Free-algebra functors as coalgebraic signatures

H. Peter Gumm

Philipps Universität Marburg

PALS Seminar Boulder 27.10.2020

Intro

State based systems

Functors and Coalgebras

Fuctor properties

Weak Pullback Preservation

Functors parameterized by algebras

Free-algebra functor

Preimage preservation

Weak kernel preservation

Conclusion

- and Breaking News

Intro

- State based systems
- Functors and Coalgebras

Functor properties

- Weak Pullback Preservation
- Functors parameterized by algebras

Free-algebra functor

- Preimage preservation
- Weak kernel preservation

Conclusion

- - and Breaking News

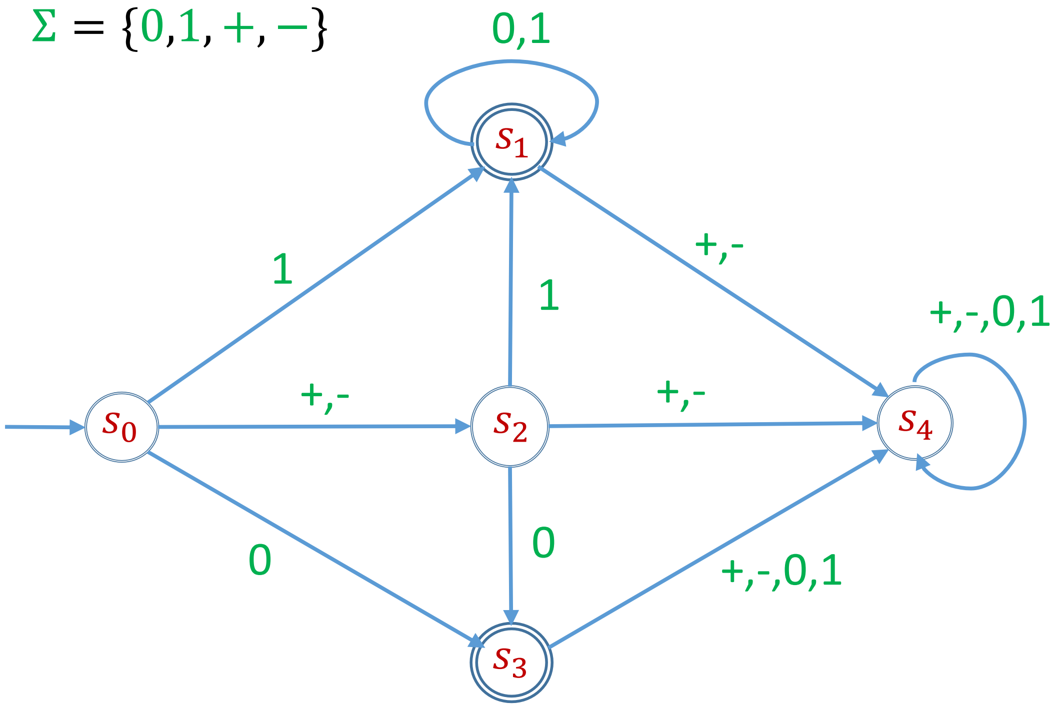
Automata

S set of states,

Σ input alphabet

- Acceptor:
  - $\delta: S \times \Sigma \rightarrow S$
  - $T \subseteq S$

$\Sigma = \{0,1,+,-\}$



e.g.

-1011 accepted

+0110 not accepted

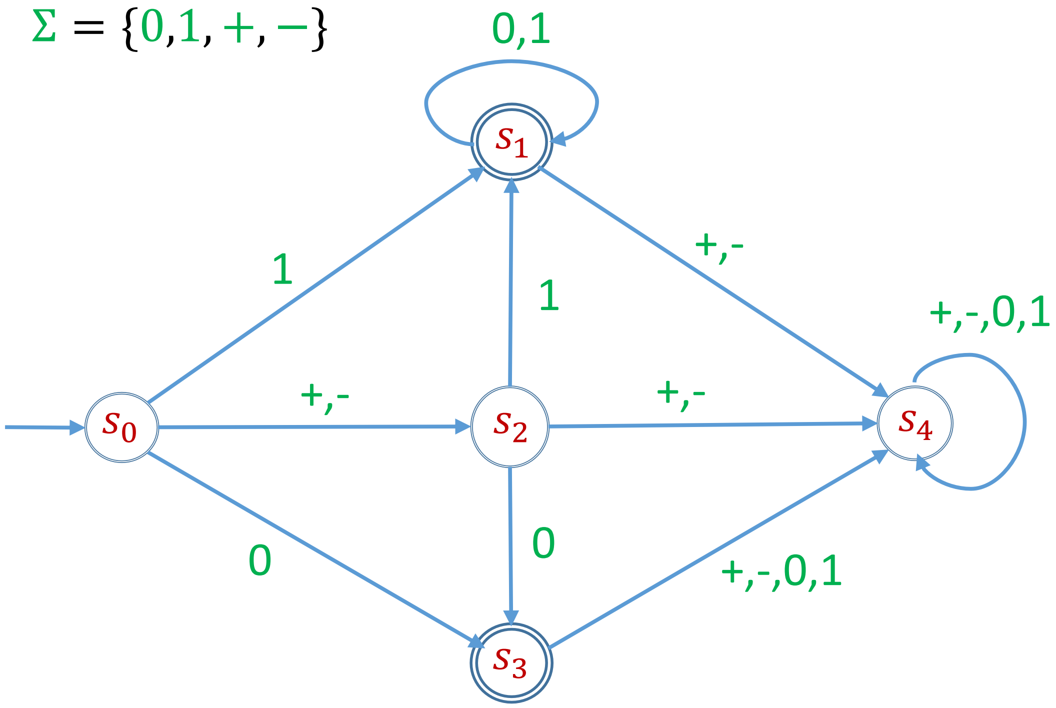
Automata

S set of states,

Σ input alphabet

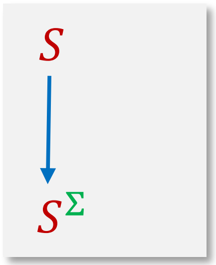
- Acceptor:
  - $\delta: S \times \Sigma \to S$
  - $T \subseteq S$

$\Sigma = \{0, 1, +, -\}$



e.g.

-1011	accepted
+0110	not accepted

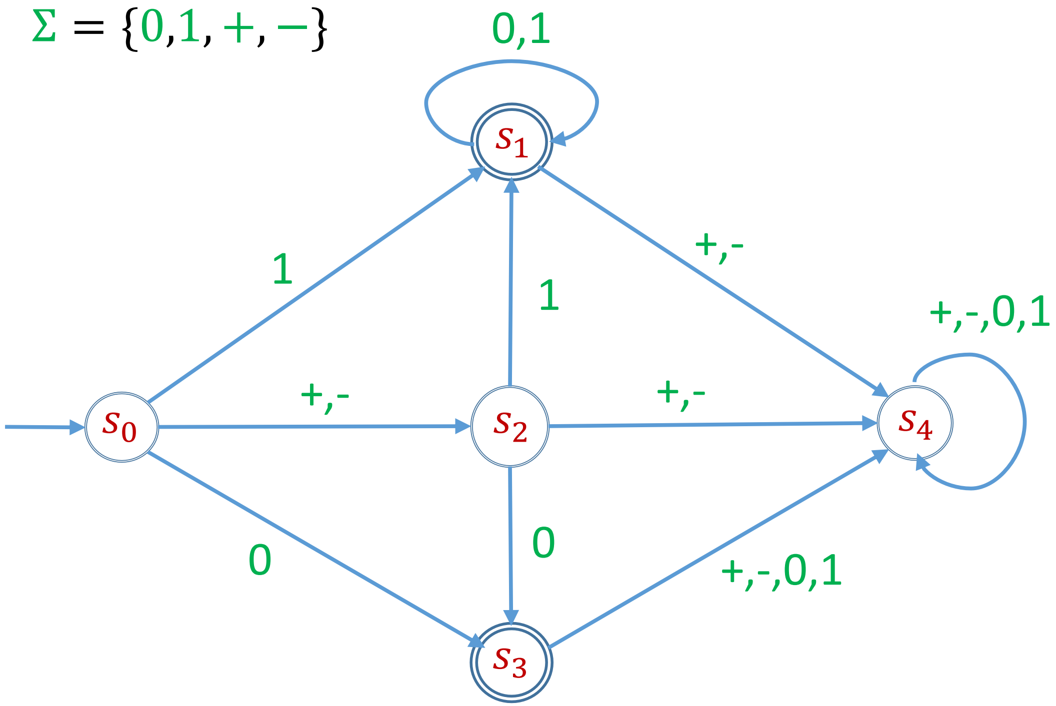


Automata

S set of states,

Σ input alphabet

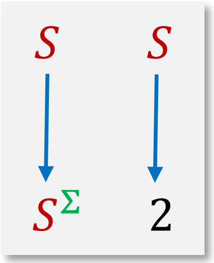
- Acceptor:
  - $\delta: S \times \Sigma \to S$
  - $T \subseteq S$



e.g.

