Applied Compositional Thinking for Engineers

Spring 2021

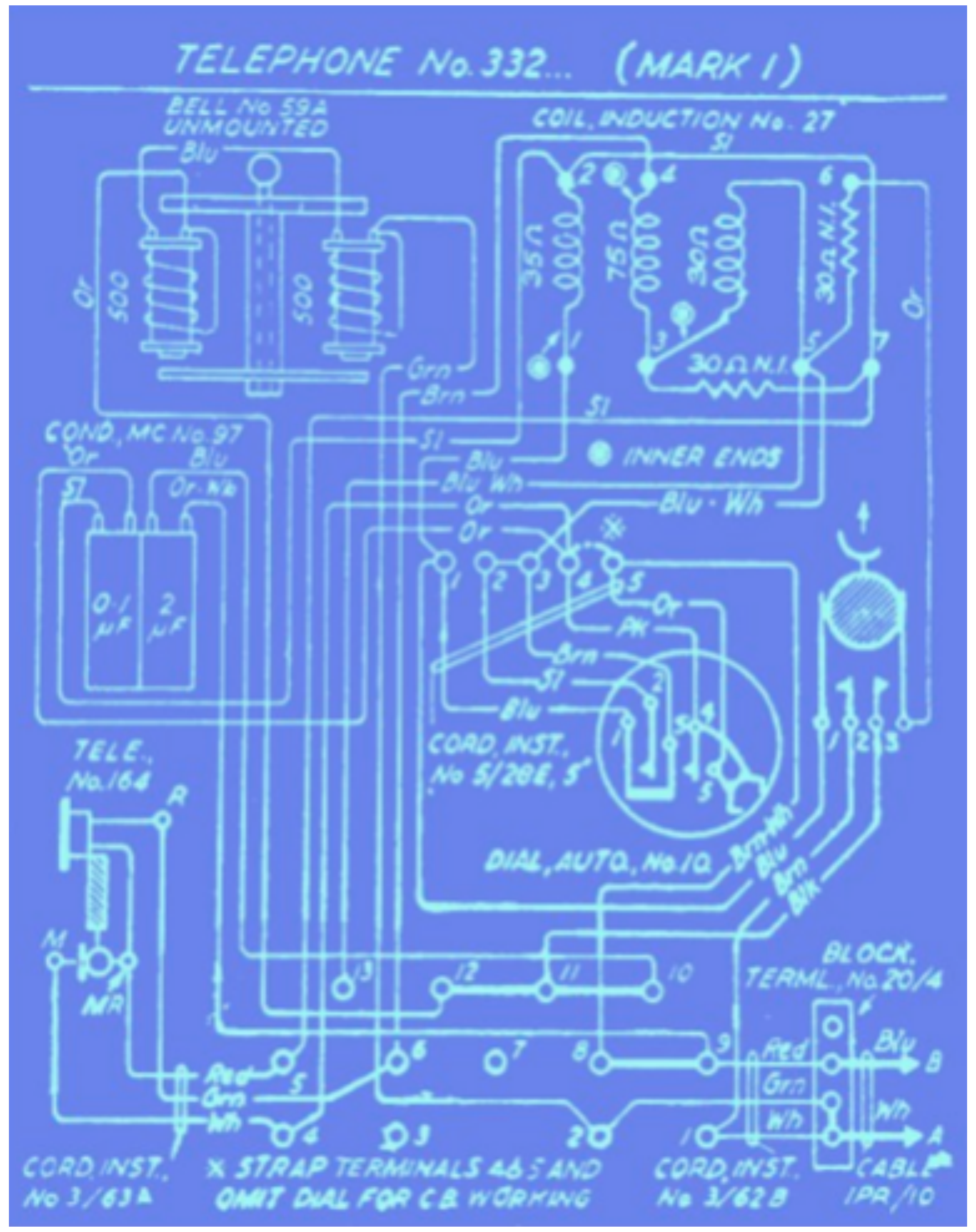
Operads
(continued)

Plan

- Operads as blueprints
- Algebras for an operad
- More examples

Operads as blueprints

Sets Operad



Operad

Sets Operad

Sets Operad

operads as a way to formally describe hierarchies of structures

"mathematical language for modular systems" - D. Spivak

Functors between operads

Let $\mathcal{O}$ and $\mathcal{O}'$ be operads. A <u>functor</u> $F: \mathcal{O} \to \mathcal{O}'$ is defined by:

<u>constituents</u>:
- $F_0 : ob(\mathcal{O}) \longrightarrow ob(\mathcal{O}')$
- $F_1 : \mathcal{O}([x_1, \dots, x_n]; y) \to \mathcal{O}'([F_0 x_1, \dots, F_0 x_n]; Fy)$

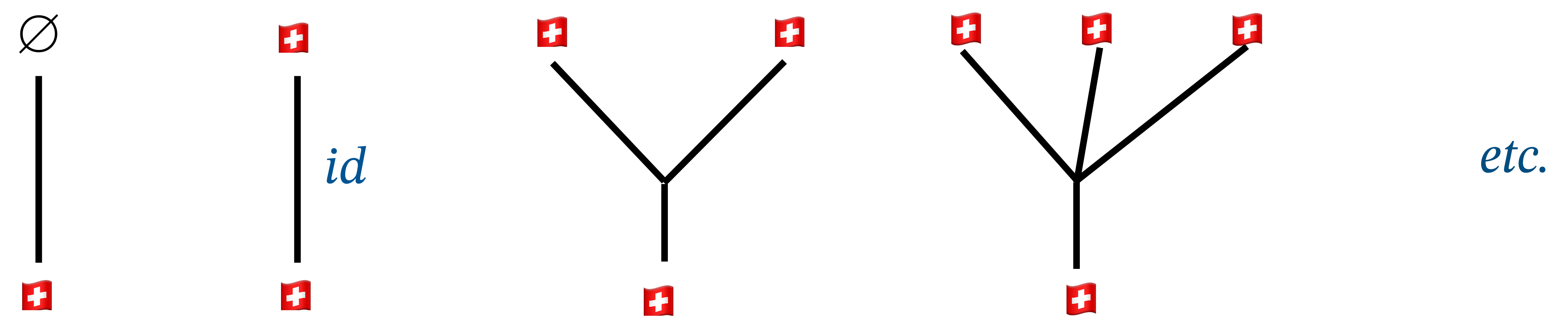
Functors between operads

Let $\mathcal{O}$ and $\mathcal{O}'$ be operads. A <u>functor</u> $F: \mathcal{O} \to \mathcal{O}'$ is defined by:

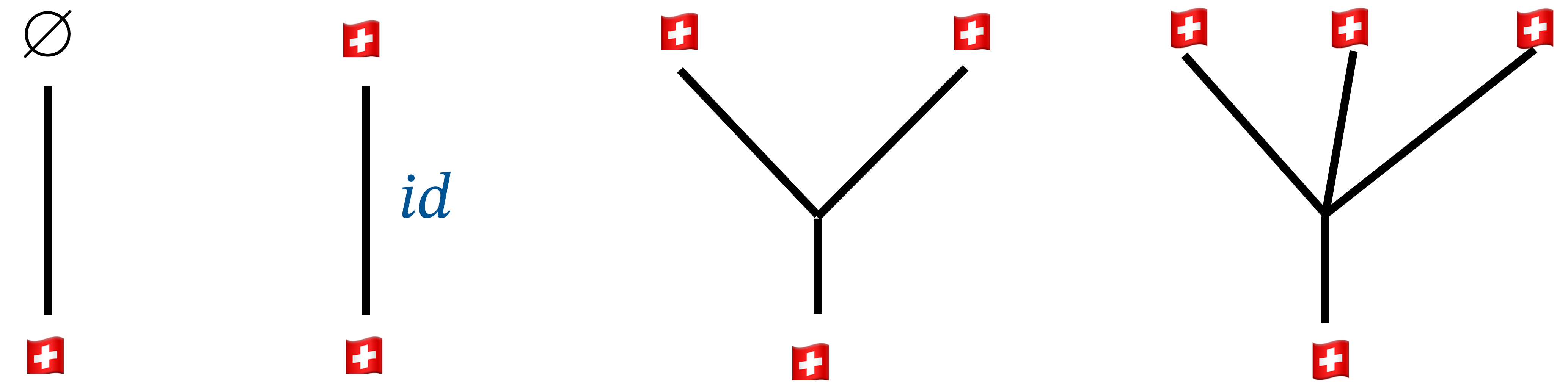
<u>constituents</u>:
- $F_0 : ob(\mathcal{O}) \longrightarrow ob(\mathcal{O}')$
- $F_1 : \mathcal{O}([x_1, \ldots, x_n]; y) \to \mathcal{O}'([F_0 x_1, \ldots, F_0 x_n]; Fy)$

<u>conditions</u>:
- compatibility with composition operations
- compatibility with identities

Example: single typed branches



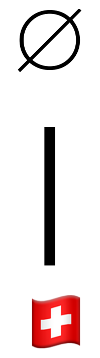
Example: single typed branches



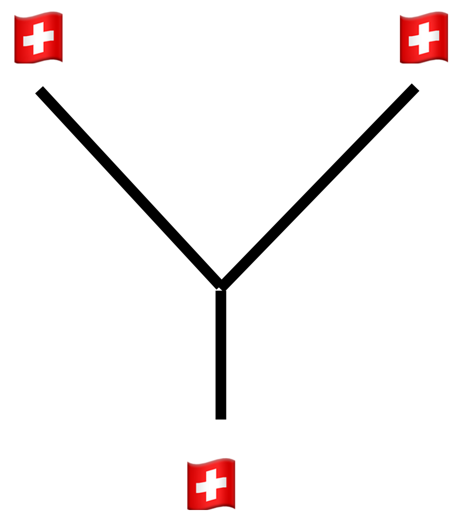
etc.

Objects: $\mathrm{ob}(\mathcal{O}) = \{$ 🇨🇭 $\}$ where $0 = [\,]$, $1 = [$🇨🇭$]$, $2 = [$🇨🇭, 🇨🇭$]$, etc.

Morphisms: $\mathcal{O}([$🇨🇭$,\dots,$🇨🇭$];$🇨🇭$) = \{\text{tree with } n \text{ branches}\} =: \mathcal{O}(n)$

$\mathcal{O}(0) =$ 

$\mathcal{O}(1) =$ 

$\mathcal{O}(2) =$ 

etc.

Example: single typed branches

Recall: the operad of sets:

Objects: all sets

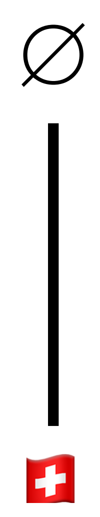
Morphisms: $\mathrm{Set}([X_1, X_2, \dots, X_n]; Y) = \{\text{functions } X_1 \times \cdots \times X_n \to Y\}$

What is a functor $\mathcal{O} \xrightarrow{F} \mathrm{Set}$?

