# A layout algorithm for higher-dimensional string diagrams

Calin Tataru

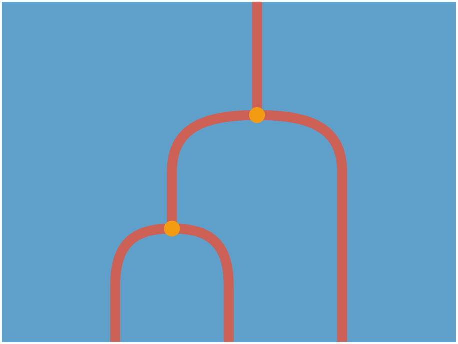
University of Cambridge

SYCO 10, 20 December 2022

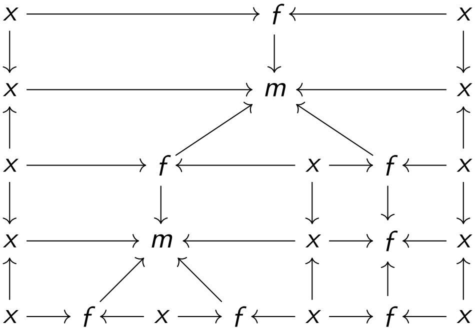
# Homotopy.io

- *Homotopy.io* is a proof assistant for higher category theory.
- It lets you build terms in finitely-presented $n$-categories.
- Terms have a direct geometrical representation.

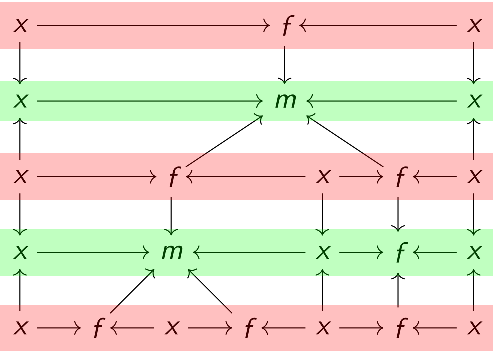
# Example of 2-diagram



# Corresponding zigzag diagram

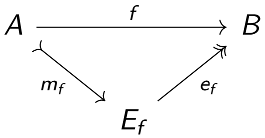


# Corresponding zigzag diagram



# Mono-epi factorization

**Poset** admits a mono-epi factorization system:




where $E_f = A \sqcup (B \setminus f[A])$, $m_f$ is the canonical inclusion, $e_f = [f, \mathrm{id}]$.

# Injectification

## Definition

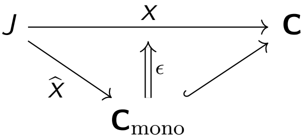
Given a diagram in a category $\mathbf{C}$,

$$X : J \to \mathbf{C}$$

an *injectification* is defined to be a diagram

$$\widehat{X} : J \to \mathbf{C}_{\mathrm{mono}}$$

equipped with a pointwise epi natural transformation



# Injectification

- We will show how to compute injectifications for finite poset-shaped diagrams $X : J \to \mathbf{Poset}$.

# Injectification

- We will show how to compute injectifications for finite poset-shaped diagrams $X : J \to \mathbf{Poset}$.
- The construction works by induction on $J$.

# Injectification

- We will show how to compute injectifications for finite poset-shaped diagrams $X : J \to \mathbf{Poset}$.
- The construction works by induction on $J$.
- For every $i \in J$, let $J{\downarrow}i = \{j \in J : j < i\}$.

# Injectification

- We will show how to compute injectifications for finite poset-shaped diagrams $X : J \to \mathbf{Poset}$.
- The construction works by induction on $J$.
- For every $i \in J$, let $J{\downarrow}i = \{j \in J : j < i\}$.
- If $J{\downarrow}i$ is empty, then define