-1011 accepted

+0110 not accepted



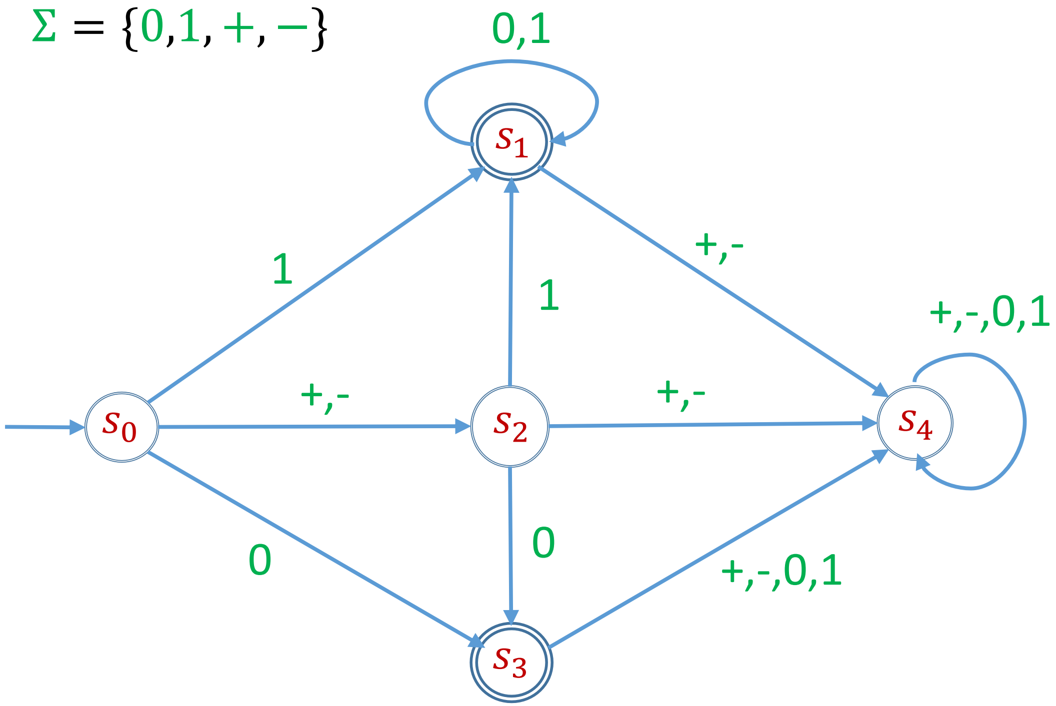
Automata

S set of states,

Σ input alphabet

- Acceptor:
  - $\delta: S \times \Sigma \to S$
  - $T \subseteq S$

$\Sigma = \{0, 1, +, -\}$



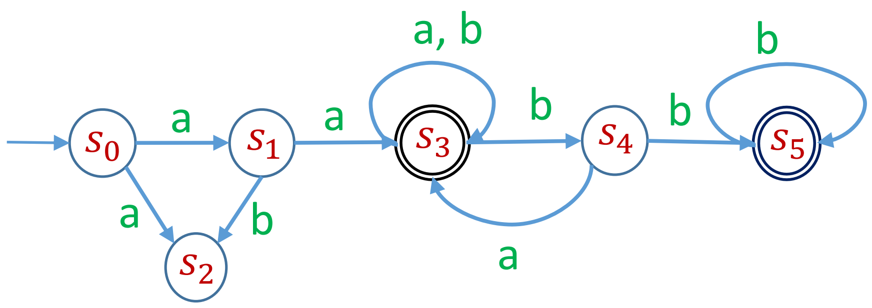
e.g.

-1011	accepted
+0110	not accepted

$$S \to S^{\Sigma} \times 2$$

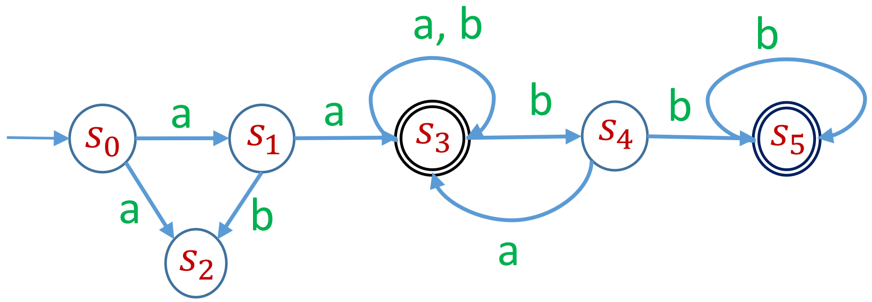
Nondeterminism

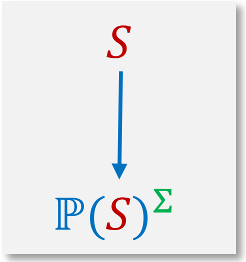
- NFA
  - $\delta: S \times \Sigma \to \mathbb{P}(S)$
  - $T \subseteq S$



Nondeterminism

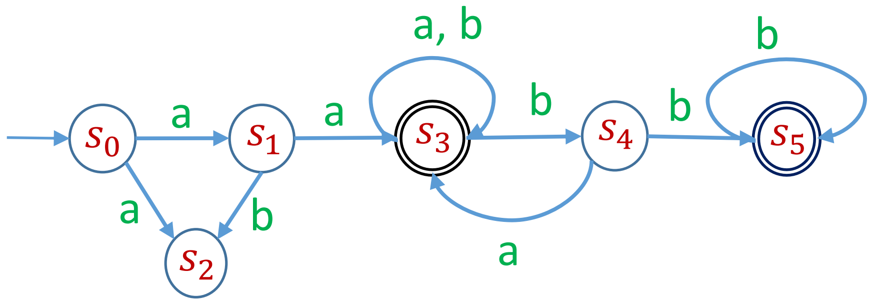
- NFA
  - $\delta: S \times \Sigma \rightarrow \mathbb{P}(S)$
  - $T \subseteq S$

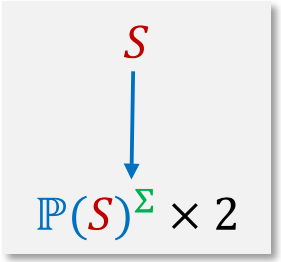




Nondeterminism

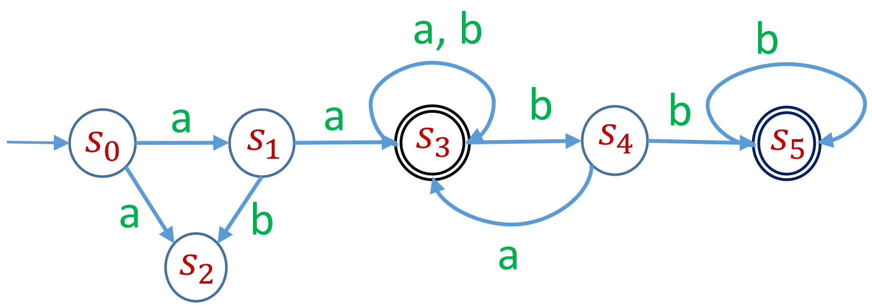
- NFA
  - $\delta: S \times \Sigma \to \mathbb{P}(S)$
  - $T \subseteq S$

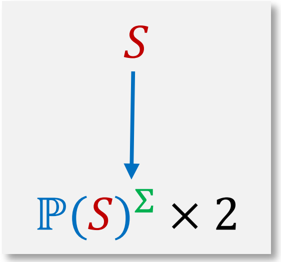




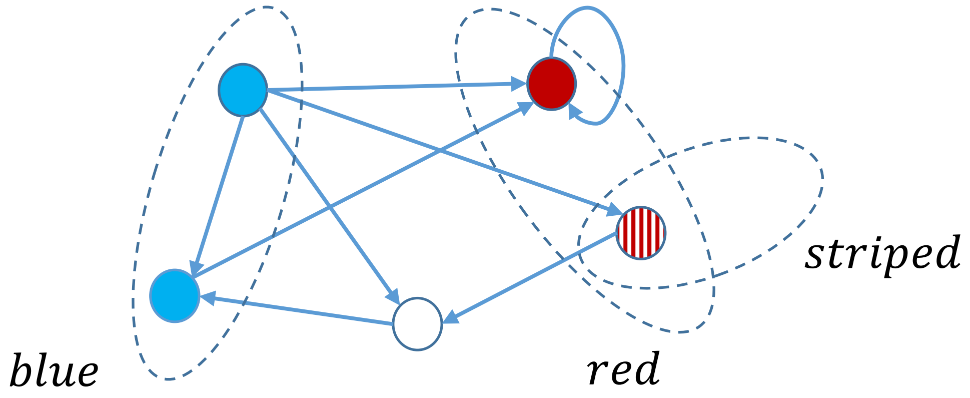
Nondeterminism

- NFA
  - $\delta : S \times \Sigma \to \mathbb{P}(S)$
  - $T \subseteq S$



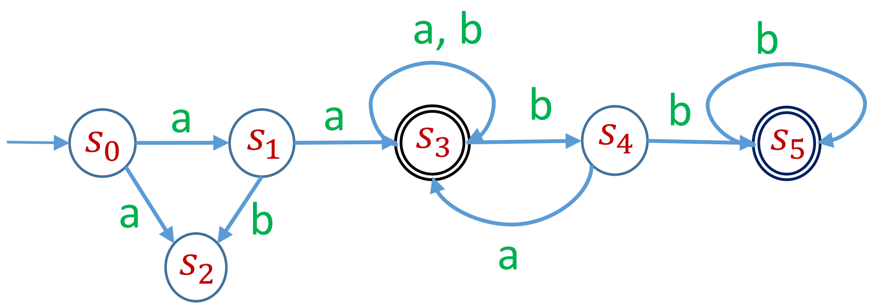


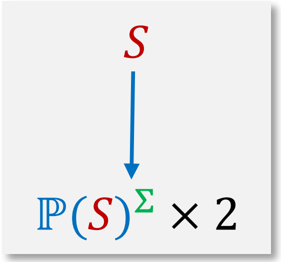
- Kripke structure
  - $R \subseteq S \times S$
  - $v : S \to \mathbb{P}(C)$



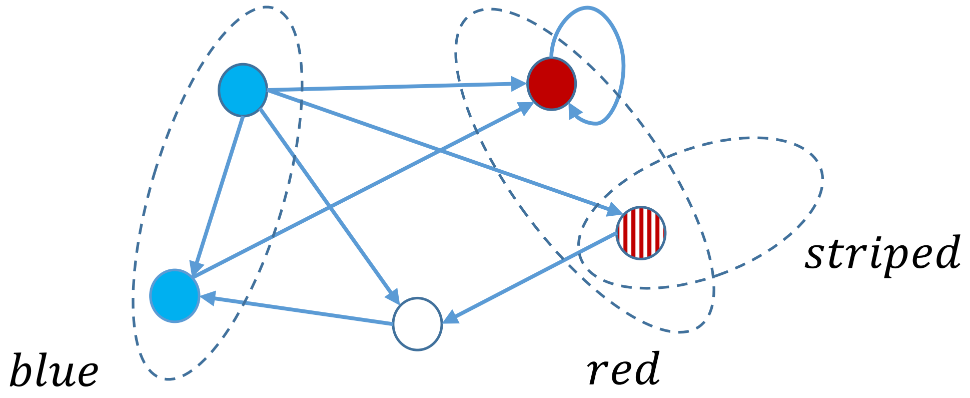
Nondeterminism

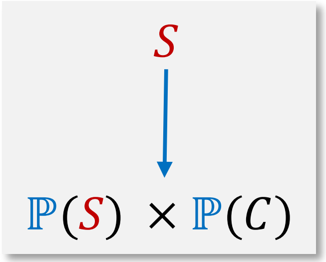
- NFA
  - $\delta: S \times \Sigma \to \mathbb{P}(S)$
  - $T \subseteq S$





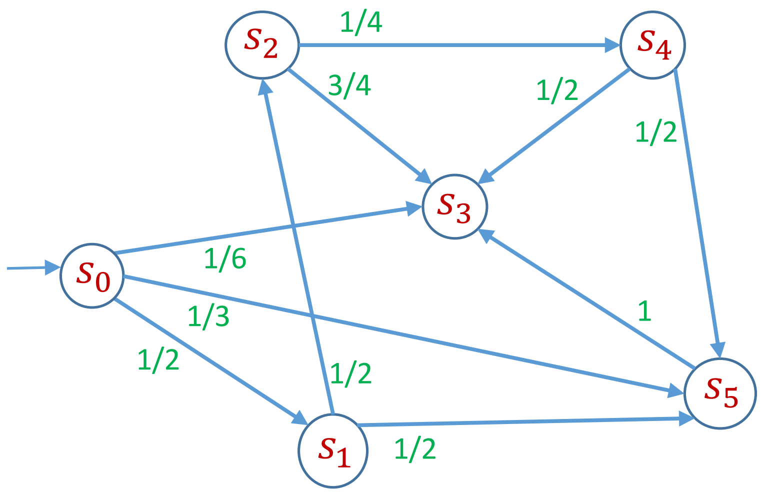
- Kripke structure
  - $R \subseteq S \times S$
  - $v : S \to \mathbb{P}(C)$