$ob(\mathcal{O}) = \{$🇨🇭$\} \xrightarrow{F} ob(\mathrm{Set}) =$ all sets

🇨🇭 $\longmapsto F($🇨🇭$) =: X$

Example: single typed branches



$\xrightarrow{F}$

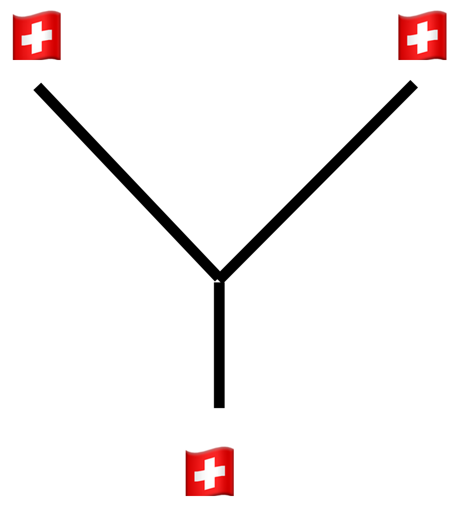
$$\begin{tikzcd} \{*\} \arrow[d, "j"] \\ X \end{tikzcd}$$

j chooses an element of X.

Set $1 := j(*)$

$\xrightarrow{F}$

$$\begin{tikzcd} X \arrow[d, "id"] \\ X \end{tikzcd}$$

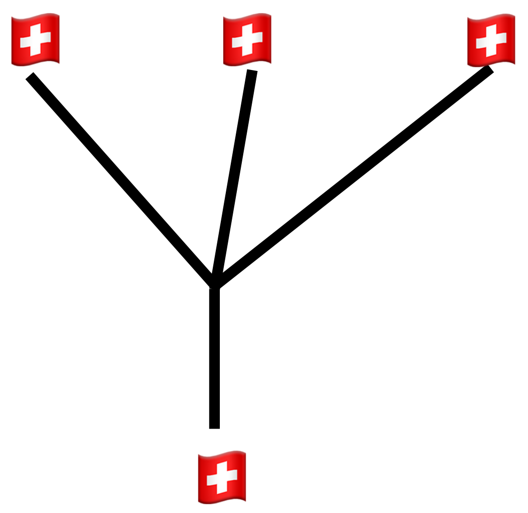


$\xrightarrow{F}$

$$\begin{tikzcd} X \times X \arrow[d, "m"] \\ X \end{tikzcd}$$

multiplication

$(a, b) \longmapsto a * b$



$\xrightarrow{F}$

$$\begin{tikzcd} X \times X \times X \arrow[d] \\ X \end{tikzcd}$$

multiplication

$(a, b, c) \longmapsto a * b * c$

Example: single typed branches

A functor $\mathcal{O} \xrightarrow{F} \mathrm{Set}$ is the "same thing"

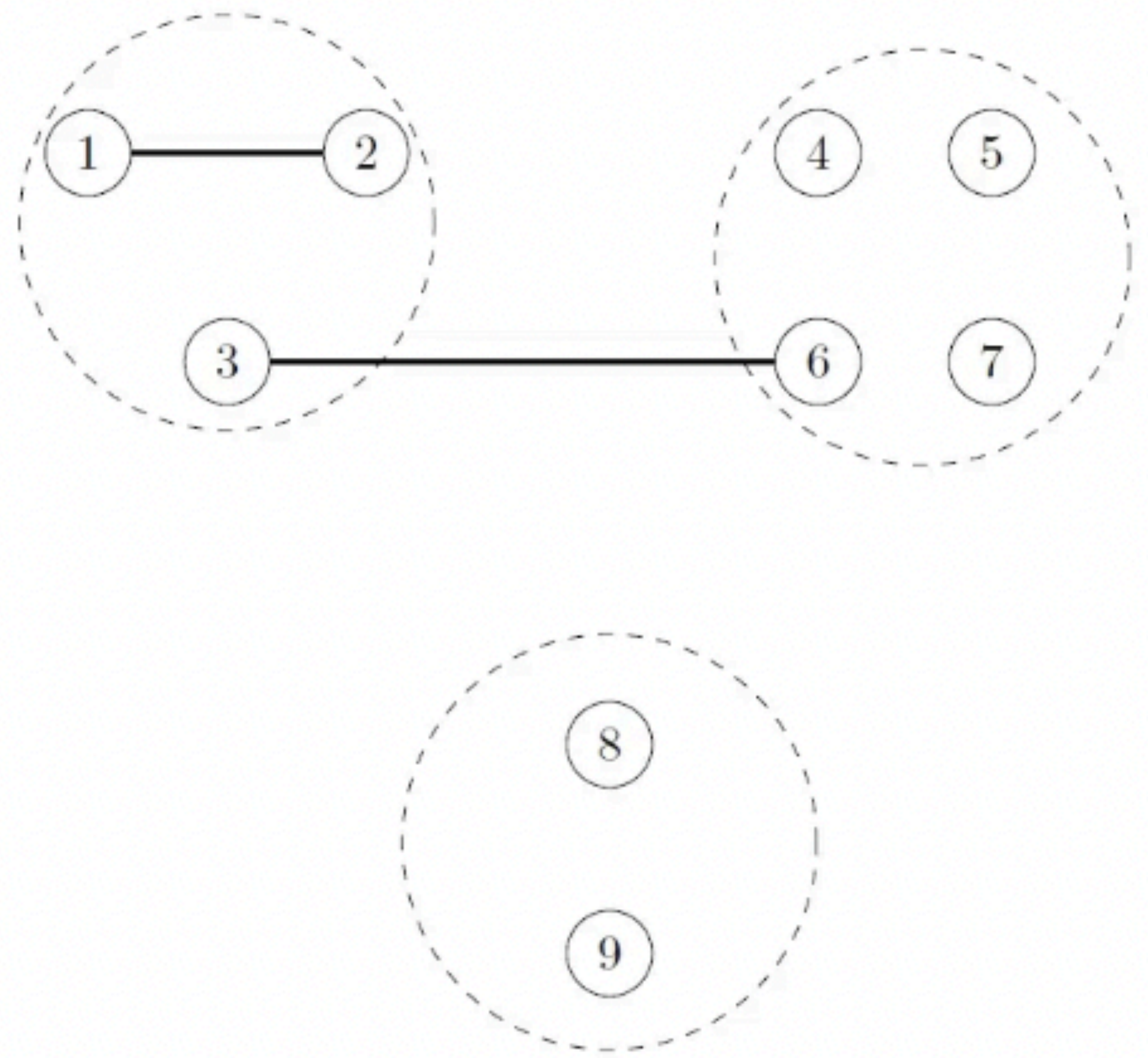
as a monoid $(X, *, 1)$!

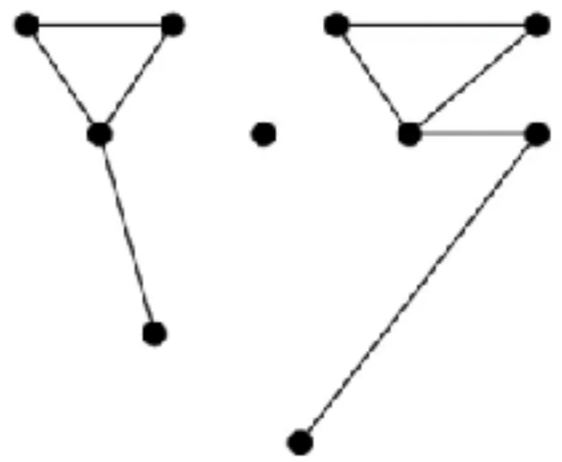
Network operads

Objects: $\mathbb{N} = \{0, 1, 2, 3, \dots\}$

"Network operad of simple graphs SG "

Morphisms:



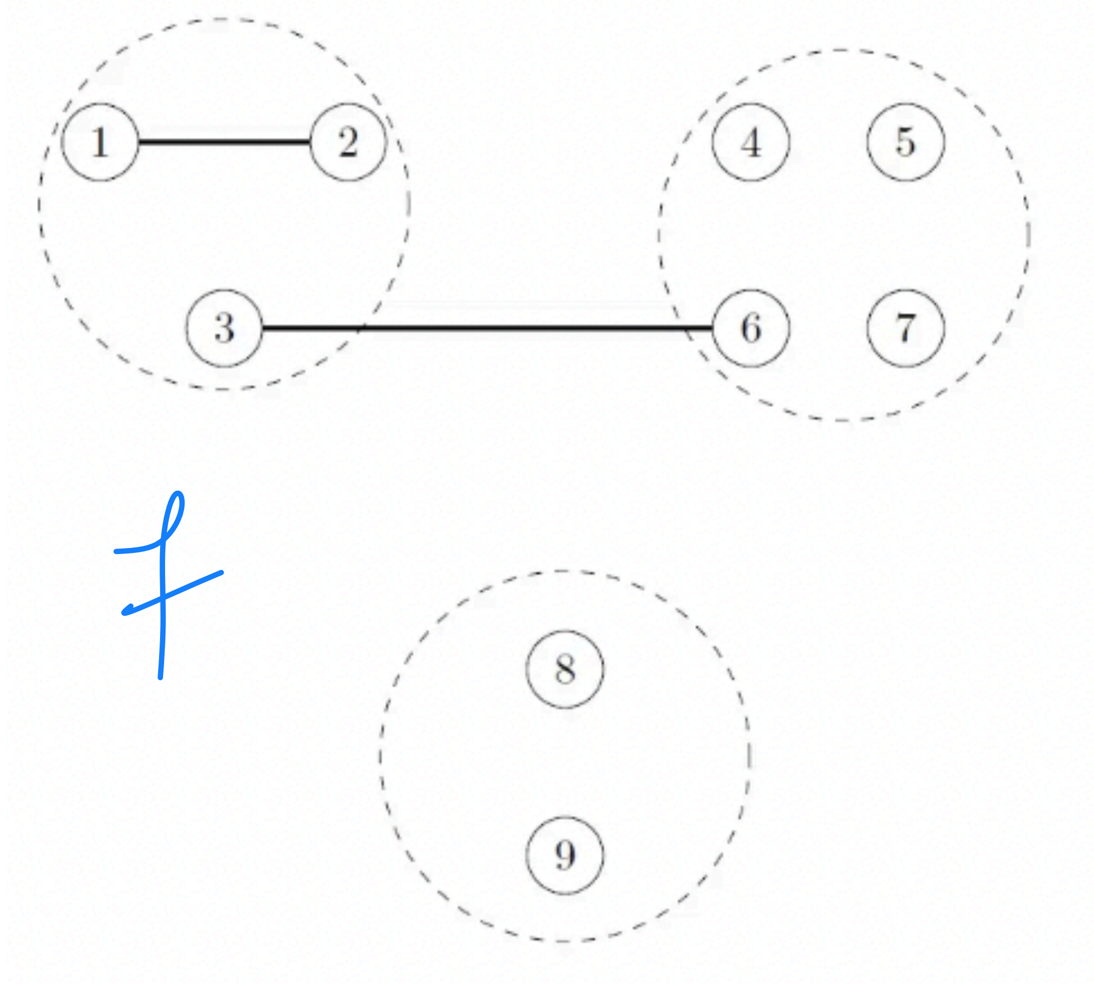
Simple graphs 

Network operads

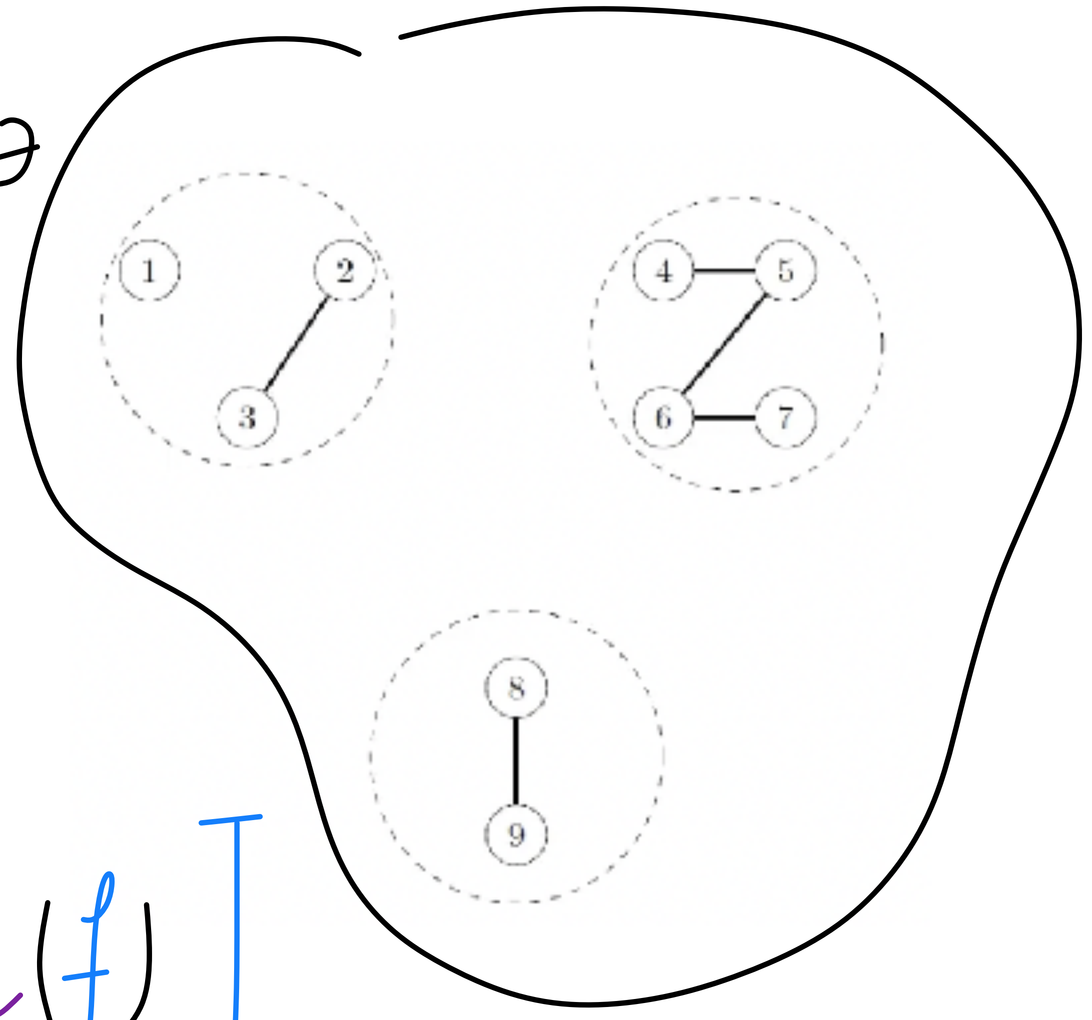
$\mathcal{SG} \xrightarrow{\mathcal{A}} \mathsf{Sets}$

$\mathcal{A} : \mathrm{ob}(\mathcal{SG}) \longrightarrow \mathsf{Sets}$

$n \longmapsto$ set of simple graphs with n vertices

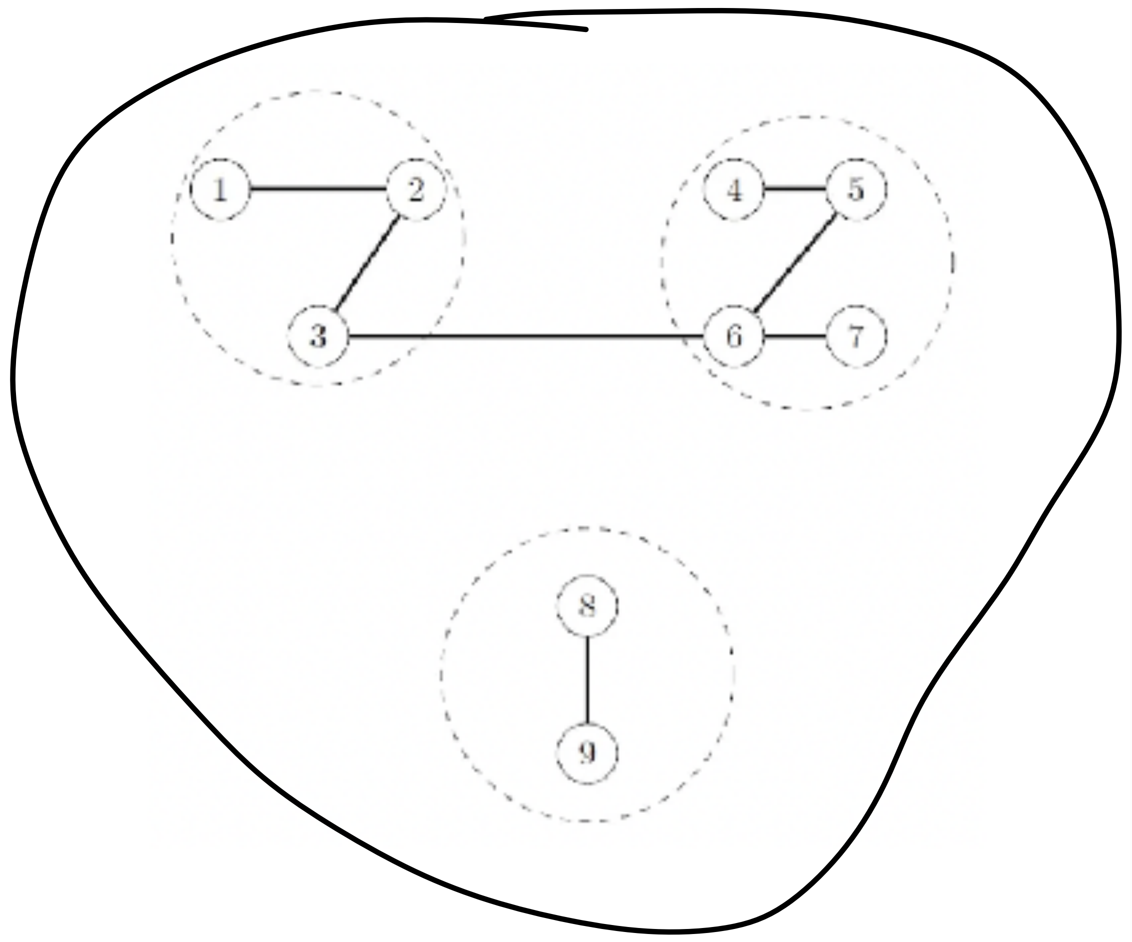


$\mathcal{A}(3) \times \mathcal{A}(4) \times \mathcal{A}(2) \ni$



$\mathcal{A}(f)$

$\mathcal{A}(9) \ni$

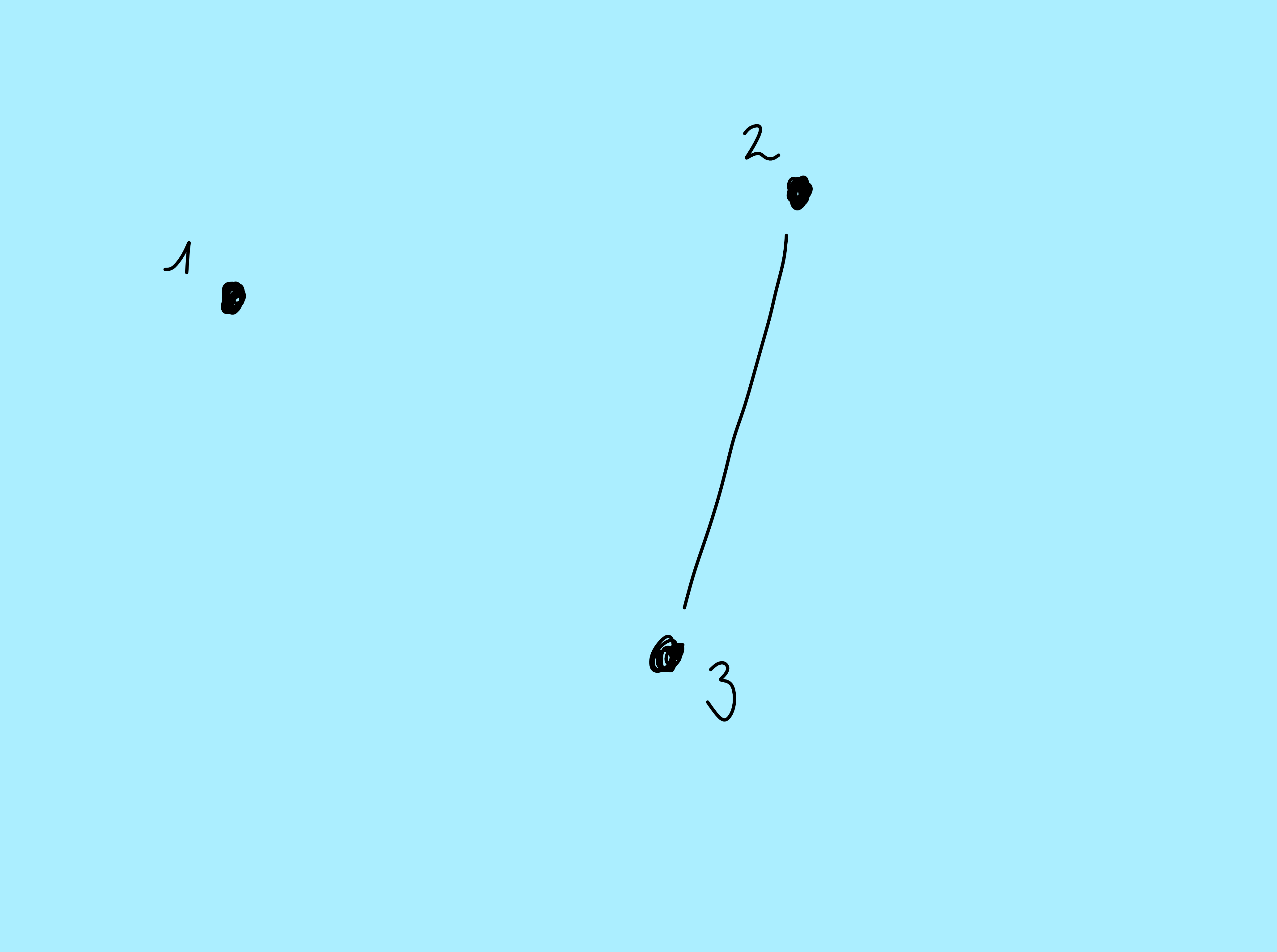


Network operads

$$SG \xrightarrow{A'} Sets$$

$$A' : \begin{array}{l} ob(SG) \longrightarrow Sets \\ n \longmapsto \text{set of simple graphs with } n \text{ vertices given by } n \text{ distinct points on a map} \end{array}$$

