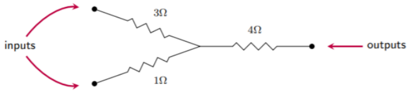
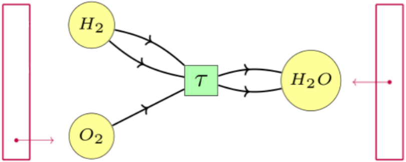
A recipe for black box functors

Maru Sarazola and Brendan Fong

What is a black box functor?

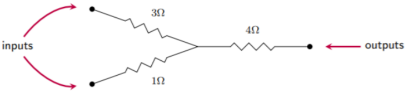
In many disciplines, network diagrams are used to model interconnected systems

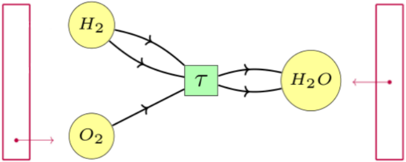




What is a black box functor?

In many disciplines, network diagrams are used to model interconnected systems





Recent work uses hypergraph categories to describe the structure of these systems (ex. electrical circuits, chemical reactions, Markov processes, automata, ...)

Hypergraph category: symmetric monoidal category where every object has a Frobenius structure, i.e. a monoid and a comonoid structure + extra laws.

What is a black box functor?

Idea:

- objects model boundary types
- morphisms formalize the syntax

What about the semantics?

What is a black box functor?

Idea:

- objects model boundary types
- morphisms formalize the syntax

What about the semantics?

We typically construct functors to other hypergraph categories where we interpret the semantics.

This often has the effect of hiding internal structure inaccessible from the boundary: we call them black box functors.

Building hypergraph categories: Cospans

Given $\mathcal{C}$ finitely cocomplete, there exists a symmetric monoidal category $\mathsf{Cospan}(\mathcal{C})$:

- objects: objects of $\mathcal{C}$
- morphisms: (iso classes of) diagrams

$$\begin{tikzcd} & S & \\ X \arrow[ur, "i"] & & Y \arrow[ul, "o"'] \end{tikzcd}$$

composition given by pullback, and $\otimes$ inherited from coproduct in $\mathcal{C}$.

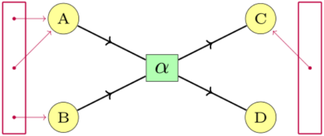
Building hypergraph categories: Cospans

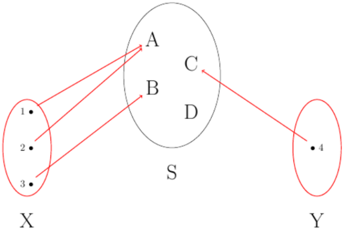
Given $\mathcal{C}$ finitely cocomplete, there exists a symmetric monoidal category $\mathsf{Cospan}(\mathcal{C})$:

- objects: objects of $\mathcal{C}$
- morphisms: (iso classes of) diagrams

$$\begin{tikzcd} & S & \\ X \arrow[ur, "i"] & & Y \arrow[ul, "o"'] \end{tikzcd}$$

composition given by pullback, and $\otimes$ inherited from coproduct in $\mathcal{C}$.





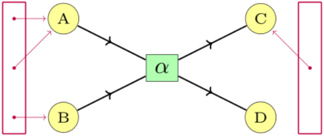
Building hypergraph categories: Cospans

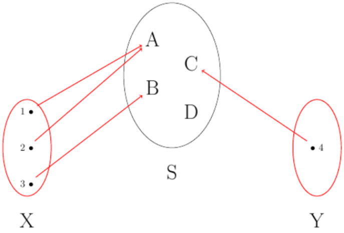
Given $\mathcal{C}$ finitely cocomplete, there exists a symmetric monoidal category $\mathsf{Cospan}(\mathcal{C})$:

- objects: objects of $\mathcal{C}$
- morphisms: (iso classes of) diagrams

$$\begin{tikzcd} & S & \\ X \arrow[ur, "i"] & & Y \arrow[ul, "o"'] \end{tikzcd}$$

composition given by pullback, and $\otimes$ inherited from coproduct in $\mathcal{C}$.