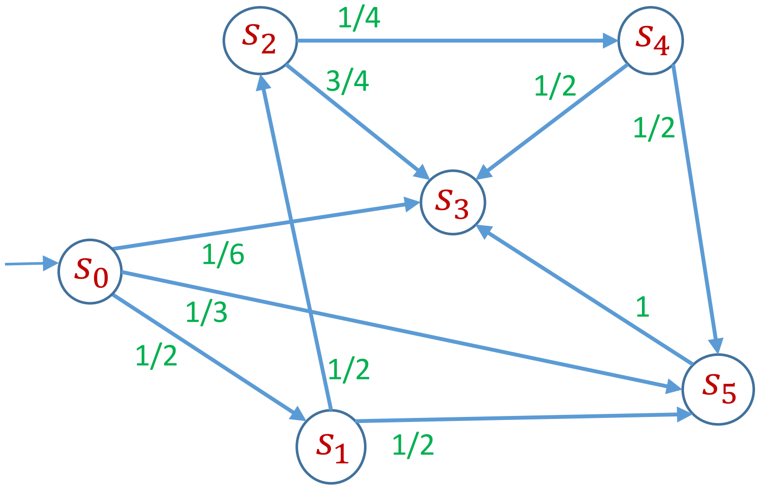
Probabilistic

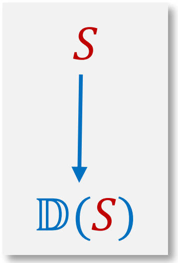
- Probabilistic systems
  - $\delta: S \times S \to [0,1]_{\mathbb{R}}$
  - $\sum_{x \in S} \delta(s, x) = 1$



Probabilistic

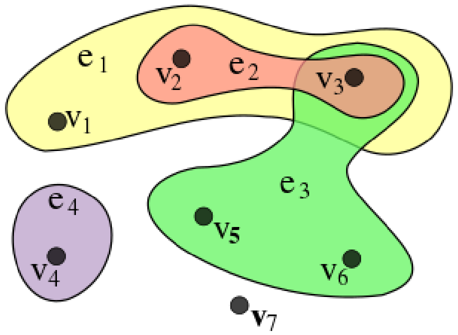
- Probabilistic systems
  - $\delta: S \times S \to [0,1]_{\mathbb{R}}$
  - $\sum_{x \in S} \delta(s,x) = 1$





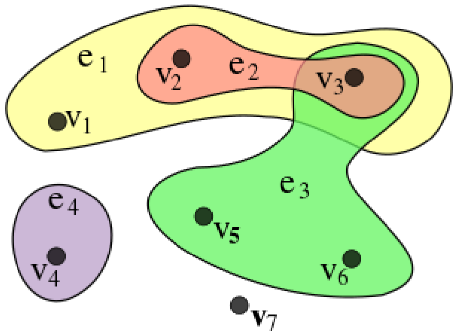
Higher order

- Neighbourhood structure
  - $R \subseteq S \times 2^S$
  - $c: S \rightarrow \mathbb{P}(C)$



Higher order

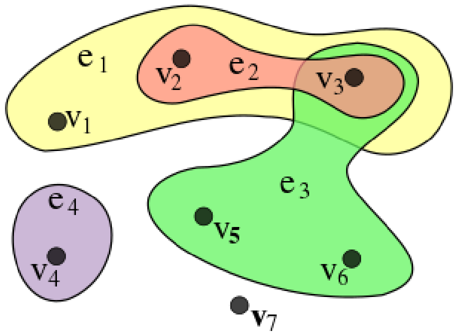
- Neighbourhood structure
  - $R \subseteq S \times 2^{S}$
  - $c: S \to \mathbb{P}(C)$

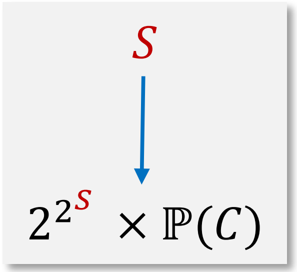


$$S \downarrow 2^{2^{S}} \times \mathbb{P}(C)$$

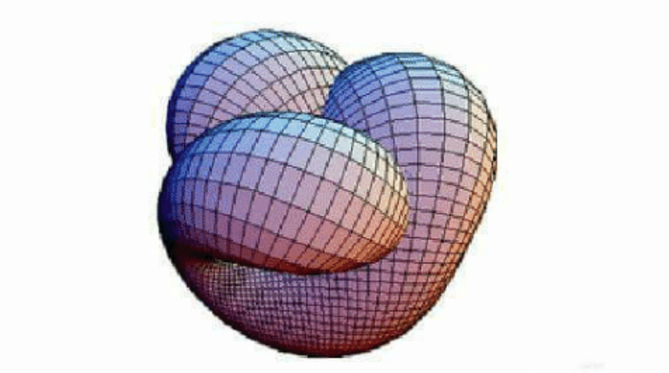
Higher order

- Neighbourhood structure
  - $R \subseteq S \times 2^S$
  - $c: S \to \mathbb{P}(C)$



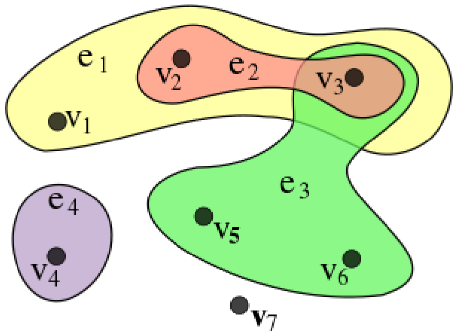


- Topological space
  - $\tau \subseteq \mathbb{P}(\mathbb{P}(S))$
    - closed under
      - unions
      - finite intersections



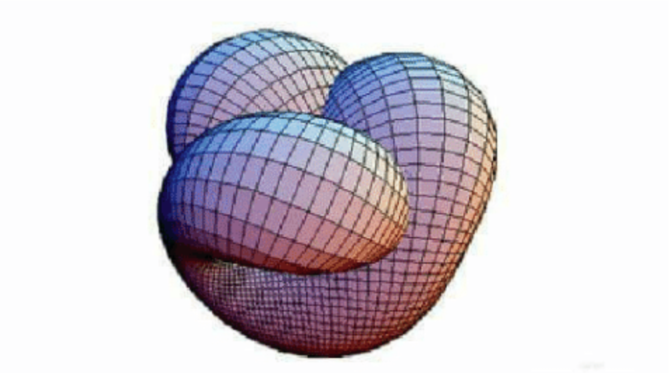
Higher order

- Neighbourhood structure
  - $R \subseteq S \times 2^{S}$
  - $c: S \to \mathbb{P}(C)$



S
$\downarrow$
$2^{2^{S}} \times \mathbb{P}(C)$

- Topological space
  - $\tau \subseteq \mathbb{P}(\mathbb{P}(S))$
    - closed under
      - unions
      - finite intersections



S
$\downarrow$
$\mathbb{F}il(S)$

Coalgebras and Algebras

$$S \xrightarrow{\alpha} F(S)$$

$$S \to S^{\Sigma} \times 2 \qquad S \to \mathbb{P}(S)^{\Sigma} \times 2 \qquad S \to \mathbb{D}(S) \qquad S \to 2^{2^{S}} \times \mathbb{P}(C) \qquad S \to \mathbb{F}il(S)$$

- Set functors fix the „signature“ of coalgebras

Coalgebras and Algebras

$$S \xrightarrow{\alpha} F(S)$$

$$S \rightarrow S^{\Sigma} \times 2 \qquad S \rightarrow \mathbb{P}(S)^{\Sigma} \times 2 \qquad S \rightarrow \mathbb{D}(S) \qquad S \rightarrow 2^{2^{S}} \times \mathbb{P}(C) \qquad S \rightarrow \mathbb{F}il(S)$$

- Set functors fix the „signature“ of coalgebras
- Algebras are upside-down coalgebras

$$A \times A \rightarrow A \qquad A^2 + A^2 \rightarrow A \qquad A^2 + A + 1 \rightarrow A \qquad \ldots \qquad \mathbb{P}(A) \rightarrow A$$

$$F(A) \xrightarrow{\alpha} A$$

Intro

State based systems

Functors and Coalgebras

Fuctor properties

Weak Pullback Preservation

Functors parameterized by algebras

Free-algebra functor

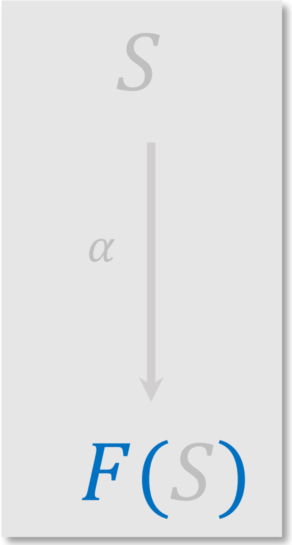
Preimage preservation

Weak kernel preservation

Conclusion

- and Breaking News

Functors

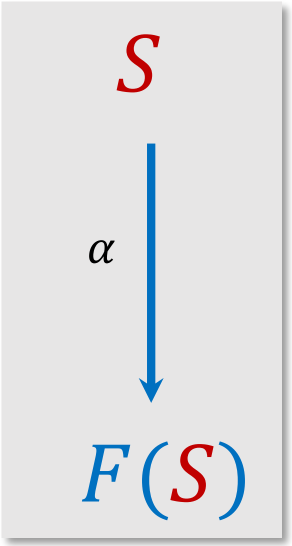


- $F(X)$ a „*natural set theoretical construction*“
- F should be a *functor*:
  - for each X construct new set $F(X)$
  - for each $f: X \to Y$ provide map $Ff: F(X) \to F(Y)$

    such that

    - $F(g \circ f) = Fg \circ Ff$
    - $F\ \iota d_A = \iota d_{F(A)}$

Functors and Coalgebras



F-coalgebra

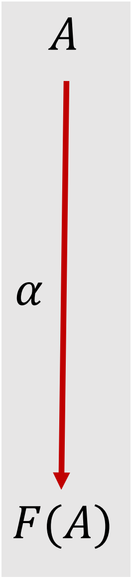
- $F(X)$ a „*natural set theoretical construction*“
- F should be a *functor*:
  - for each X construct new set $F(X)$
  - for each $f: X \to Y$ provide map $Ff: F(X) \to F(Y)$