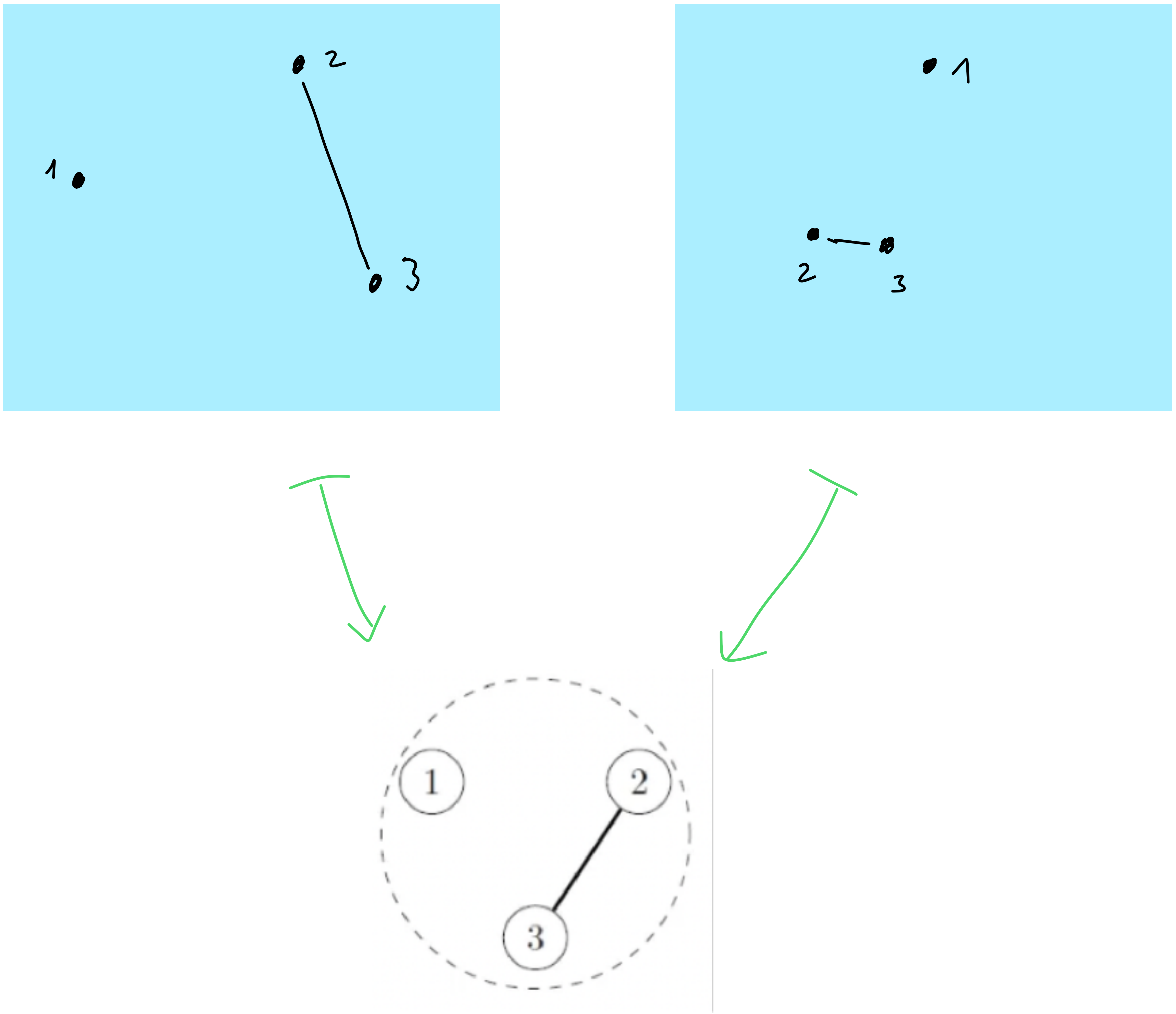
$A(3) \ni$



Network operads

$\mathcal{A}'$ is more "fine grained" than $\mathcal{A}$:

$\mathcal{A}'(3) \ni$



simplify

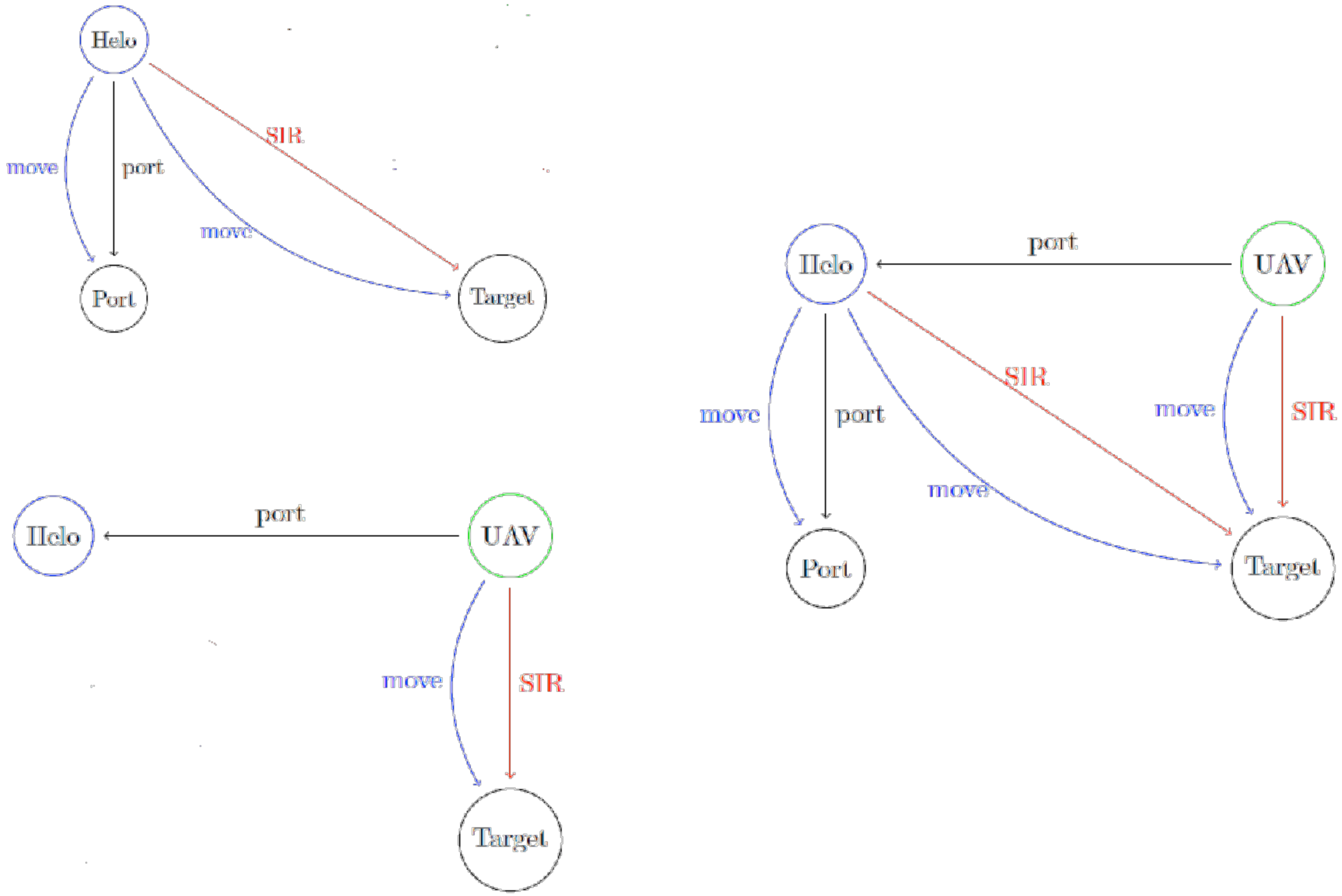
$\mathcal{A}(3) \ni$

Network operads

We have a transformation of algebras of the operad SG:

$$\begin{tikzcd}
 & \mathrm{Sets} \\
SG \arrow[ur, "A'"] \arrow[dr, "A"'] \arrow[r, phantom, "\Downarrow \text{``simplify''}"] & {} \\
 & \mathrm{Sets}
\end{tikzcd}$$

Application: search and rescue



Network Models

John C. Baez, John Foley, Joe Moeller, Blake S. Pollard

Recall: actions

Let $(M, *, 1)$ be a monoid. An <u>action</u> of M on a set X is a morphism of monoids

$$M \xrightarrow{\alpha} \mathrm{End}(X)$$

$$m \longmapsto \big(\alpha(m) : X \to X\big)$$

This is equivalent to:

$$\begin{cases} M \times X \longrightarrow X \\ (m, x) \longmapsto \alpha(m)(x) \end{cases} + \text{ axioms}$$

Recall: actions

Let $\mathcal{C}$ be a category. An <u>action</u> of $\mathcal{C}$ is a functor $A : \mathcal{C} \to \mathbf{Set}$

This generalizes $\alpha : M \to \mathrm{End}(X)$!

Algebras for an operad

<u>Definition</u>: Let $\mathcal{O}$ be an operad.

An <u>algebra</u> for $\mathcal{O}$ is a functor (of operads)

$$\mathcal{O} \xrightarrow{A} \mathrm{Sets}$$

Algebras for an operad

<u>Definition</u>: Let $\mathcal{O}$ be an operad.