Flaw: the nodes may have more information that's not being recorded (like the rate α, or the resistance values in a circuit).

Building hyp cats: ~~Cospans~~ Decorated cospans

$$\left(\begin{tikzcd} & S & \\ X \arrow[ur, "i"] & & Y \arrow[ul, "o"'] \end{tikzcd} , \quad \begin{matrix} \text{extra info} \\ \text{on the apex} \end{matrix} \right)$$

Building hyp cats: ~~Cospans~~ Decorated cospans

$$\left(\begin{tikzcd} & S & \\ X \arrow[ur, "i"] & & Y \arrow[ul, "o"'] \end{tikzcd} , \quad \begin{array}{c} \text{extra info} \\ \text{on the apex} \end{array} \right)$$

Not obvious that these compose, but:

Thm. [Fong]

If the decorations are given by a symmetric lax monoidal functor

$$F : (\mathcal{C}, +) \to (\mathsf{Set}, \times),$$

then we can form a category $F\mathsf{Cospan}$ whose objects are the same as $\mathcal{C}$, and whose morphisms are (iso classes of) decorated cospans

$$\left(\begin{tikzcd} & S & \\ X \arrow[ur, "i"] & & Y \arrow[ul, "o"'] \end{tikzcd} , \quad \begin{tikzcd} FS \\ 1 \arrow[u, "s"'] \end{tikzcd} \right)$$

Building hyp cats: ~~Cospans~~ Decorated cospans

Thm. [Fong]

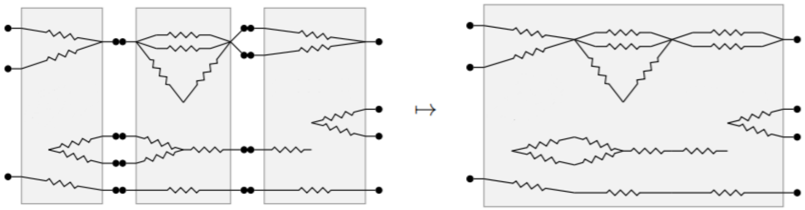
$F\mathsf{Cospan}$ is a hypergraph category with $\otimes$ inherited from coproduct in $\mathcal{C}$.

Building hyp cats: ~~Cospans~~ Decorated cospans

Thm. [Fong]

$F\mathsf{Cospan}$ is a hypergraph category with $\otimes$ inherited from coproduct in $\mathcal{C}$.

Flaw: from our perspective, this is not efficient: cospans accumulate inaccessible information.



Building hyp cats: Decorated ~~cospans~~ corelations

A **factorization system** is a pair $(\mathcal{E}, \mathcal{M})$ of subcategories of $\mathcal{C}$ such that every map $f \in \mathcal{C}$ factors as $f = me$ for $m \in \mathcal{M},\ e \in \mathcal{E}$.

An $(\mathcal{E}, \mathcal{M})$**-corelation** is a cospan $X \xrightarrow{i} S \xleftarrow{o} Y$ such that the universal map $[i, 0]$ belongs to $\mathcal{E}$

$$\begin{tikzcd}
& S & \\
& X + Y \arrow[u, dashed, "{[i,o]}"] & \\
X \arrow[ur, "\iota_X"'] \arrow[uur, bend left, "i"] & & Y \arrow[ul, "\iota_Y"] \arrow[uul, bend right, "o"']
\end{tikzcd}$$

Building hyp cats: Decorated ~~cospans~~ corelations

A **factorization system** is a pair $(\mathcal{E}, \mathcal{M})$ of subcategories of $\mathcal{C}$ such that every map $f \in \mathcal{C}$ factors as $f = me$ for $m \in \mathcal{M},\ e \in \mathcal{E}$.

An $(\mathcal{E}, \mathcal{M})$**-corelation** is a cospan $X \xrightarrow{i} S \xleftarrow{o} Y$ such that the universal map $[i, 0]$ belongs to $\mathcal{E}$

$$\begin{tikzcd}
 & S & \\
 & X + Y \arrow[u, dashed, "{[i,o]}"] & \\
X \arrow[uur, bend left, "i"] \arrow[ur, "\iota_X"'] & & Y \arrow[uul, bend right, "o"'] \arrow[ul, "\iota_Y"]
\end{tikzcd}$$

Idea: the maps in $\mathcal{E}$ control how much of the apex is “reached” by the boundary.