  such that

  - $F(g \circ f) = Fg \circ Ff$
  - $F\ \iota d_A = \iota d_{F(A)}$

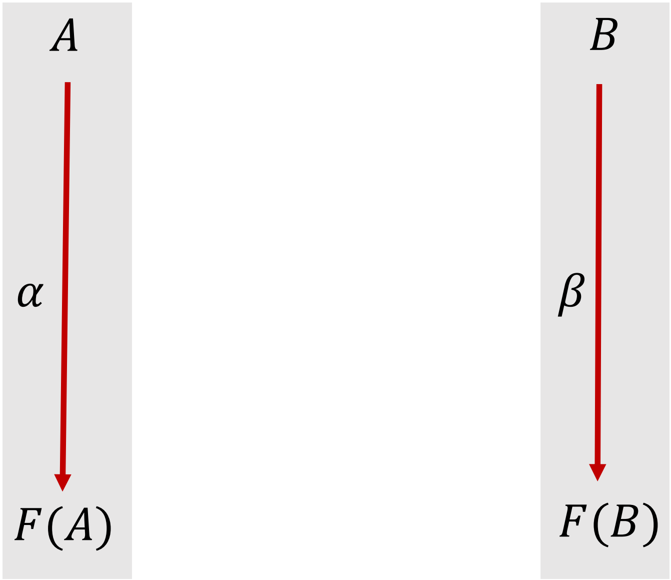
F-coalgebras

Homomorphism



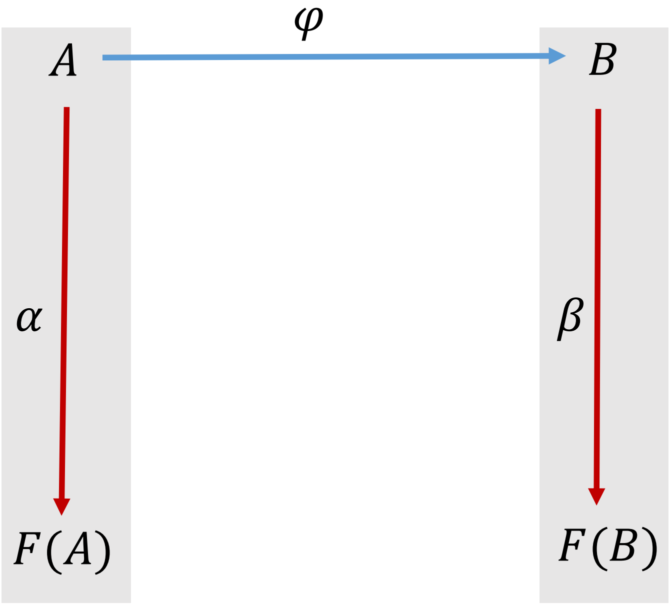
F-coalgebras

Homomorphism



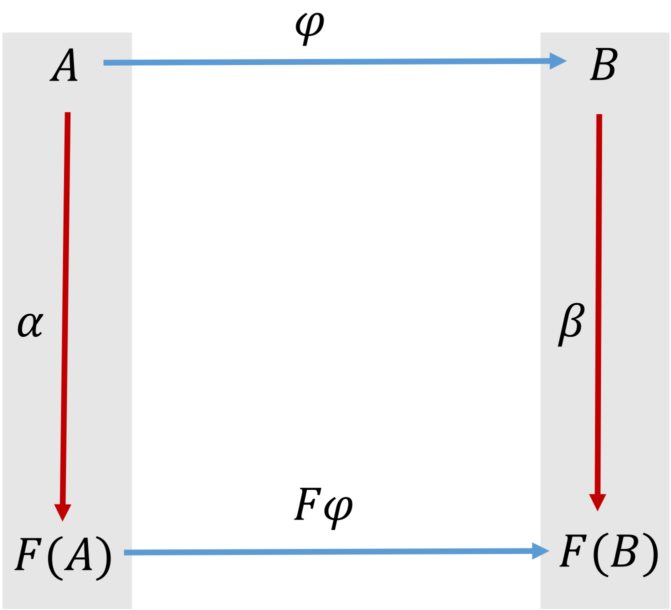
F-coalgebras and homomorphisms

Homomorphism

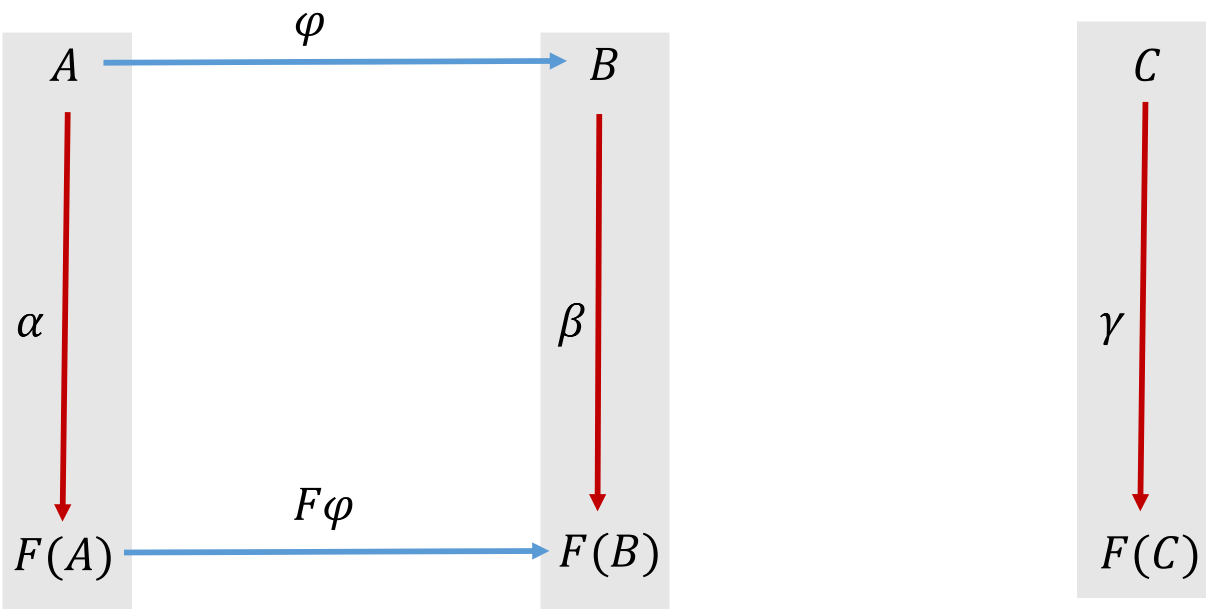


F-coalgebras and homomorphisms

Homomorphism

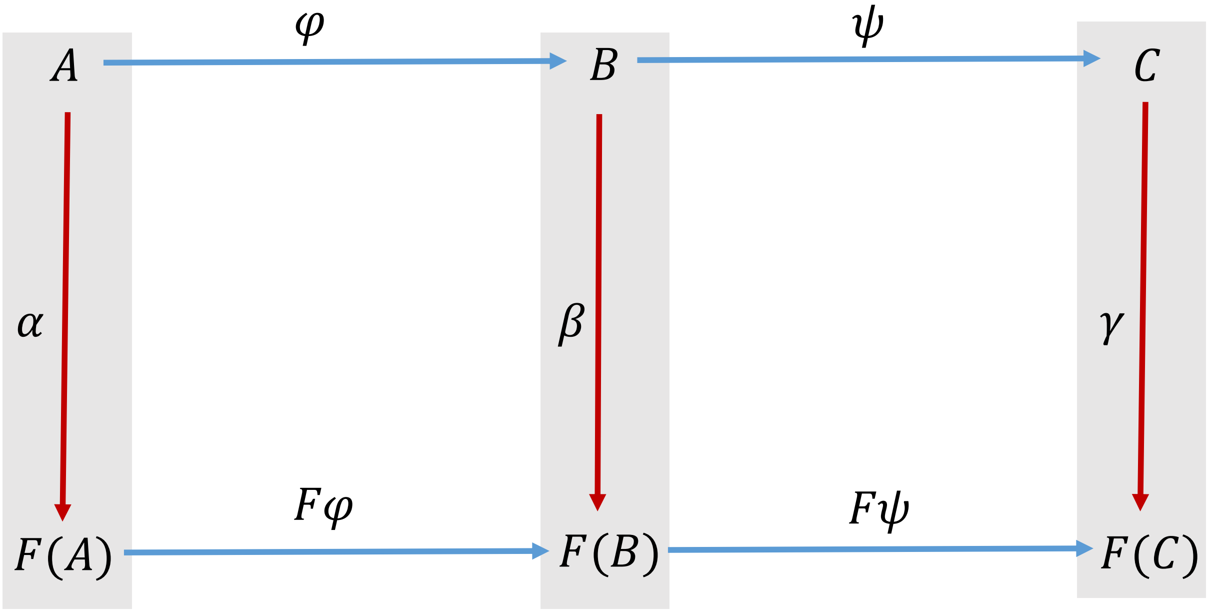


F-coalgebras and homomorphisms



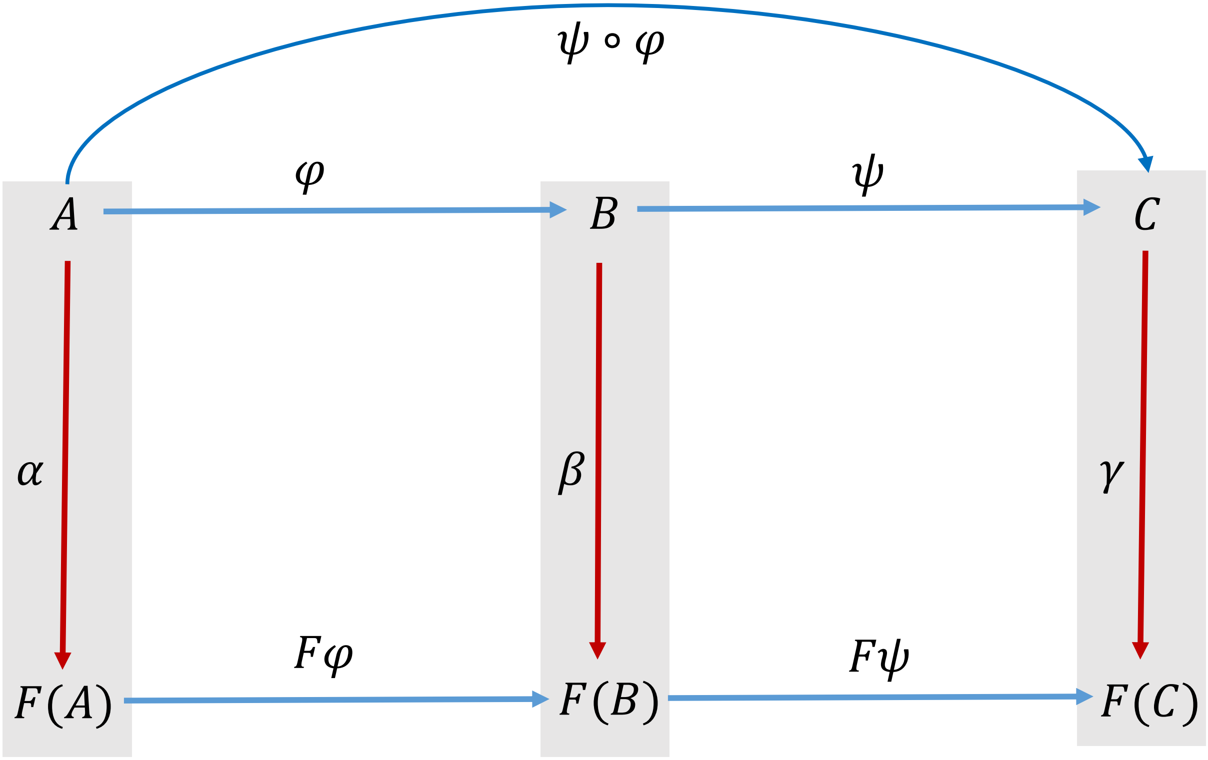
Homomorphisms compose

F-coalgebras and homomorphisms



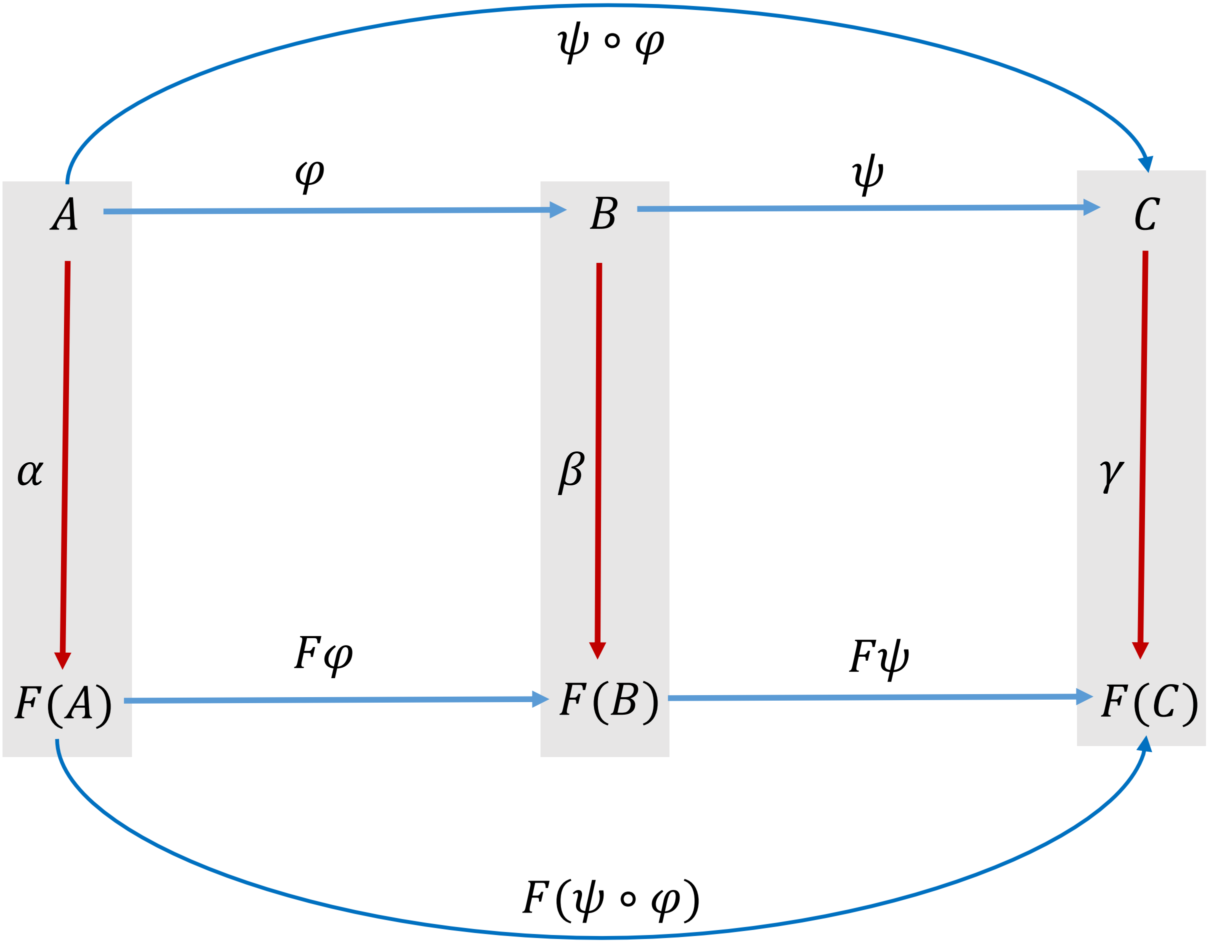
Homomorphisms compose

F-coalgebras and homomorphisms



