfies $C^*(\Lambda_R) \sim_{ME} C^*(\Lambda)$.

Proof.

Pick an edge $f \in s^{-1}(v)$; define a Cuntz-Krieger Λ_R-family in $C^*(\Lambda) = C^*\{p_v, s_e\})$ by

$$q_v = p_v; \qquad t_e = \begin{cases} s_f s_e, & r(e) = v \\ s_f, & r(e) \neq v \end{cases}.$$

Again, check that $C^*(\{q_v, t_e\}) \cong C^*(\Lambda_R)$, using gauge-invariant uniqueness theorem, and that it's a full corner in $C^*(\Lambda)$. □

The end

Thanks for listening!

This research was supported by the National Science Foundation.

- M. Boyle and D. Huang, Poset block equivalence of integral matrices, Trans. Amer. Math. Soc. **355**.
- T. Bates and D. Pask, Flow equivalence of graph algebras, Ergodic Theory Dynam. Systems **24** (2004), no. 2, 367–382.
- T. Crisp and D. Gow, Contractible subgraphs and Morita equivalence of graph C^*-algebras, Proc. Amer. Math. Soc. **134** (2006), no. 7, 2003–2013.
- S. Eilers, G. Restorff, E. Ruiz, and A. P. W. Sørensen, The complete classification of unital graph C^*-algebras: Geometric and strong, arXiv:1611.07120, 2016.
- M. Rørdam, Classification of Cuntz-Krieger algebras, K-Theory **9** (1995), no. 1, 31–58.