Building hyp cats: Decorated ~~cospans~~ corelations

Thm. [Fong]

Given a factorization system $(\mathcal{E}, \mathcal{M})$ with $\mathcal{M}$ stable under pushouts, and a symmetric lax monoidal functor

$$F\colon (\mathcal{C} ⨟ \mathcal{M}^{\mathrm{op}}, +) \to (\mathsf{Set}, \times),$$

we can form a category $F\mathsf{Corel}$ whose objects are the same as $\mathcal{C}$, and whose morphisms are (iso classes of) decorated corelations

$$\left(\begin{tikzcd} & S & \\ X \arrow[ur, "i"] & & Y \arrow[ul, "o"'] \end{tikzcd} , \begin{tikzcd} FS \\ 1 \arrow[u, "s"'] \end{tikzcd} \right)$$

Building hyp cats: Decorated ~~cospans~~ corelations

Thm. [Fong]

Given a factorization system $(\mathcal{E}, \mathcal{M})$ with $\mathcal{M}$ stable under pushouts, and a symmetric lax monoidal functor

$$F\colon (\mathcal{C}⨟\mathcal{M}^{\mathrm{op}}, +) \to (\mathsf{Set}, \times),$$

we can form a category $F\mathsf{Corel}$ whose objects are the same as $\mathcal{C}$, and whose morphisms are (iso classes of) decorated corelations

$$\left(\begin{tikzcd} & S & \\ X \arrow[ur, "i"] & & Y \arrow[ul, "o"'] \end{tikzcd} \quad , \quad \begin{tikzcd} FS \\ 1 \arrow[u, "s"'] \end{tikzcd} \right)$$

Thm. [Fong]

$F\mathsf{Corel}$ is a hypergraph category with $\otimes$ inherited from coproduct in $\mathcal{C}$.

Recipe for black box functors

Thm. [Fong]

Consider two symmetric lax monoidal functors

$$F\colon (\mathcal{C}\,⨾\,\mathcal{M}^{\mathrm{op}}, +) \to (\mathsf{Set}, \times)$$

$$F'\colon (\mathcal{C}'\,⨾\,\mathcal{M}'^{\mathrm{op}}, +) \to (\mathsf{Set}, \times).$$

A cocontinuous functor $A : \mathcal{C} \to \mathcal{C}'$ such that $A(\mathcal{M}) \subseteq \mathcal{M}'$, together with a monoidal natural transformation

$$\begin{tikzcd} \mathcal{C}\,⨾\,\mathcal{M}^{\mathrm{op}} \arrow[r, "F"] \arrow[d, "A"'] \arrow[dr, phantom, "\Downarrow\alpha"] & \mathsf{Set} \\ \mathcal{C}'\,⨾\,\mathcal{M}'^{\mathrm{op}} \arrow[ur, "F'"'] & \end{tikzcd}$$

induce a hypergraph functor $F\mathsf{Corel} \to F'\mathsf{Corel}$, mapping $X \mapsto A(X)$.

Recipe for black box functors

Why is this desirable? It reduces defining a black box functor to checking conditions in $\mathcal{C}, \mathcal{C}'$ and Set, instead of working with decorated corelations.

Recipe for black box functors

Why is this desirable? It reduces defining a black box functor to checking conditions in $\mathcal{C}, \mathcal{C}'$ and Set, instead of working with decorated corelations.

Problem: what happens when we want a black box functor between hypergraph categories, but (at least) one of them is not of the form $F\mathsf{Corel}$?

We want a “recipe” for these black box functors as well.

Recipe for black box functors

Why is this desirable? It reduces defining a black box functor to checking conditions in $\mathcal{C}, \mathcal{C}'$ and Set, instead of working with decorated corelations.

Problem: what happens when we want a black box functor between hypergraph categories, but (at least) one of them is not of the form $F\mathsf{Corel}$?

We want a "recipe" for these black box functors as well.

Solution: we should be working in a different category!