Homomorphisms compose

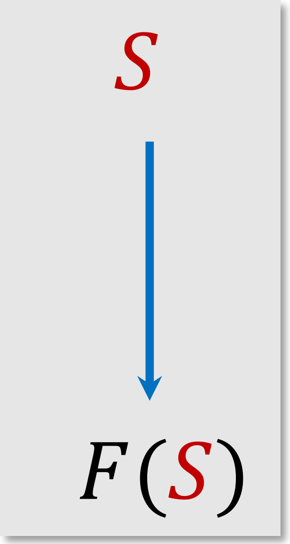
F-coalgebras and homomorphisms



Homomorphisms compose

$F(\psi \circ \varphi) = F\psi \circ F\varphi$

Set_F



- F-coalgebras form a category Set_F
  - Homomorphism theorems
  - Substructures, quotients, congruences, …
  - Co-equations, Co-Birkhoff
  - Modal logic
  - …
- Properties of F determine structure of Set_F
  - *weak pullback preservation*

Intro

State based systems
Functors and Coalgebras

Fuctor properties

Weak Pullback Preservation
Functors parameterized by algebras

Free-algebra functor

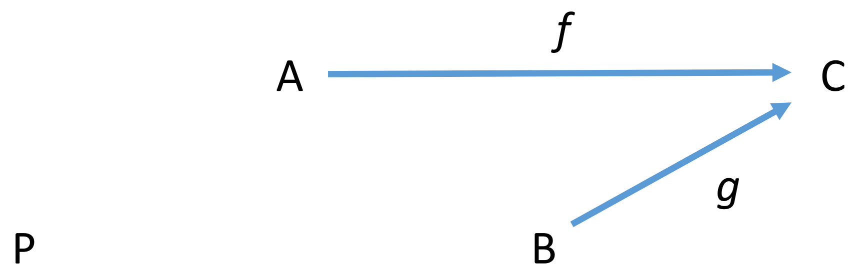
Preimage preservation
Weak kernel preservation