An <u>algebra</u> for $\mathcal{O}$ is a functor (of operads)

$$\mathcal{O} \xrightarrow{A} \mathrm{Sets}$$

<u>Intuition</u>: An algebra A for $\mathcal{O}$ is a "concrete implementation" of $\mathcal{O}$

Sets as algebras

Let $\mathcal{O}$ be the following operad

Objects: $\{*\}$

Morphisms: Notation: $\mathcal{O}(n) := \mathcal{O}([*,\dots,*], *)$

$\mathcal{O}(0) = \emptyset$

$\mathcal{O}(1) = \{*\}$

$\mathcal{O}(n) = \emptyset \quad \forall n > 1$

An algebra $A : \mathcal{O} \to \mathrm{Sets}$ is the same thing as a choice of a set X $\quad (X = A(*))$

Semigroups as algebras

Let $\mathcal{O}$ be the following operad

Objects: $\{*\}$

Morphisms: Notation; $\mathcal{O}(n) := \mathcal{O}([*,\dots,*],*)$

$\mathcal{O}(0) = \emptyset$

$\mathcal{O}(n) = \{\star\} \quad \forall n \geq 1$

An algebra $A : \mathcal{O} \to \mathrm{Sets}$ is the same thing as

a choice of a semigroup $\langle S, \mu \rangle$

$S = A(*)$

$\mu : S \times S \longrightarrow S$ is $A(\star) : A(*) \times A(*) \longrightarrow A(*)$

Monoid actions an algebras

Let $\mathcal{O}$ be the following operad (fix a monoid M)

Objects: $\{*\}$

Morphisms: Notation ; $\mathcal{O}(n) := \mathcal{O}([*, \dots, *], *)$

$\mathcal{O}(n) = M$ if $n = 1$, $\emptyset$ else

An algebra $A : \mathcal{O} \to Sets$ is the same thing as a choice of a set $X = A(*)$ together with an action $M \to End(X)$.

Cospan operad

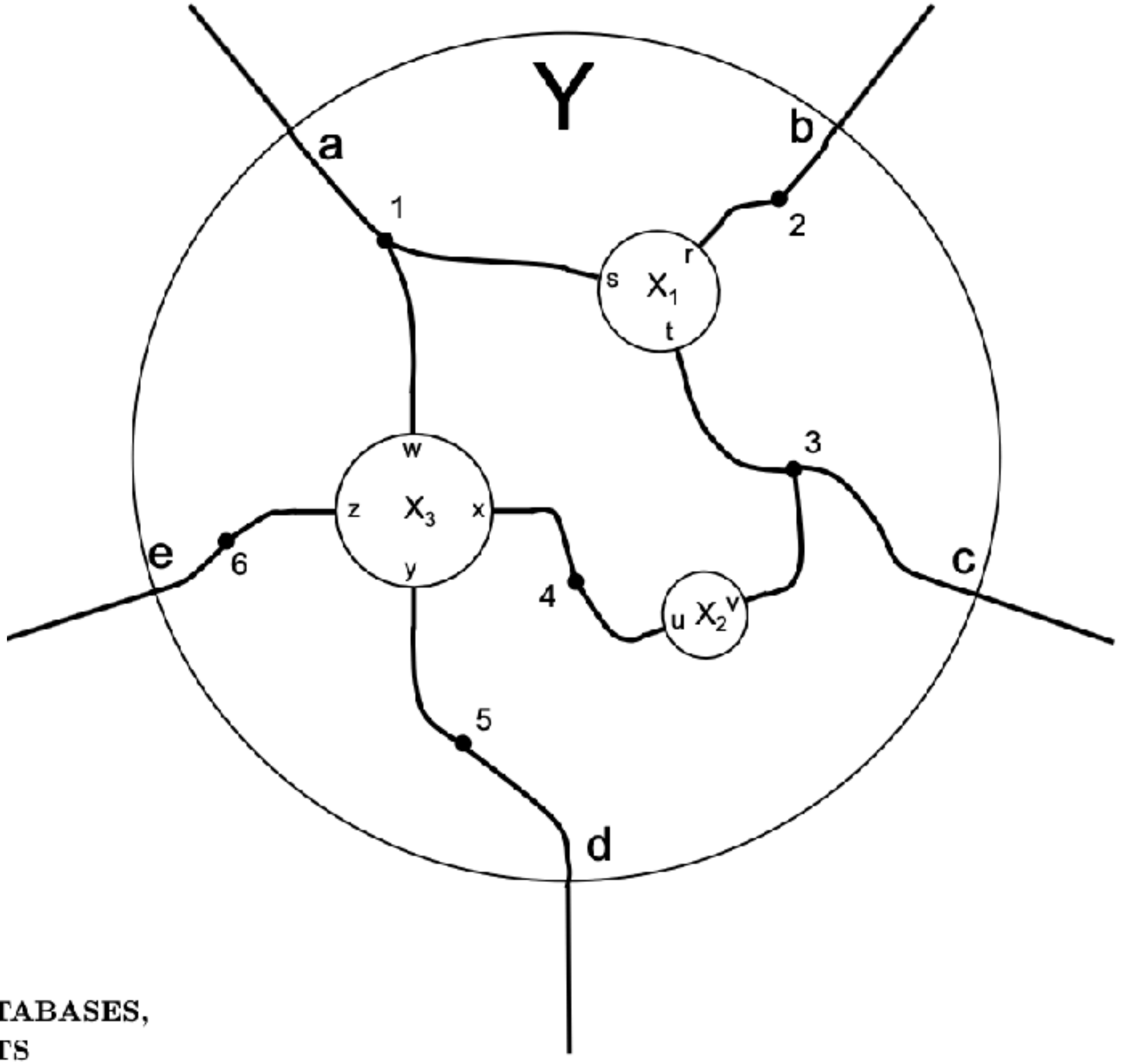
Objects: finite sets

Morphisms:

$[X_1, X_2, X_3] \xrightarrow{\phi} Y$

is

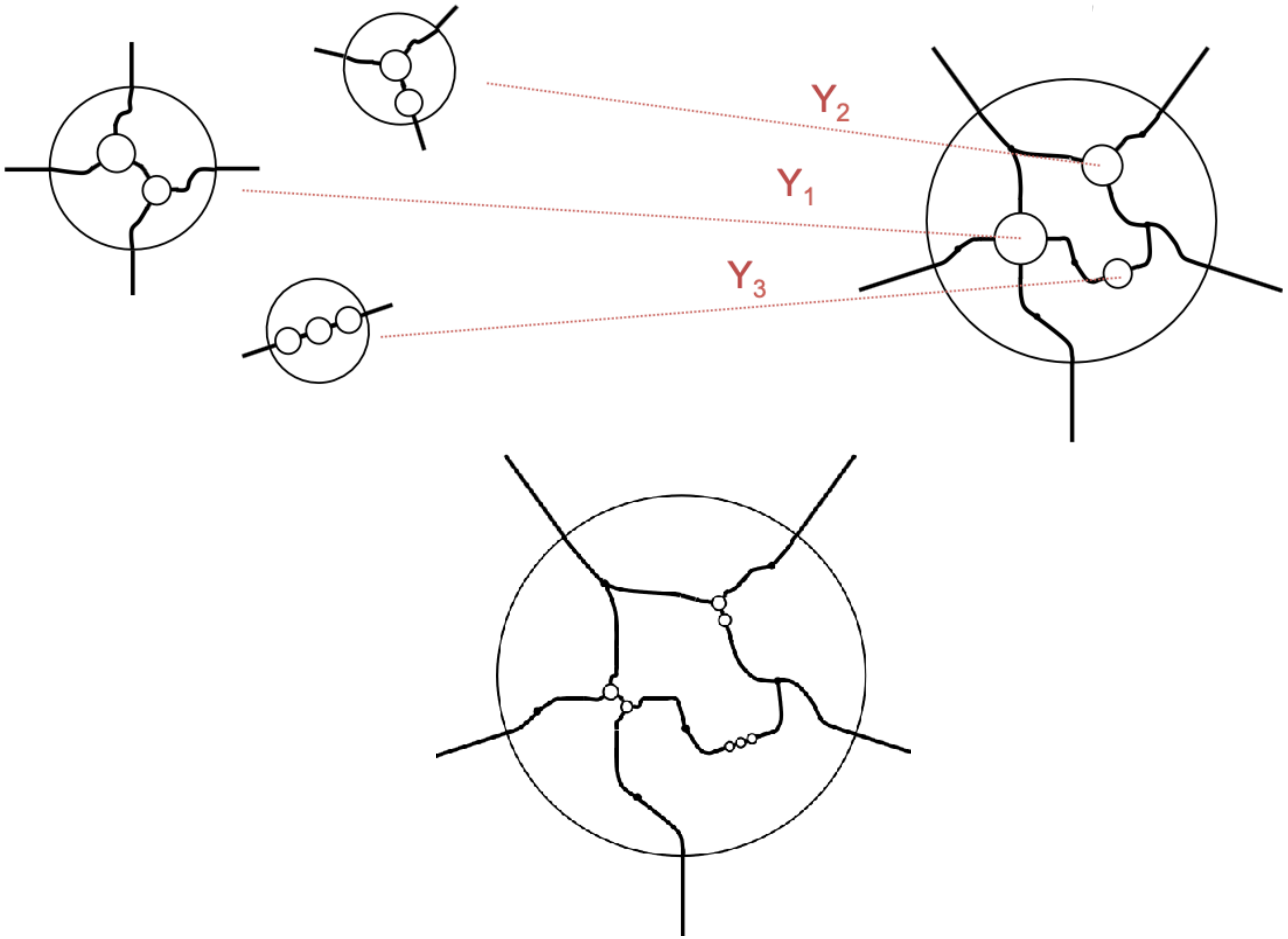
$X_1 \sqcup X_2 \sqcup X_3 \to C \leftarrow Y$



THE OPERAD OF WIRING DIAGRAMS: FORMALIZING A GRAPHICAL LANGUAGE FOR DATABASES, RECURSION, AND PLUG-AND-PLAY CIRCUITS

DAVID I. SPIVAK

Cospan operad

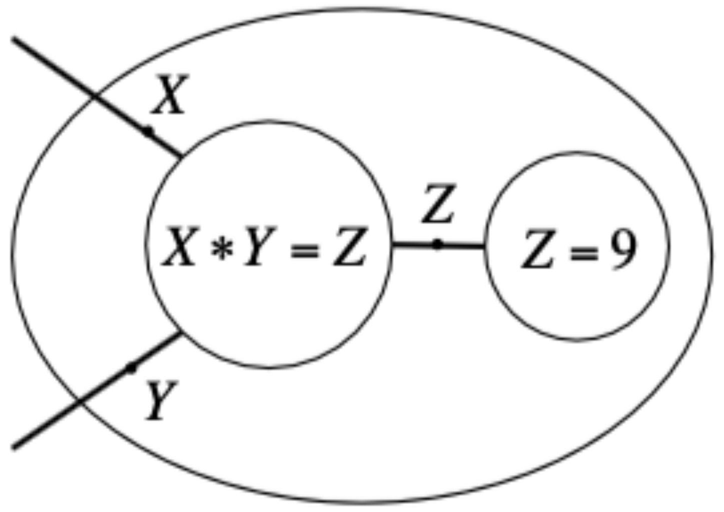


Cospan operad

$Rel_{\mathbb{Z}} : Cospan \longrightarrow Sets$

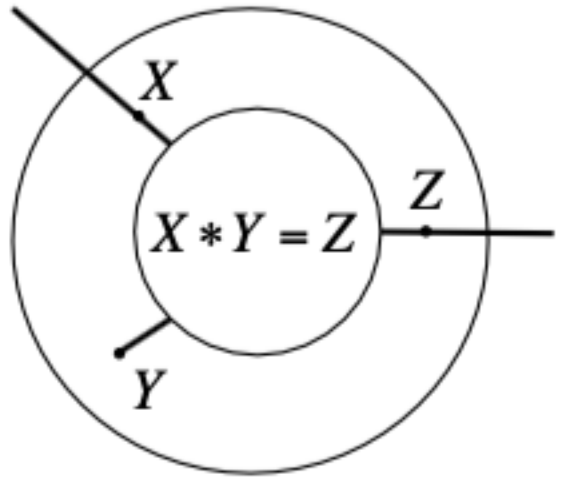
$Rel_{\mathbb{Z}}(X) = \mathcal{P}(\mathbb{Z}^X)$ = subsets of $\{(n_1, \dots, n_{|X|}) \mid n_i \in \mathbb{Z}\}$