The category $\mathsf{DecData}$ of decorating data

We define the category $\mathsf{DecData}$, having

- objects: tuples $(\mathcal{C}, (\mathcal{E}, \mathcal{M}), F)$ for $F\colon (\mathcal{C}⨟\mathcal{M}^{\mathrm{op}}, +) \to (\mathsf{Set}, \times)$
- morphisms: pairs (A, α) where $A : \mathcal{C} \to \mathcal{C}'$ with $A(\mathcal{M}) \subseteq \mathcal{M}'$ and $\alpha : F \Rightarrow F'A$.

The category DecData of decorating data

We define the category DecData, having

- objects: tuples $(\mathcal{C}, (\mathcal{E}, \mathcal{M}), F)$ for $F\colon (\mathcal{C}{\fatsemi}\mathcal{M}^{\mathrm{op}}, +) \to (\mathsf{Set}, \times)$
- morphisms: pairs (A, α) where $A : \mathcal{C} \to \mathcal{C}'$ with $A(\mathcal{M}) \subseteq \mathcal{M}'$ and $\alpha : F \Rightarrow F'A$.

Thm. [Fong, S]

The decorated corelations construction assembles into a functor

$$(-)\mathsf{Corel}\colon\ \mathsf{DecData} \longrightarrow \mathsf{Hyp}$$

which, on objects, takes decorating data $(\mathcal{C}, (\mathcal{E}, \mathcal{M}), F)$ to the hypergraph category $F\mathsf{Corel}$, and whose action on morphisms is given by the recipe mentioned earlier.

The category DecData of decorating data

We define the category DecData, having

- objects: tuples $(\mathcal{C}, (\mathcal{E}, \mathcal{M}), F)$ for $F\colon (\mathcal{C} ⨟ \mathcal{M}^{\mathrm{op}}, +) \to (\mathsf{Set}, \times)$
- morphisms: pairs (A, α) where $A : \mathcal{C} \to \mathcal{C}'$ with $A(\mathcal{M}) \subseteq \mathcal{M}'$ and $\alpha : F \Rightarrow F'A$.

Thm. [Fong, S]

The decorated corelations construction assembles into a functor

$$(-)\mathsf{Corel}\colon\ \mathsf{DecData} \longrightarrow \mathsf{Hyp}$$

which, on objects, takes decorating data $(\mathcal{C}, (\mathcal{E}, \mathcal{M}), F)$ to the hypergraph category $F\mathsf{Corel}$, and whose action on morphisms is given by the recipe mentioned earlier.

Claim: DecData is the place to live!

$\mathsf{DecData}$ is the right setting

We can construct a functor

$$\mathsf{Alg}\colon \mathsf{Hyp} \to \mathsf{DecData}$$

that actually only produces decorating data $(\mathcal{C}, (\mathsf{Isos}, \mathcal{C}), F)$ with trivial factorization system.

DecData is the right setting

We can construct a functor

$$\mathsf{Alg}\colon \mathsf{Hyp} \to \mathsf{DecData}$$

that actually only produces decorating data $(\mathcal{C}, (\text{Isos}, \mathcal{C}), F)$ with trivial factorization system.

Let $\mathsf{CospanAlg}$ be the full subcategory of $\mathsf{DecData}$ whose objects are the tuples $(\mathcal{C}, (\text{Isos}, \mathcal{C}), F)$.

$\mathsf{DecData}$ is the right setting

We can construct a functor

$$\mathsf{Alg}\colon \mathsf{Hyp} \to \mathsf{DecData}$$

that actually only produces decorating data $(\mathcal{C}, (\mathsf{Isos}, \mathcal{C}), F)$ with trivial factorization system.

Let $\mathsf{CospanAlg}$ be the full subcategory of $\mathsf{DecData}$ whose objects are the tuples $(\mathcal{C}, (\mathsf{Isos}, \mathcal{C}), F)$.