Conclusion

- and Breaking News

F weakly preserves pullbacks

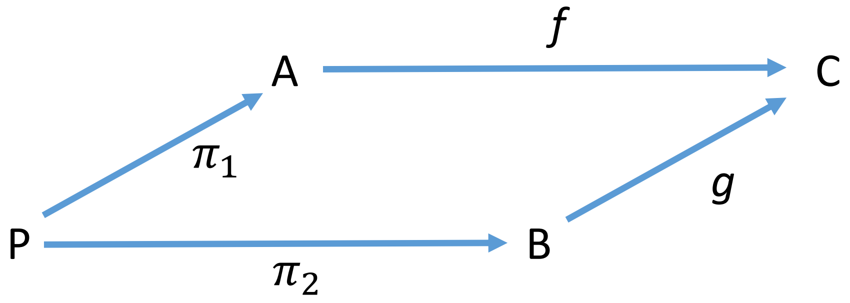
- (Weak) pullback



$P = \{(a, b) \in A \times B \mid f(a) = g(b)\}$

F weakly preserves pullbacks

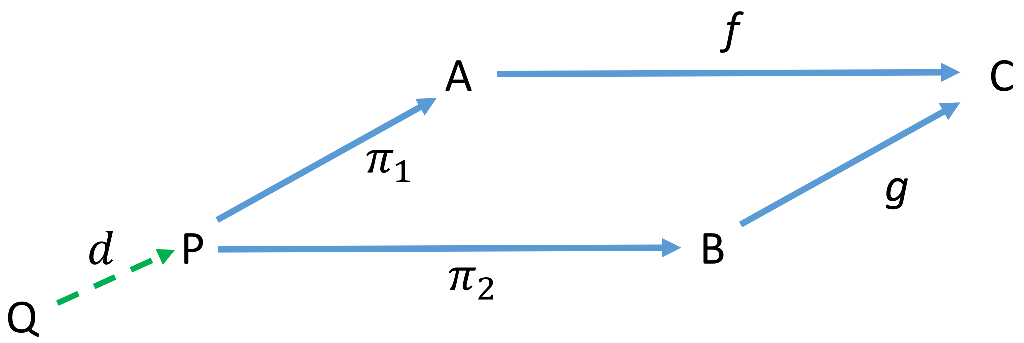
- (Weak) pullback



$P = \{(a,b) \in A \times B \mid f(a) = g(b)\}$

F weakly preserves pullbacks

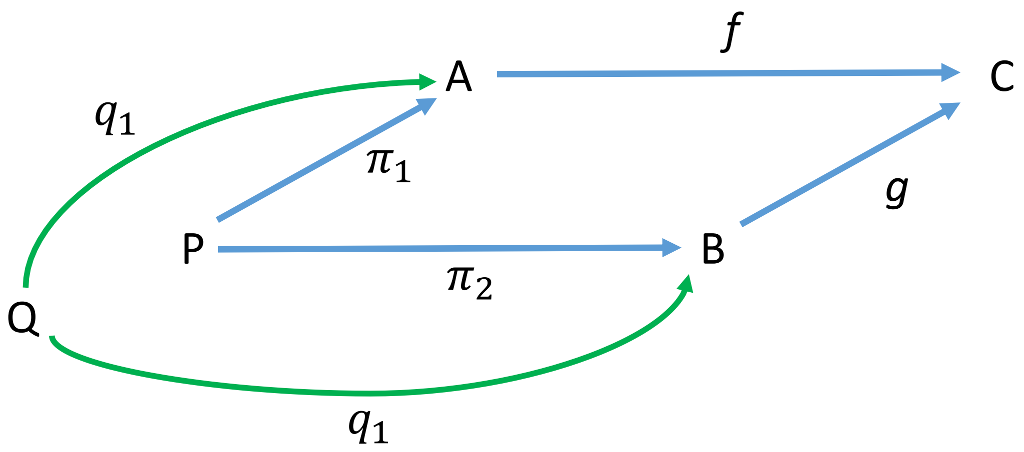
- (Weak) pullback



$$P = \{(a, b) \in A \times B \mid f(a) = g(b)\}$$

F weakly preserves pullbacks

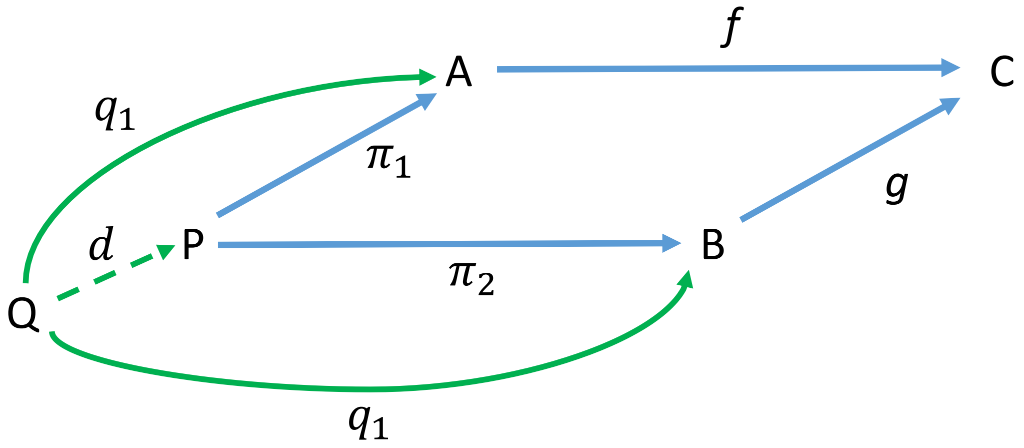
- (Weak) pullback



$$P = \{(a, b) \in A \times B \mid f(a) = g(b)\}$$

F weakly preserves pullbacks

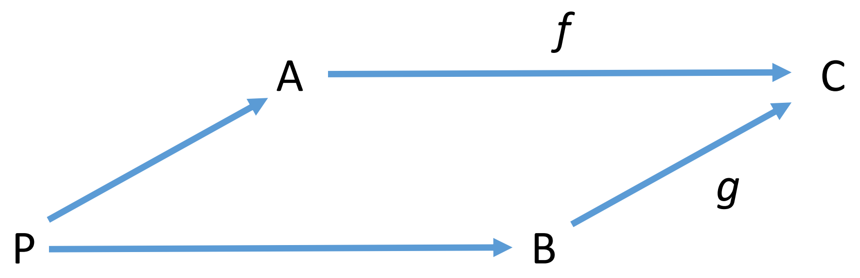
- (Weak) pullback



$P = \{(a, b) \in A \times B \mid f(a) = g(b)\}$

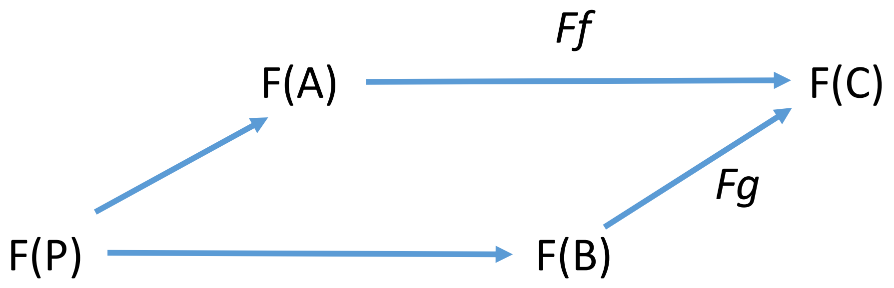
F weakly preserves pullbacks

- (Weak) pullback



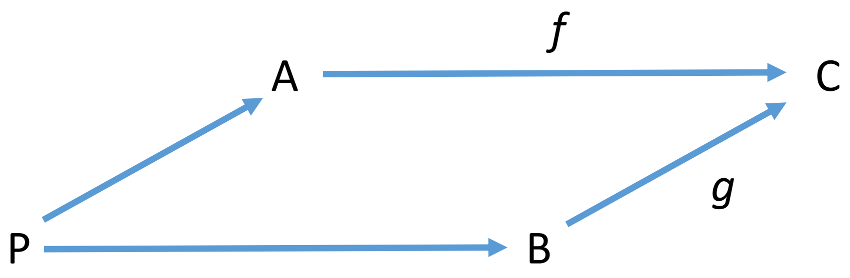
$$P = \{(a, b) \in A \times B \mid f(a) = g(b)\}$$

- apply F



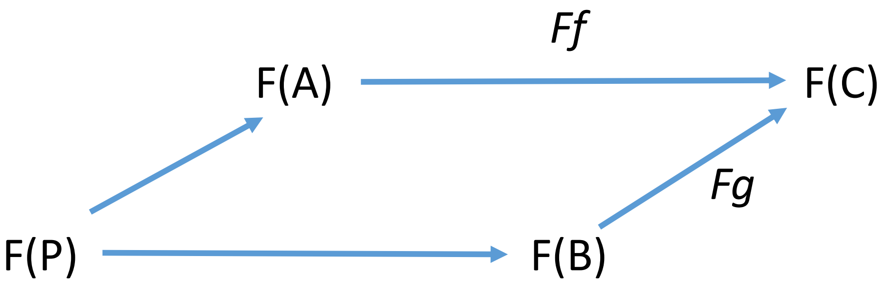
F weakly preserves pullbacks

- (Weak) pullback



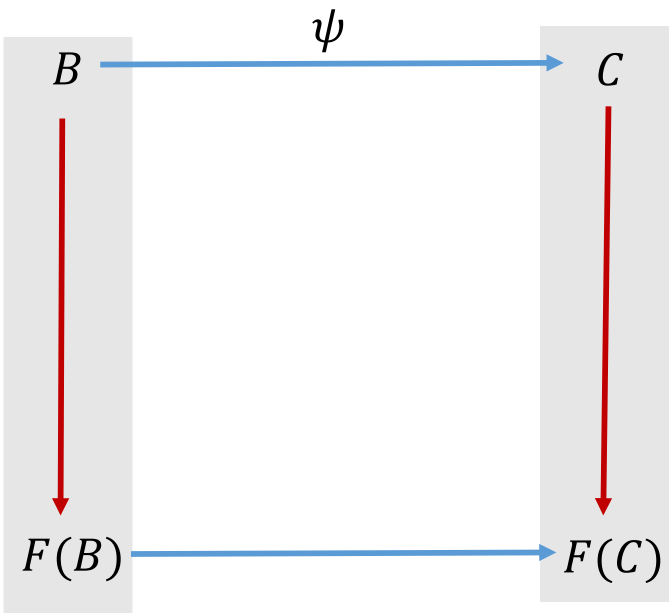
$$P = \{(a,b) \in A \times B \mid f(a) = g(b)\}$$

- apply F

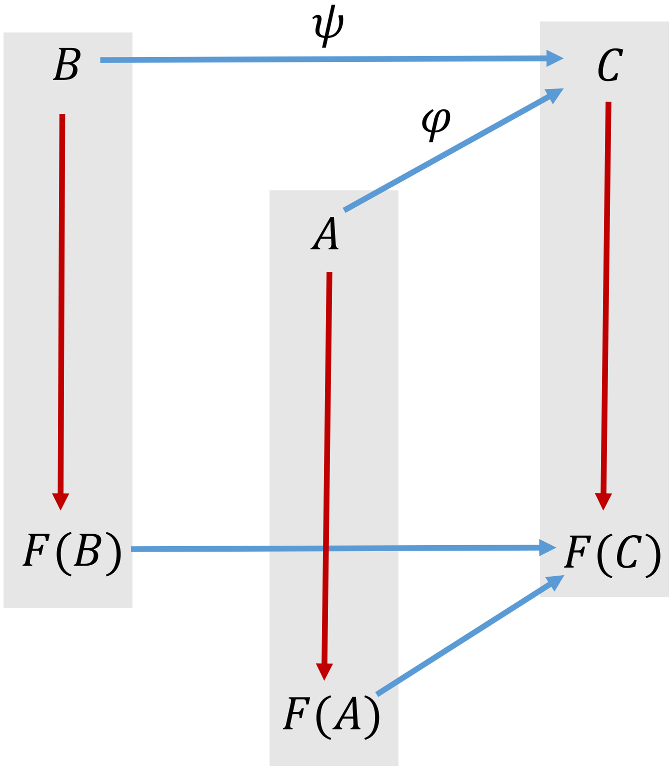


Is this a weak pullback diagram?