$Rel_{\mathbb{Z}}(X_1 \sqcup \dots \sqcup X_n \longrightarrow X \leftarrow Y) : \mathcal{P}(\mathbb{Z}^{X_1}) \times \dots \times \mathcal{P}(\mathbb{Z}^{X_n}) \to \mathcal{P}(\mathbb{Z}^Y)$



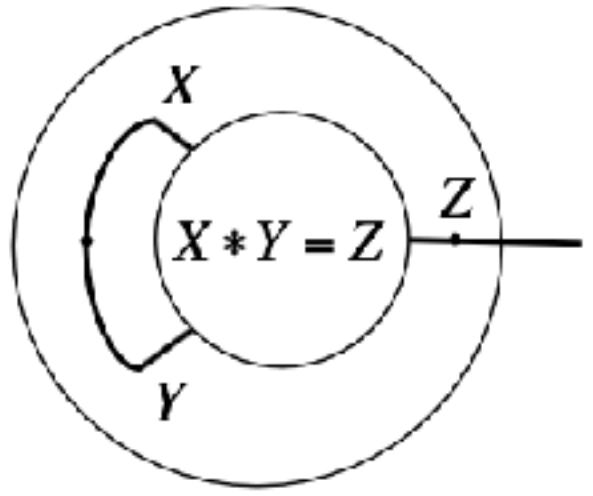
"all pairs of integers whose product is 9"

$$\begin{tikzcd} & \{X, Y\} \arrow[d, "g_1"] \\ \{X, Y, Z\} \amalg \{Z\} \arrow[r, "f_1"'] & \{X, Y, Z\} \end{tikzcd}$$



"all pairs of integers in which one is divisible by the other."

$$\begin{tikzcd} & \{X, Z\} \arrow[d, "g_2"] \\ \{X, Y, Z\} \arrow[r, "f_2"'] & \{X, Y, Z\} \end{tikzcd}$$

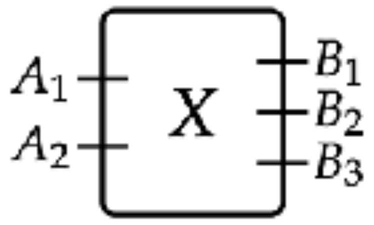


"all perfect squares"

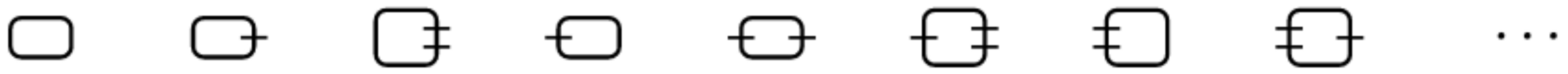
$$\begin{tikzcd} & \{Z\} \arrow[d, "g_3"] \\ \{X, Y, Z\} \arrow[r, "f_3"'] & \{XY, Z\} \end{tikzcd}$$

Wiring diagram operads

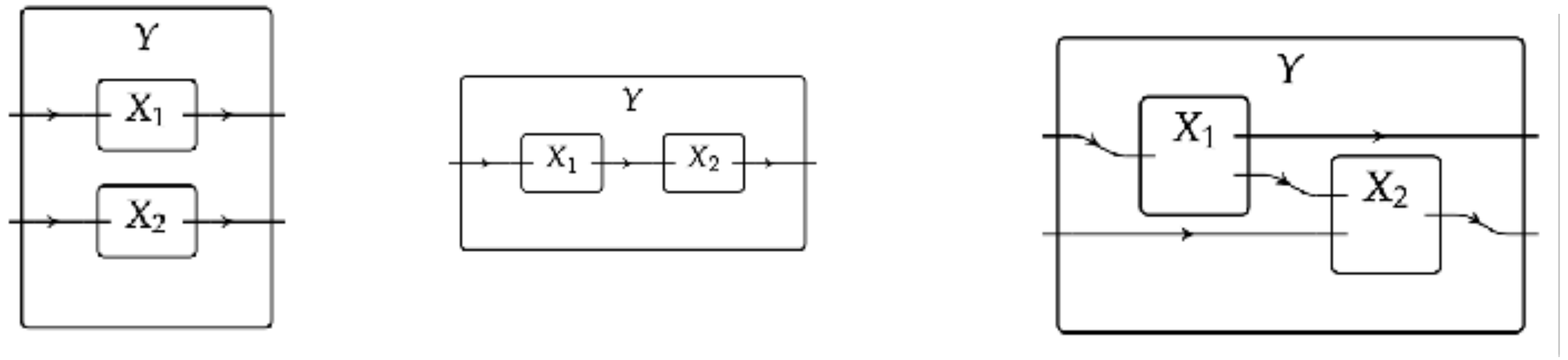
Objects:



$X^{\mathrm{in}} = \langle A_1, A_2 \rangle$ $\qquad$ $X^{\mathrm{out}} = \langle B_1, B_2, B_3 \rangle$



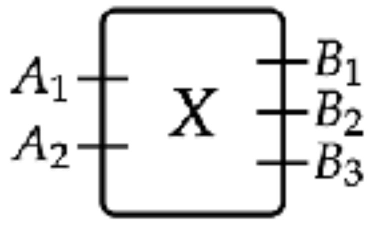
Morphisms:



$\cdots$

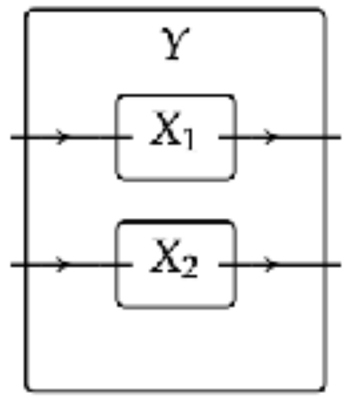
Wiring diagram operads

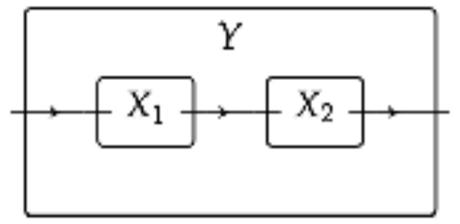
Objects:

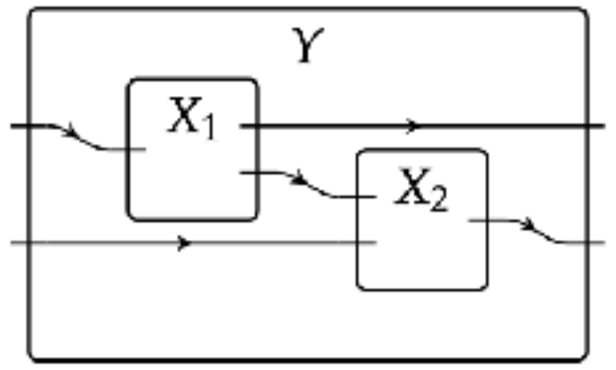


$$X^{\text{in}} = \langle A_1, A_2 \rangle$$
$$X^{\text{out}} = \langle B_1, B_2, B_3 \rangle$$

Morphisms:

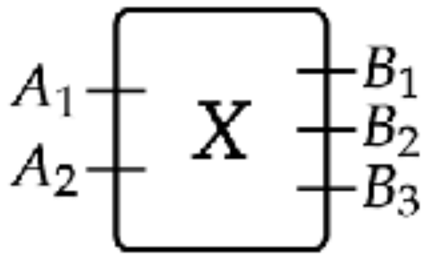






Wiring diagrams with feedback

Objects:



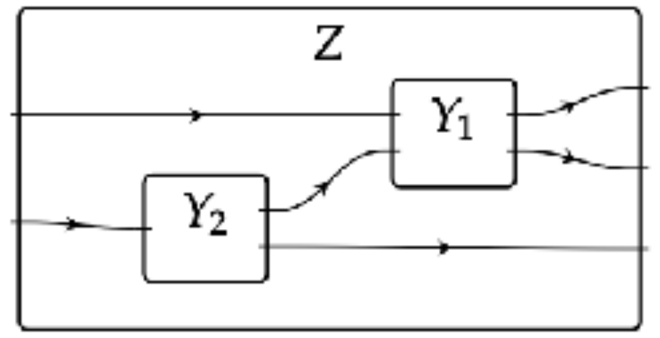
$$X = (X^{\text{in}}, X^{\text{out}})$$

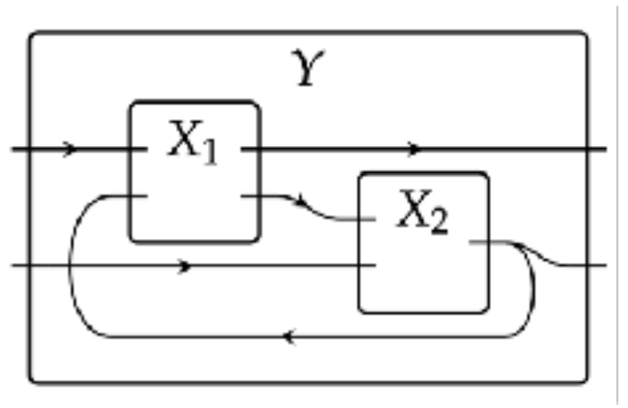
$$X^{\text{in}} = \langle A_1, A_2 \rangle$$
$$X^{\text{out}} = \langle B_1, B_2, B_3 \rangle$$

Morphisms:

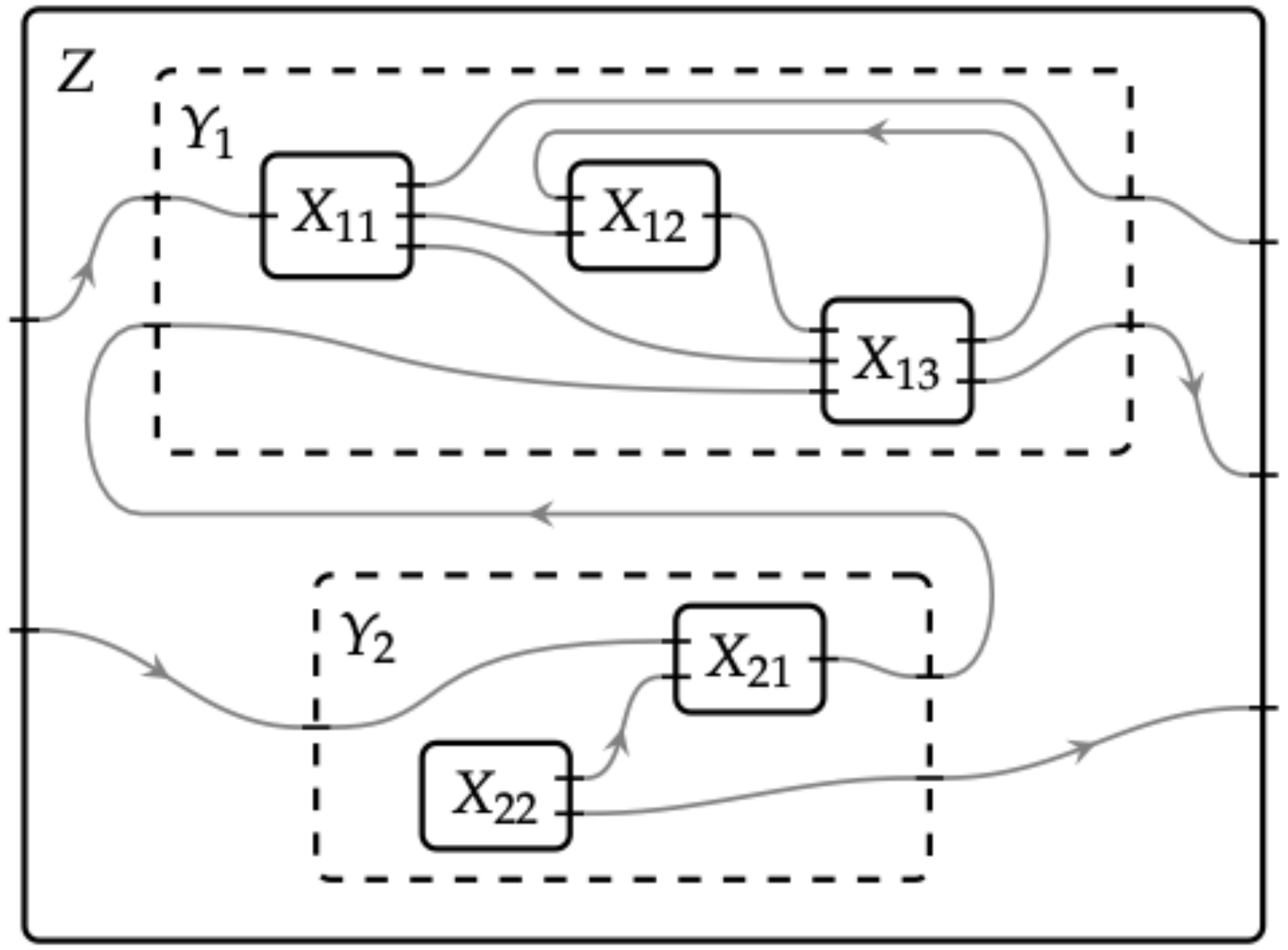
$$\varphi\colon X \to Y$$

$$\varphi^{\text{in}}\colon X^{\text{in}} \to Y^{\text{in}} + X^{\text{out}}$$
$$\varphi^{\text{out}}\colon Y^{\text{out}} \to X^{\text{out}}$$