$$\widehat{X}j := Xj \qquad \epsilon_j := \mathrm{id}_{Xj}$$

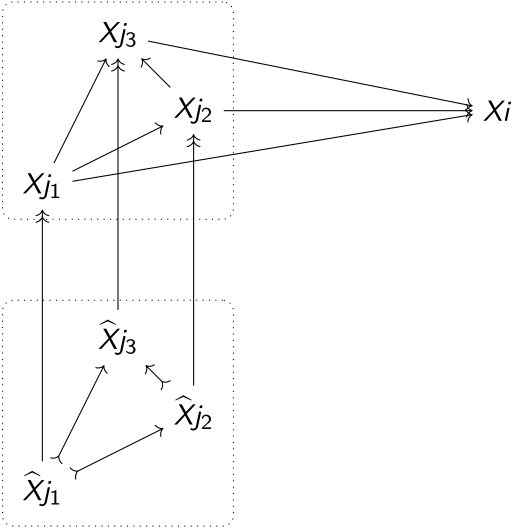
# Injectification

- We will show how to compute injectifications for finite poset-shaped diagrams $X : J \to \mathbf{Poset}$.
- The construction works by induction on $J$.
- For every $i \in J$, let $J{\downarrow}i = \{j \in J : j < i\}$.
- If $J{\downarrow}i$ is empty, then define

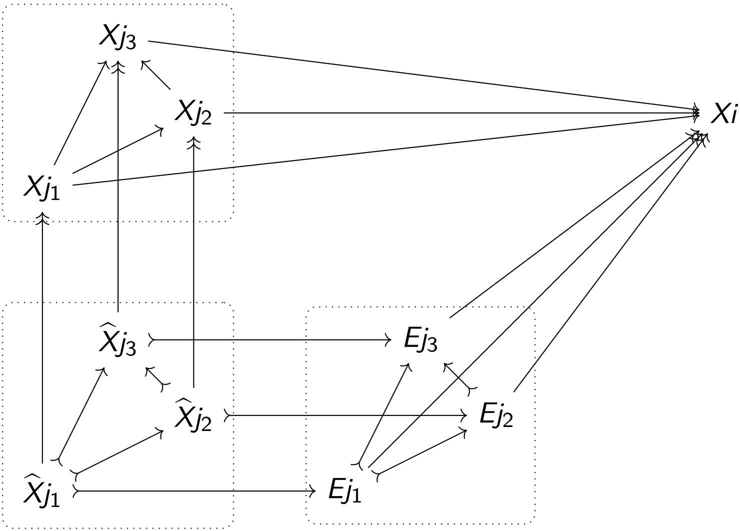
$$\widehat{X}j := Xj \qquad \epsilon_j := \mathrm{id}_{Xj}$$

- If not, $\widehat{X}$ and $\epsilon$ must already be defined on $J{\downarrow}i$.