Observational equivalence

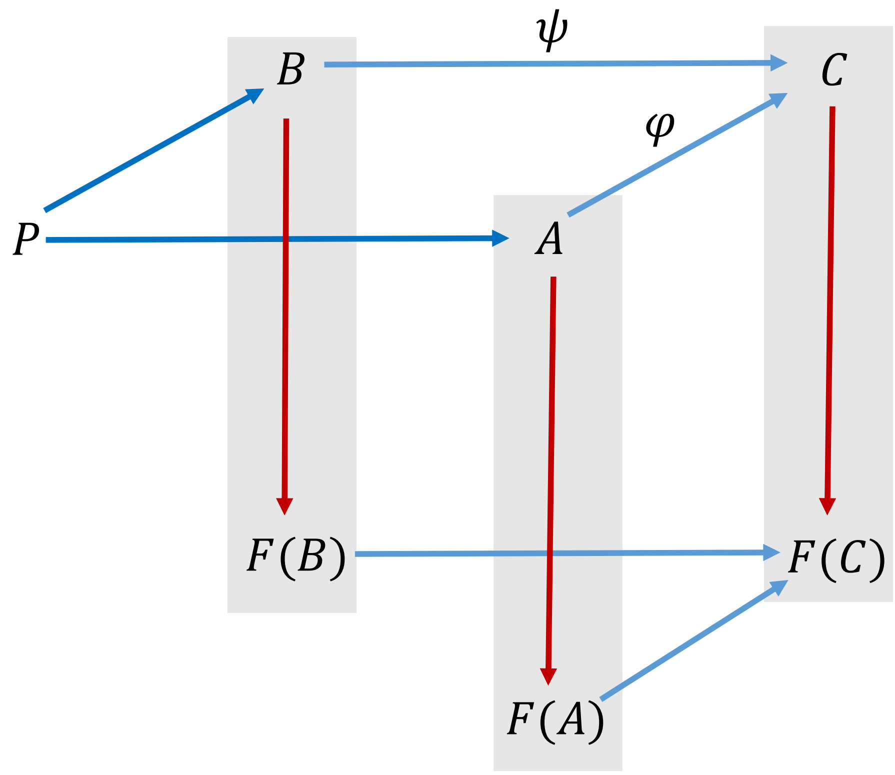


Observational equivalence



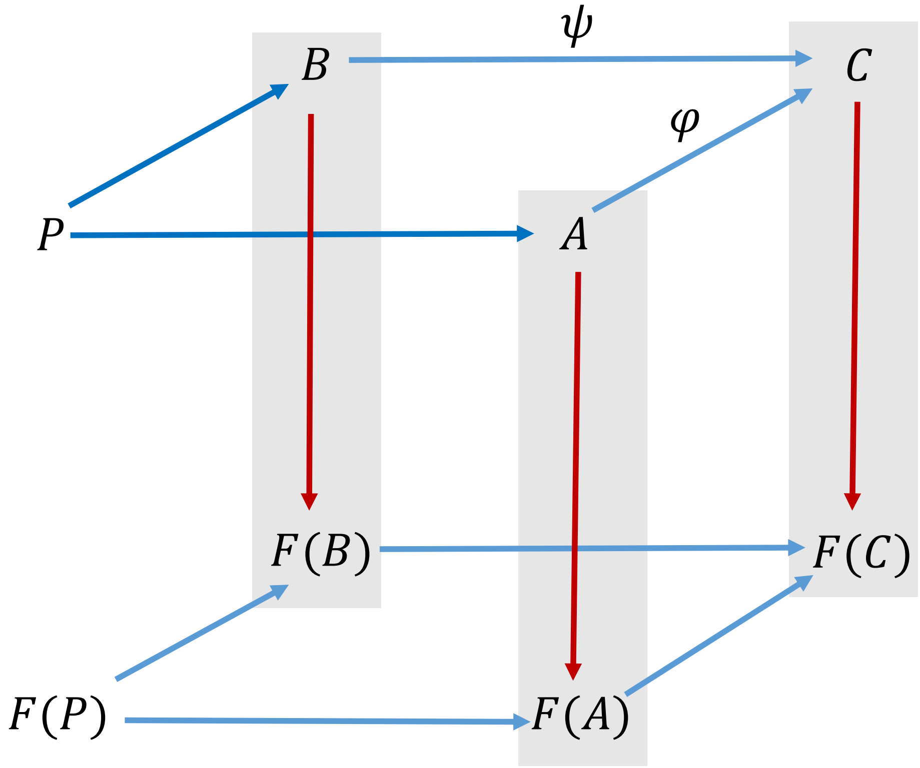
Observational equivalence

Observational equivalence …



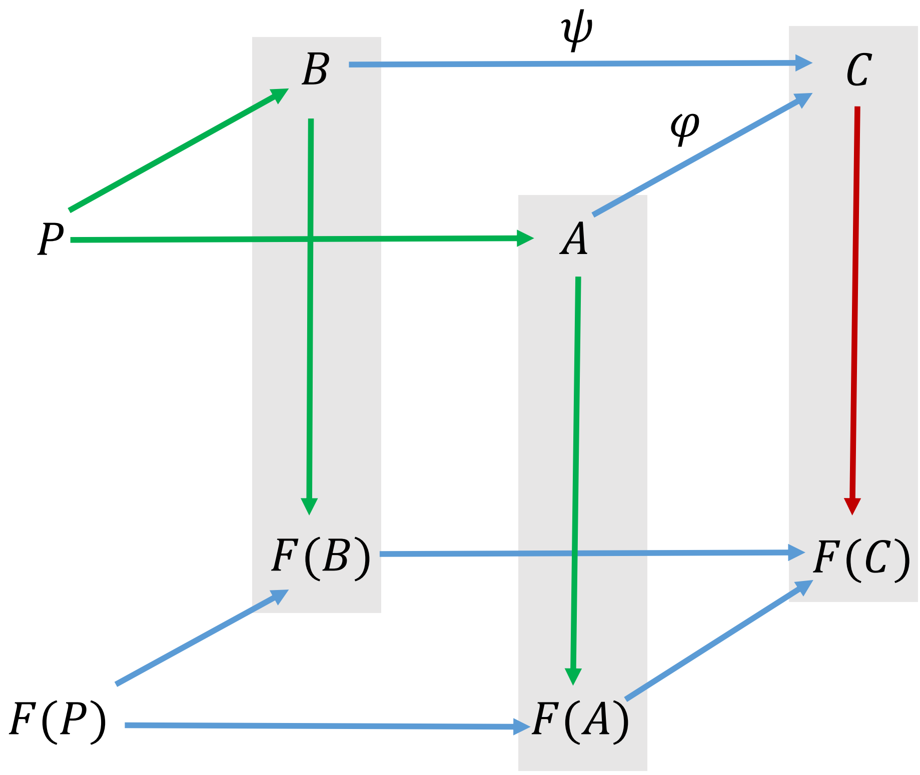
Observational equivalence

Observational equivalence …



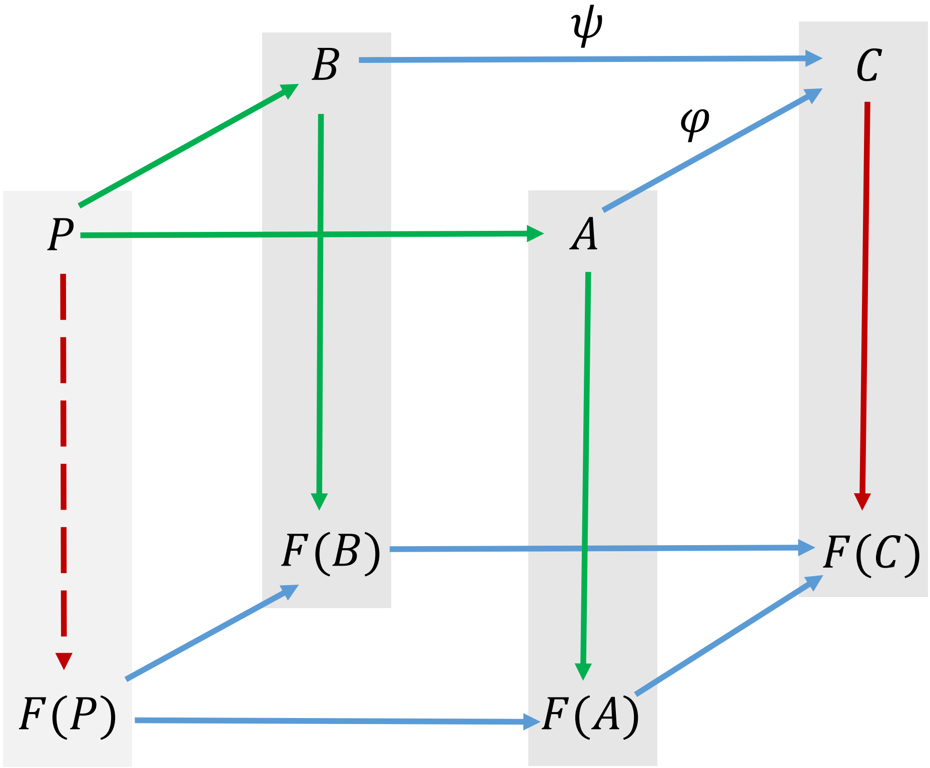
F weakly preserves pullbacks

Observational equivalence …



Bisimilarity

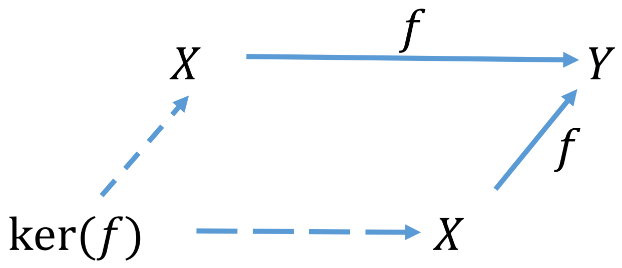
Observational equivalence …



… is bisimilarity

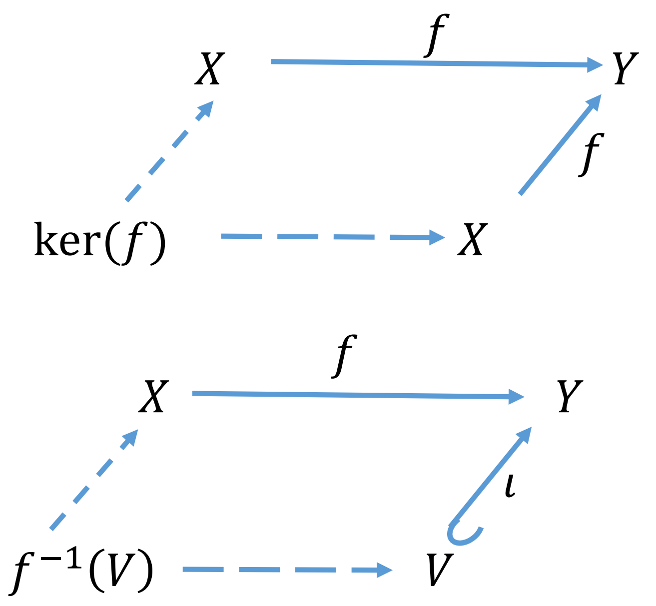
Weak pullback preservation

- F weakly preserves *pullbacks*, iff F preserves
  - *kernel pairs*



Weak pullback preservation

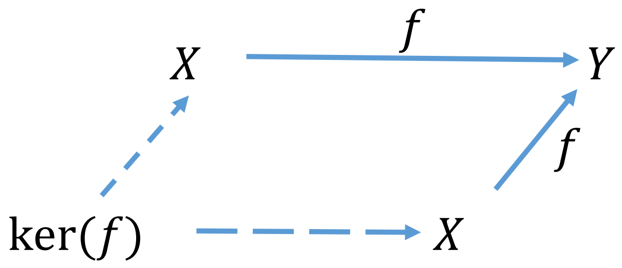
- F weakly preserves *pullbacks*, iff F preserves
  - *kernel pairs*
  - and *preimages*



Weak pullback preservation

- F weakly preserves *pullbacks*, iff F preserves
  - *kernel pairs*
  - and *preimages*

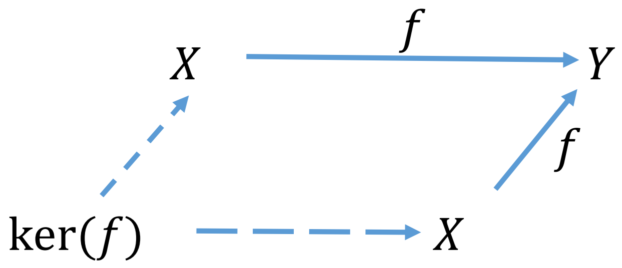
- F weakly preserves *kernel pairs*,
  - iff F preserves pullbacks of epis
  - iff observational equivalence = bisimilarity



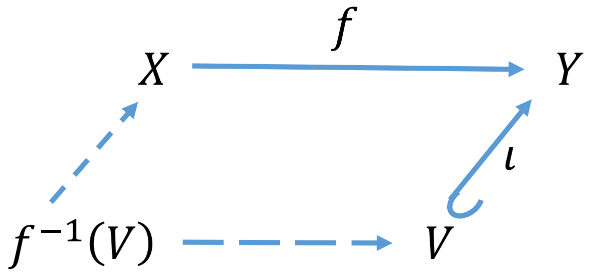
Weak pullback preservation

- F weakly preserves *pullbacks*, iff F preserves
  - *kernel pairs*
  - and *preimages*