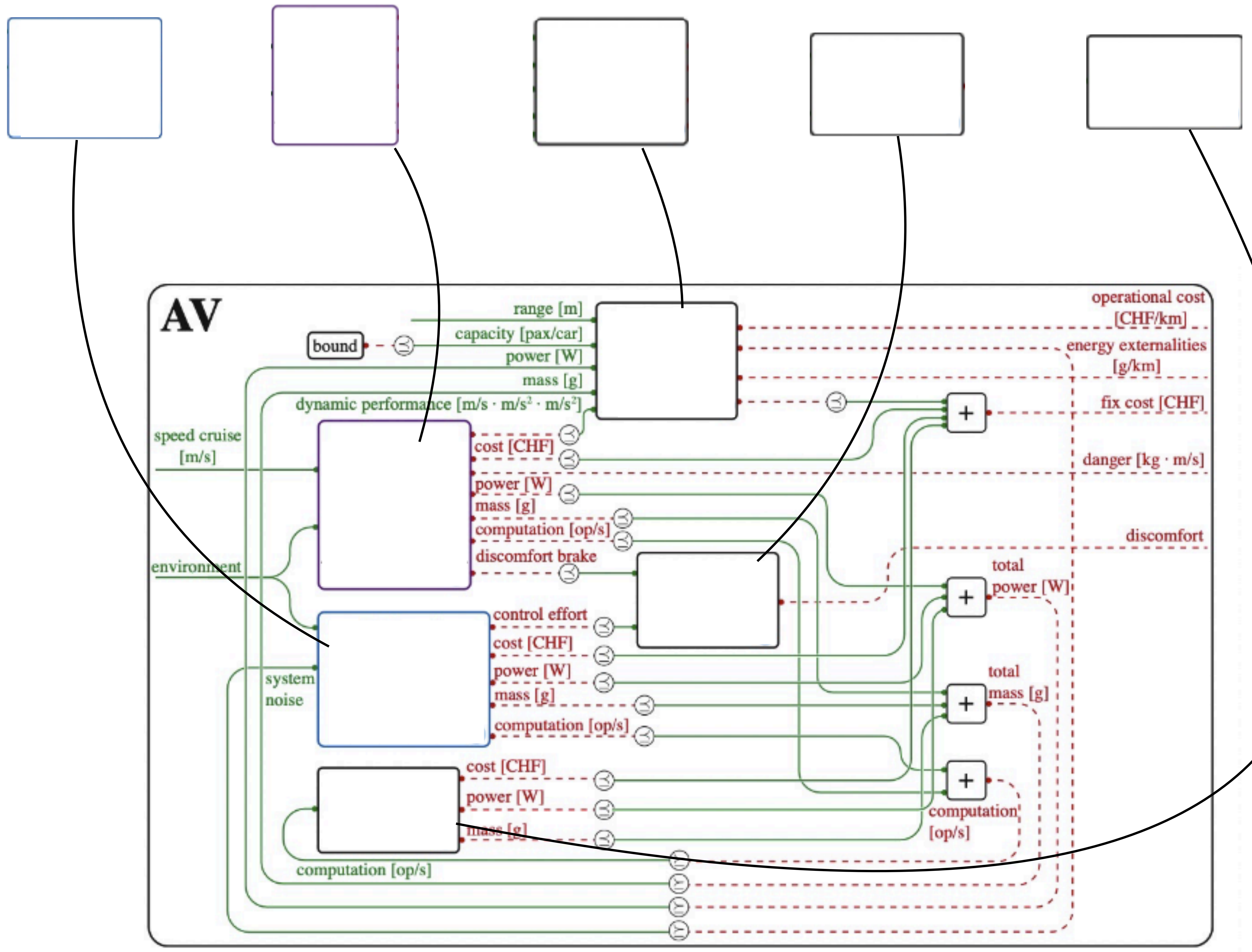




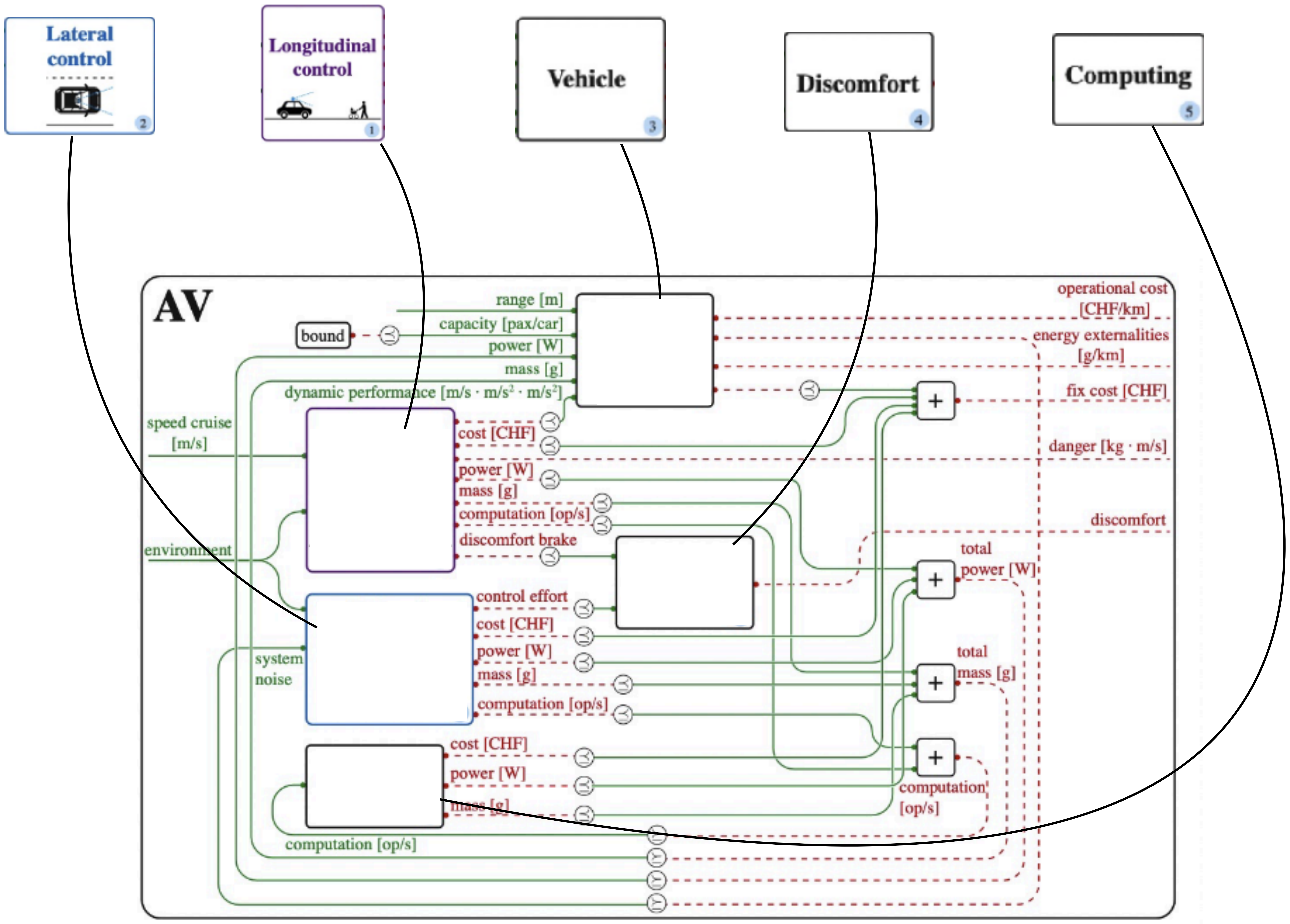
Composition



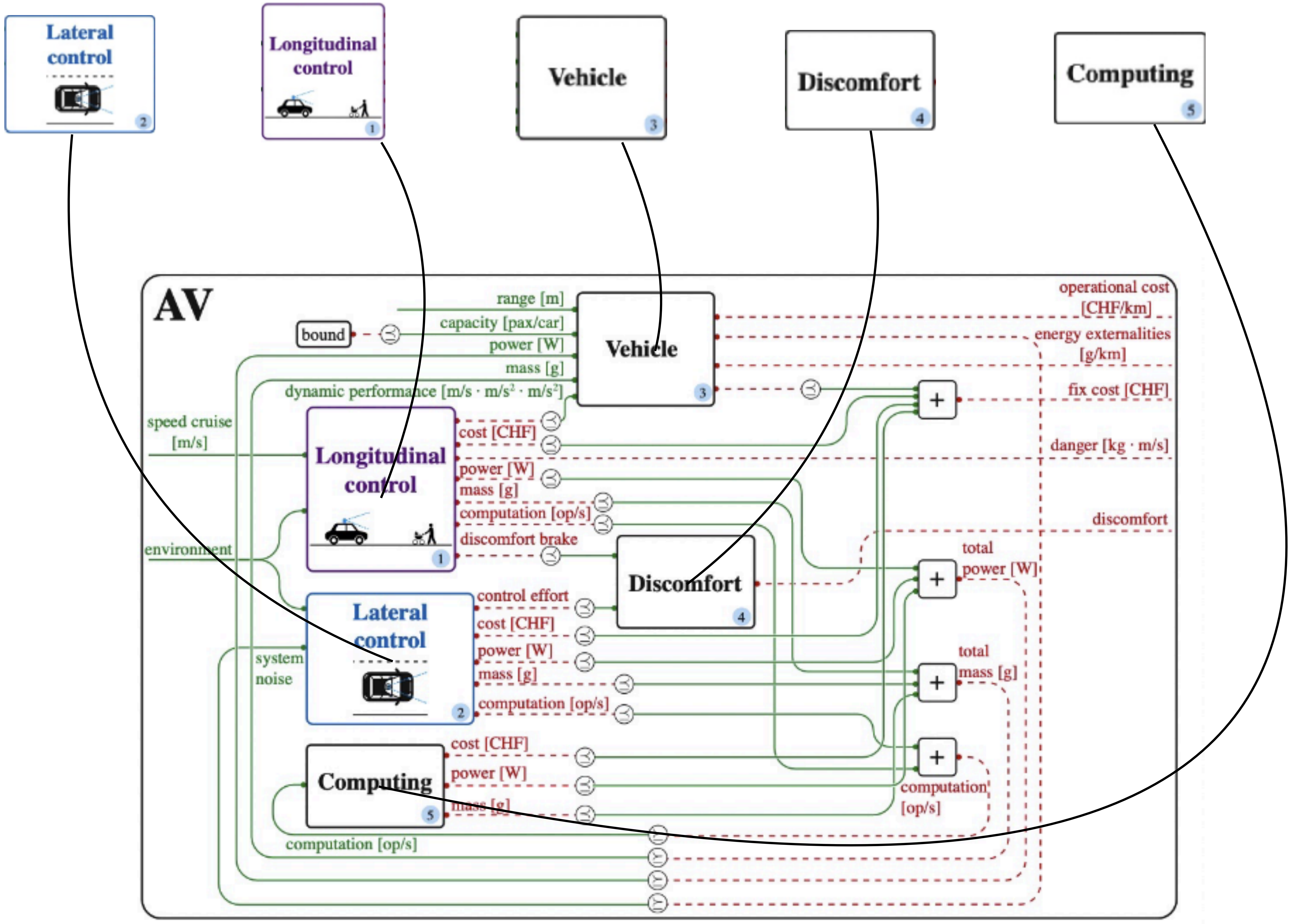
Codesign


AV
range [m]
capacity [pax/car]
power [W]
mass [g]
dynamic performance [m/s · m/s² · m/s²]
bound
operational cost [CHF/km]
energy externalities [g/km]
fix cost [CHF]
danger [kg · m/s]
discomfort
speed cruise [m/s]
environment
system noise
cost [CHF]
power [W]
mass [g]
computation [op/s]
discomfort brake
control effort
cost [CHF]
power [W]
mass [g]
computation [op/s]
cost [CHF]
power [W]
mass [g]
computation [op/s]
total power [W]
total mass [g]
computation [op/s]

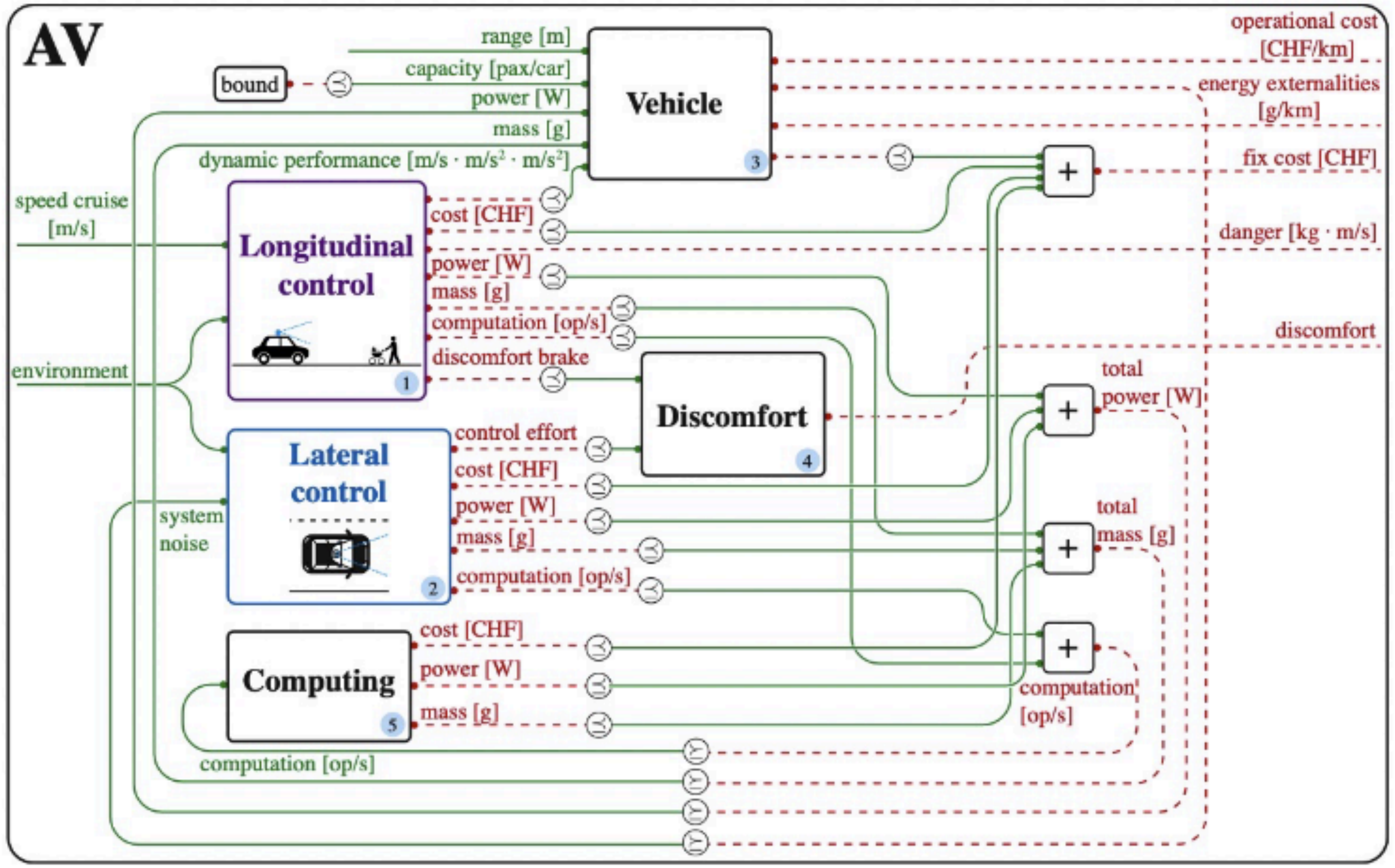
Codesign


Lateral control
Longitudinal control
Vehicle
Discomfort
Computing
AV
bound
range [m]
capacity [pax/car]
power [W]
mass [g]
dynamic performance [m/s · m/s² · m/s²]
speed cruise [m/s]
environment
system noise
cost [CHF]
power [W]
mass [g]
computation [op/s]
discomfort brake
control effort
cost [CHF]
power [W]
mass [g]
computation [op/s]
cost [CHF]
power [W]
mass [g]
computation [op/s]
operational cost [CHF/km]
energy externalities [g/km]
fix cost [CHF]
danger [kg · m/s]
discomfort
total power [W]
total mass [g]
computation [op/s]

Codesign


Lateral control
Longitudinal control
Vehicle
Discomfort
Computing
AV
bound
range [m]
capacity [pax/car]
power [W]
mass [g]
dynamic performance [m/s · m/s² · m/s²]
speed cruise [m/s]
environment
system noise
Longitudinal control
cost [CHF]
power [W]
mass [g]
computation [op/s]
discomfort brake
Vehicle
Discomfort
Lateral control
control effort
cost [CHF]
power [W]
mass [g]
computation [op/s]
Computing
cost [CHF]
power [W]
mass [g]
computation [op/s]
operational cost [CHF/km]
energy externalities [g/km]
fix cost [CHF]
danger [kg · m/s]
discomfort
total power [W]
total mass [g]
computation [op/s]

Codesign



AV
bound
range [m]
capacity [pax/car]
power [W]
mass [g]
dynamic performance [m/s · m/s² · m/s²]
Vehicle
operational cost [CHF/km]
energy externalities [g/km]
fix cost [CHF]
danger [kg · m/s]
discomfort
speed cruise [m/s]
environment
Longitudinal control
cost [CHF]
power [W]
mass [g]
computation [op/s]
discomfort brake
Discomfort
total power [W]
Lateral control
system noise
control effort
cost [CHF]
power [W]
mass [g]
computation [op/s]
total mass [g]
Computing
cost [CHF]
power [W]
mass [g]
computation [op/s]
computation [op/s]