Explicitly, objects are pairs $(\mathcal{C}, F)$ for a lax monoidal functor

$$F\colon \mathcal{C} ⨟ \mathcal{M}^{\mathrm{op}} \to \mathsf{Set}$$

$\mathsf{DecData}$ is the right setting

We can construct a functor

$$\mathsf{Alg}\colon\ \mathsf{Hyp} \to \mathsf{DecData}$$

that actually only produces decorating data $(\mathcal{C}, (\mathsf{Isos}, \mathcal{C}), F)$ with trivial factorization system.

Let $\mathsf{CospanAlg}$ be the full subcategory of $\mathsf{DecData}$ whose objects are the tuples $(\mathcal{C}, (\mathsf{Isos}, \mathcal{C}), F)$.

Explicitly, objects are pairs $(\mathcal{C}, F)$ for a lax monoidal functor

$$F\colon \mathsf{Cospan}(\mathcal{C}) \to \mathsf{Set}$$

$\mathsf{DecData}$ is the right setting

Let $\mathsf{CospanAlg}$ be the full subcategory of $\mathsf{DecData}$ whose objects are pairs $(\mathcal{C}, F)$ for a lax monoidal functor

$$F \colon \mathsf{Cospan}(\mathcal{C}) \to \mathsf{Set}$$

DecData is the right setting

Let CospanAlg be the full subcategory of DecData whose objects are pairs $(\mathcal{C}, F)$ for a lax monoidal functor

$$F\colon \mathsf{Cospan}(\mathcal{C}) \to \mathsf{Set}$$

$$\mathsf{Hyp} \xrightarrow{\mathsf{Alg}} \mathsf{CospanAlg}$$

$\mathsf{DecData}$ is the right setting

Let $\mathsf{CospanAlg}$ be the full subcategory of $\mathsf{DecData}$ whose objects are pairs $(\mathcal{C}, F)$ for a lax monoidal functor

$$F\colon \mathsf{Cospan}(\mathcal{C}) \to \mathsf{Set}$$

$$\mathsf{Hyp} \xrightarrow{\mathsf{Alg}} \mathsf{CospanAlg} \xhookrightarrow{\iota} \mathsf{DecData}$$

DecData is the right setting

Let CospanAlg be the full subcategory of DecData whose objects are pairs $(\mathcal{C}, F)$ for a lax monoidal functor

$$F\colon \mathsf{Cospan}(\mathcal{C}) \to \mathsf{Set}$$

$$\mathsf{Hyp} \xrightarrow{\mathsf{Alg}} \mathsf{CospanAlg} \xhookrightarrow{\iota} \mathsf{DecData}$$

Prop. [Fong,S]

There exists a functor $\mathsf{Kan}\colon\ \mathsf{DecData} \to \mathsf{CospanAlg}$, mapping $(\mathcal{C}, (\mathcal{E}, \mathcal{M}), F) \mapsto (\mathcal{C},\quad)$

DecData is the right setting

Let CospanAlg be the full subcategory of DecData whose objects are pairs $(\mathcal{C}, F)$ for a lax monoidal functor

$$F\colon \mathsf{Cospan}(\mathcal{C}) \to \mathsf{Set}$$

$$\mathsf{Hyp} \xrightarrow{\mathsf{Alg}} \mathsf{CospanAlg} \overset{\iota}{\hookrightarrow} \mathsf{DecData}$$

Prop. [Fong,S]

There exists a functor $\mathsf{Kan}\colon\ \mathsf{DecData} \to \mathsf{CospanAlg}$, mapping $(\mathcal{C}, (\mathcal{E}, \mathcal{M}), F) \mapsto (\mathcal{C},\quad)$

$$\begin{tikzcd} \mathcal{C} ⨾ \mathcal{M}^{\mathrm{op}} \arrow[r, "F"] & \mathsf{Set} \\ \mathsf{Cospan}(\mathcal{C}) \arrow[ur, dashed] & \end{tikzcd}$$

$\mathsf{DecData}$ is the right setting

Let $\mathsf{CospanAlg}$ be the full subcategory of $\mathsf{DecData}$ whose objects are pairs $(\mathcal{C}, F)$ for a lax monoidal functor

$$F \colon \mathsf{Cospan}(\mathcal{C}) \to \mathsf{Set}$$

$$\mathsf{Hyp} \xrightarrow{\mathsf{Alg}} \mathsf{CospanAlg} \overset{\iota}{\hookrightarrow} \mathsf{DecData}$$