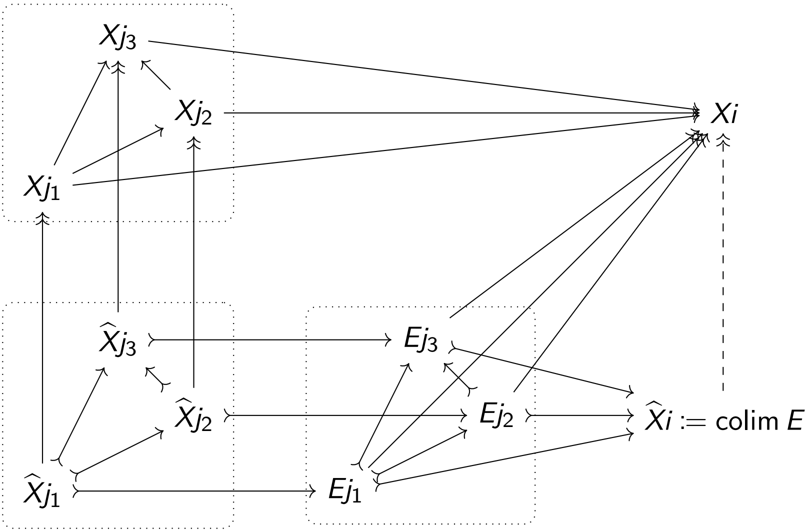
# Injectification



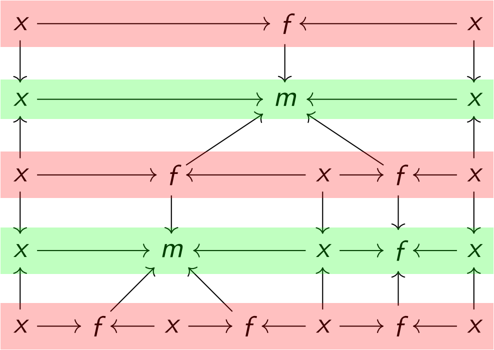
# Injectification



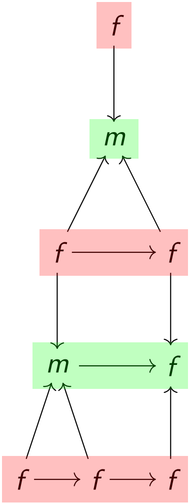
# Injectification



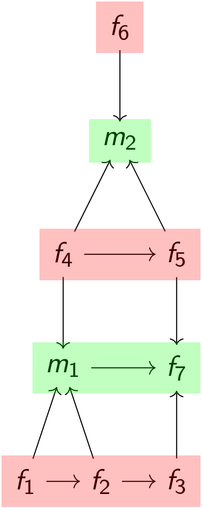
# Layout



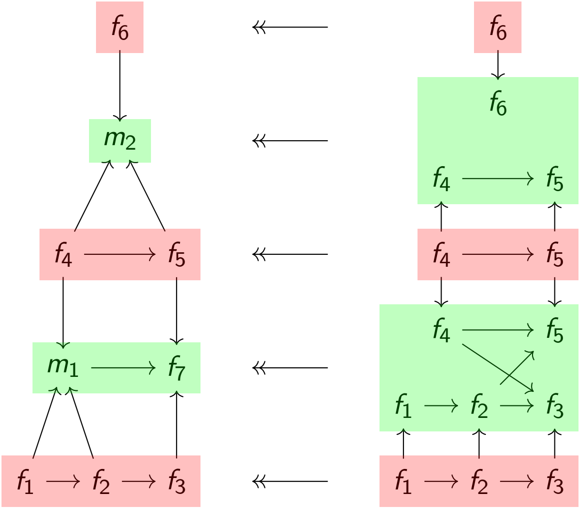
# Layout



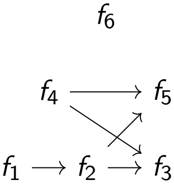
# Layout



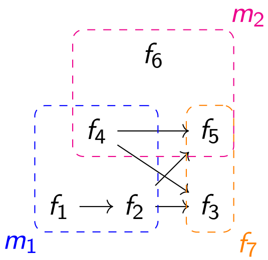
# Layout



# Layout



# Layout



# Distance constraints

$$f_2 - f_1 \geq 1$$
$$f_3 - f_2 \geq 1$$
$$f_3 - f_4 \geq 1$$
$$f_5 - f_2 \geq 1$$
$$f_5 - f_4 \geq 1$$

# Fair averaging constraints (strict)

$$\begin{aligned}
\tfrac{1}{2}(f_1 + f_2) - f_4 &= 0 \\
f_3 - f_5 &= 0 \\
\tfrac{1}{2}(f_4 + f_5) - f_6 &= 0
\end{aligned}$$

# Fair averaging constraints (weak)

$$\begin{array}{ll} \text{minimize} & c_1 + c_2 + c_3 \\ \text{subject to} & \left|\frac{1}{2}(f_1 + f_2) - f_4\right| \leq c_1 \\ & |f_3 - f_5| \leq c_2 \\ & \left|\frac{1}{2}(f_4 + f_5) - f_6\right| \leq c_3 \end{array}$$

# New version of the tool

- Available at `beta.homotopy.io`

# New version of the tool

- Available at `beta.homotopy.io`
- **New layout algorithm**

# New version of the tool

- Available at `beta.homotopy.io`
- **New layout algorithm**
  - Re-engineered 3D renderer

# New version of the tool

- Available at `beta.homotopy.io`
- **New layout algorithm**
  - Re-engineered 3D renderer
  - 4D renderer (i.e. smooth movies of 3D diagrams)

# New version of the tool

- Available at `beta.homotopy.io`
- **New layout algorithm**
  - Re-engineered 3D renderer
  - 4D renderer (i.e. smooth movies of 3D diagrams)
- Invertible generators

# New version of the tool

- Available at `beta.homotopy.io`
- **New layout algorithm**
  - Re-engineered 3D renderer
  - 4D renderer (i.e. smooth movies of 3D diagrams)
- Invertible generators
- Oriented manifold diagrams

# New version of the tool

- Available at `beta.homotopy.io`
- **New layout algorithm**
  - Re-engineered 3D renderer
  - 4D renderer (i.e. smooth movies of 3D diagrams)
- Invertible generators
- Oriented manifold diagrams
- TikZ export (finally!)