Prop. [Fong,S]

There exists a functor $\mathsf{Kan} \colon \mathsf{DecData} \to \mathsf{CospanAlg}$, mapping $(\mathcal{C}, (\mathcal{E}, \mathcal{M}), F) \mapsto (\mathcal{C}, \quad)$

$$\begin{tikzcd}
\mathcal{C} ⨟ \mathcal{M}^{\mathrm{op}} \arrow[r, "F"] \arrow[d, hook] & \mathsf{Set} \\
\mathsf{Cospan}(\mathcal{C}) \arrow[ur, dashed] &
\end{tikzcd}$$

$\mathsf{DecData}$ is the right setting

Let $\mathsf{CospanAlg}$ be the full subcategory of $\mathsf{DecData}$ whose objects are pairs $(\mathcal{C}, F)$ for a lax monoidal functor

$$F\colon \mathsf{Cospan}(\mathcal{C}) \to \mathsf{Set}$$

$$\mathsf{Hyp} \xrightarrow{\mathsf{Alg}} \mathsf{CospanAlg} \overset{\iota}{\hookrightarrow} \mathsf{DecData}$$

Prop. [Fong,S]

There exists a functor $\mathsf{Kan}\colon\ \mathsf{DecData} \to \mathsf{CospanAlg}$, mapping $(\mathcal{C}, (\mathcal{E}, \mathcal{M}), F) \mapsto (\mathcal{C}, \quad)$

$$\begin{tikzcd}
\mathcal{C} ⨾ \mathcal{M}^{\mathrm{op}} \arrow[r, "F"] \arrow[d, hook] \arrow[dr, phantom, "\Downarrow\kappa"] & \mathsf{Set} \\
\mathsf{Cospan}(\mathcal{C}) \arrow[ur, dashed, "\mathsf{Lan}F"'] &
\end{tikzcd}$$

DecData is the right setting

Let CospanAlg be the full subcategory of DecData whose objects are pairs $(\mathcal{C}, F)$ for a lax monoidal functor

$$F\colon \mathsf{Cospan}(\mathcal{C}) \to \mathsf{Set}$$

$$\mathsf{Hyp} \xrightarrow{\mathsf{Alg}} \mathsf{CospanAlg} \overset{\iota}{\hookrightarrow} \mathsf{DecData}$$

Prop. [Fong,S]

There exists a functor $\mathsf{Kan}\colon\ \mathsf{DecData} \to \mathsf{CospanAlg}$, mapping $(\mathcal{C}, (\mathcal{E}, \mathcal{M}), F) \mapsto (\mathcal{C}, \mathsf{Lan}F)$.

$$\begin{tikzcd}
\mathcal{C} ⨟ \mathcal{M}^{\mathrm{op}} \arrow[r, "F"] \arrow[d, hook] & \mathsf{Set} \\
\mathsf{Cospan}(\mathcal{C}) \arrow[ur, dashed] &
\end{tikzcd}$$

$\mathsf{DecData}$ is the right setting

Thm. [Fong,S]

In fact,

$$\begin{tikzcd} \mathsf{Hyp} \arrow[r, shift left, "\mathsf{Alg}"] & \mathsf{CospanAlg} \arrow[l, shift left, "(-)\mathsf{Corel}|"] \arrow[r, hook, shift left, "\iota"] & \mathsf{DecData} \arrow[l, shift left, "\mathsf{Kan}"] \end{tikzcd}$$

DecData is the right setting

Thm. [Fong,S]

In fact, we have adjunctions

$$\begin{tikzcd} \mathsf{Hyp} \arrow[r, shift left, "\mathsf{Alg}"] \arrow[r, phantom, "\perp"] & \mathsf{CospanAlg} \arrow[l, shift left, "(-)\mathsf{Corel}|"] \arrow[r, hook, shift left, "\iota"] \arrow[r, phantom, "\top"] & \mathsf{DecData} \arrow[l, shift left, "\mathsf{Kan}"] \end{tikzcd}$$

DecData is the right setting

Thm. [Fong,S]

In fact, we have adjunctions

$$\begin{tikzcd} \mathsf{Hyp} \arrow[r, shift left, "\mathsf{Alg}"] \arrow[r, phantom, "\perp"] & \mathsf{CospanAlg} \arrow[l, shift left, "(-)\mathsf{Corel}|"] \arrow[r, hook, shift left, "\iota"] \arrow[r, phantom, "\top"] & \mathsf{DecData} \arrow[l, shift left, "\mathsf{Kan}"] \arrow[ll, bend left, "(-)\mathsf{Corel}"] \end{tikzcd}$$

DecData is the right setting

Thm. [Fong,S]

In fact, we have adjunctions

$$\begin{tikzcd}
\mathsf{Hyp} \arrow[r, shift left=1.5ex, "\mathsf{Alg}"] \arrow[r, phantom, "\perp"] & \mathsf{CospanAlg} \arrow[l, shift left=1.5ex, "(-)\mathsf{Corel}|"] \arrow[r, hook, shift left=1.5ex, "\iota"] \arrow[r, phantom, "\top"] & \mathsf{DecData} \arrow[l, shift left=1.5ex, "\mathsf{Kan}"] \arrow[ll, bend left=60, red, "(-)\mathsf{Corel}"]
\end{tikzcd}$$

This decomposition helps us prove that, given $\mathcal{H} \in \mathsf{Hyp}$,

$$\mathcal{H} \mapsto \mathsf{Alg}\mathcal{H} \mapsto \iota\mathsf{Alg}\mathcal{H} \mapsto (-)\mathsf{Corel}\iota\mathsf{Alg}\mathcal{H} \simeq \mathcal{H}$$

as hypergraph categories.

Our recipe for black box functors

$$\mathsf{Hyp} \underset{(-)\mathsf{Corel}|}{\overset{\mathsf{Alg}}{\underset{\perp}{\rightleftarrows}}} \mathsf{CospanAlg} \underset{\mathsf{Kan}}{\overset{\iota}{\underset{\top}{\rightleftarrows}}} \mathsf{DecData}$$

We have showed that every hypergraph category can be built, up to equivalence, from $\mathsf{DecData}$ via the decorated corelations construction.

Our recipe for black box functors

$$\begin{tikzcd} \mathsf{Hyp} \arrow[r, shift left=2, "\mathsf{Alg}"] \arrow[r, phantom, "\perp"] & \mathsf{CospanAlg} \arrow[l, shift left=2, "(-)\mathsf{Corel}|"] \arrow[r, shift left=2, hook, "\iota"] \arrow[r, phantom, "\top"] & \mathsf{DecData} \arrow[l, shift left=2, "\mathsf{Kan}"] \end{tikzcd}$$

We have showed that every hypergraph category can be built, up to equivalence, from $\mathsf{DecData}$ via the decorated corelations construction.

In particular, every hypergraph functor is, up to equivalence, a functor $F\mathsf{Corel} \to F'\mathsf{Corel}$ obtained from the previous recipe $(A : \mathcal{C} \to \mathcal{C}',\ \alpha : F \Rightarrow F'A)$.

Our recipe for black box functors

$$\begin{tikzcd} \mathsf{Hyp} \arrow[r, shift left, "\mathsf{Alg}"] \arrow[r, phantom, "\perp"] & \mathsf{CospanAlg} \arrow[l, shift left, "(-)\mathsf{Corel}|"] \arrow[r, hook, shift left, "\iota"] \arrow[r, phantom, "\top"] & \mathsf{DecData} \arrow[l, shift left, "\mathsf{Kan}"] \end{tikzcd}$$

We have showed that every hypergraph category can be built, up to equivalence, from DecData via the decorated corelations construction.

In particular, every hypergraph functor is, up to equivalence, a functor $F\mathsf{Corel} \to F'\mathsf{Corel}$ obtained from the previous recipe $(A : \mathcal{C} \to \mathcal{C}',\ \alpha : F \Rightarrow F'A)$.

Then, given two arbitrary hypergraph categories, we can push them all the way from Hyp to DecData, construct the black box functor in DecData where things are easier to work with, and push them back to Hyp, where we recover our original data up to equivalence.

Thank you!