- F weakly preserves *kernel pairs*,
  - iff F preserves pullbacks of epis
  - iff observational equivalence = bisimilarity



- F weakly preserves *preimages*
  - iff $HS(\mathfrak{K}) = SH(\mathfrak{K})$, for any class $\mathfrak{K}$
    (provided $|F(1)| > 1$)



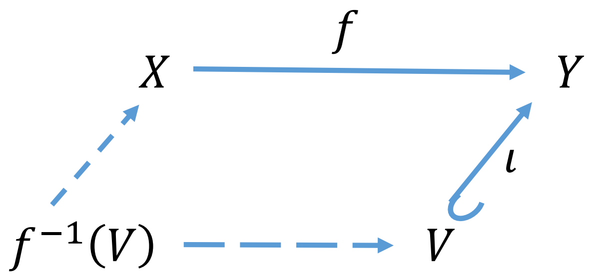
Functors weakly preserving special pullbacks

- all pullbacks
  - $F(X) = \ \dots\ X,\ \ X^n, \Sigma_{i \in I} X^{n_i},\ \mathbb{P}(X),\ \mathbb{P}_\omega(X),\ \mathbb{F}(X),\ D^X, \dots$

Functors weakly preserving special pullbacks

- all pullbacks
  - $F(X) = \ldots X,\ X^n, \Sigma_{i \in I} X^{n_i},\ \mathbb{P}(X),\ \mathbb{P}_\omega(X),\ \mathbb{F}(X),\ D^X, \ldots$

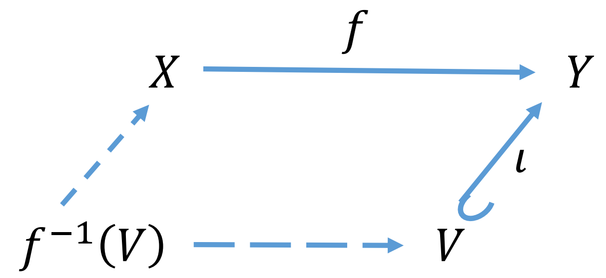
- preimages
  - $F(X) = \ldots \mathbb{P}_{<n}(X),\ X_2^3,\ L^X, \ldots$



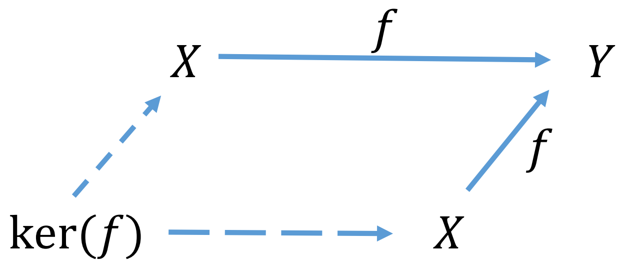
Functors weakly preserving special pullbacks

- all pullbacks
  - $F(X) = \ \dots\ X,\ \ X^n, \Sigma_{i\in I} X^{n_i},\ \mathbb{P}(X),\ \mathbb{P}_\omega(X),\ \mathbb{F}(X),\ D^X, \dots$

- preimages
  - $F(X) = \ \dots\ \mathbb{P}_{<n}(X),\ X_2^3\ ,\ L^X, \dots$



- kernel pairs
  - $F(X) = \ \dots\ 2^{2^X},\ X^2 - X + 1, Odd(X), \dots$



Intro

State based systems

Functors and Coalgebras

Functor properties

Weak Pullback Preservation

Functors parameterized by algebras

Free-algebra functor

Preimage preservation

Weak kernel preservation

Conclusion

- and Breaking News

Parameterizing functors by algebras

- Let $F_{\mathcal{A}}$ depend on some algebra $\mathcal{A}$

- Choose $\mathcal{A}$ so that $F_{\mathcal{A}}$ has desirable properties, e.g. (weakly) preserves
    - *kernel pairs*
    - *preimages*
    - *pullbacks*

L-fuzzy functor

1. Generalize $\mathbb{P}(X) = 2^X$

For $U \subseteq X$ put $\chi_U(x) = \begin{cases} 1, & x \in U \\ 0, & x \notin U \end{cases}$

- Replace $2 = \{0,1\}$ by a complete lattice L
  - on objects: L^X
  - on maps $f: X \to Y$:

$$L^f(\sigma)(y) := \bigvee_{f(x)=y} \sigma(x)$$

L-fuzzy functor

1. Generalize $\mathbb{P}(X) = \mathbf{2}^X$

For $U \subseteq X$ put $\chi_U(x) = \begin{cases} 1, & x \in U \\ 0, & x \notin U \end{cases}$

- Replace $2 = \{0,1\}$ by a complete lattice L
  - on objects: L^X
  - on maps $f\colon X \to Y$:

$$L^f(\sigma)\,(y) := \bigvee_{f(x)=y} \sigma(x)$$

- $L^{(-)}$ always preserves preimages ✓

L-fuzzy functor

1. Generalize $\mathbb{P}(X) = \mathbf{2}^X$

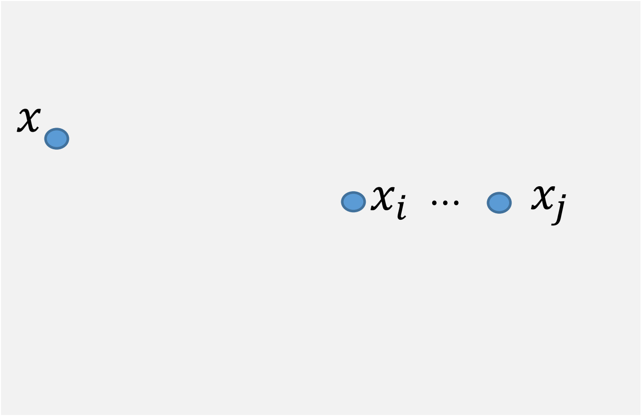
For $U \subseteq X$ put $\chi_U(x) = \begin{cases} 1, & x \in U \\ 0, & x \notin U \end{cases}$

- Replace $2 = \{0,1\}$ by a complete lattice L
  - on objects: L^X
  - on maps $f: X \to Y$:

$$L^f(\sigma)(y) := \bigvee_{f(x)=y} \sigma(x)$$

- $L^{(-)}$ always preserves preimages ✓
- $L^{(-)}$ weakly preserves kernel pairs iff L is JID ✓

$$x \wedge \bigvee\{x_i \mid i \in I\} = \bigvee\{x \wedge x_i \mid i \in I\}$$



L-fuzzy functor

1. Generalize $\mathbb{P}(X) = \mathbf{2}^X$

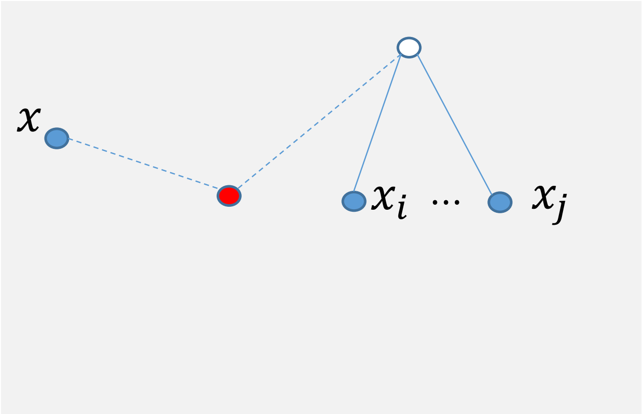
For $U \subseteq X$ put $\chi_U(x) = \begin{cases} 1, & x \in U \\ 0, & x \notin U \end{cases}$

- Replace $2 = \{0,1\}$ by a complete lattice L
  - on objects: L^X
  - on maps $f : X \to Y$:

$$L^f(\sigma)(y) := \bigvee_{f(x)=y} \sigma(x)$$

- $L^{(-)}$ always preserves preimages ✓
- $L^{(-)}$ weakly preserves kernel pairs iff L is JID ✓

$$x \wedge \bigvee\{x_i \mid i \in I\} = \bigvee\{x \wedge x_i \mid i \in I\}$$



L-fuzzy functor

1. Generalize $\mathbb{P}(X) = \mathbf{2}^X$

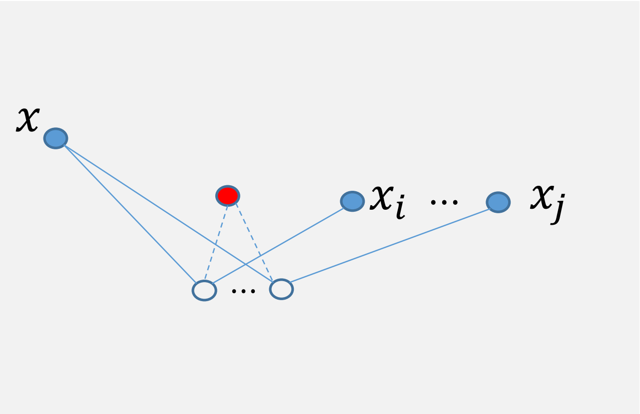
For $U \subseteq X$ put $\chi_U(x) = \begin{cases} 1, & x \in U \\ 0, & x \notin U \end{cases}$

- Replace $2 = \{0,1\}$ by a complete lattice L
  - on objects: L^X
  - on maps $f: X \to Y$:

$$L^f(\sigma)\,(y) := \bigvee_{f(x)=y} \sigma(x)$$

- $L^{(-)}$ always preserves preimages ✓
- $L^{(-)}$ weakly preserves kernel pairs iff L is JID ✓

$$x \wedge \bigvee\{x_i \mid i \in I\} = \bigvee\{x \wedge x_i \mid i \in I\}$$



L-fuzzy functor

1. Generalize $\mathbb{P}(X) = \mathbf{2}^X$

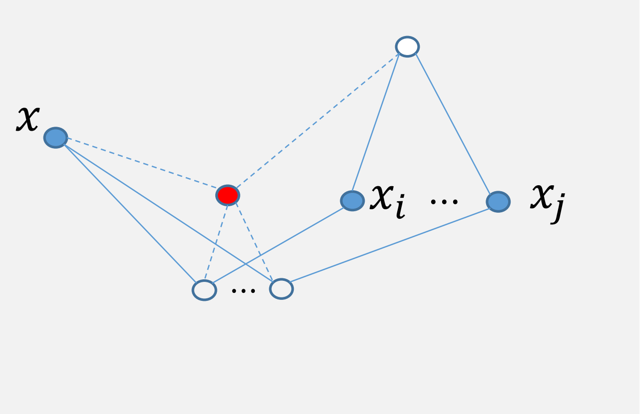
For $U \subseteq X$ put $\chi_U(x) = \begin{cases} 1, & x \in U \\ 0, & x \notin U \end{cases}$

- Replace $2 = \{0,1\}$ by a complete lattice L
  - on objects: L^X
  - on maps $f: X \to Y$:

$$L^f(\sigma)\,(y) := \bigvee_{f(x)=y} \sigma(x)$$

- $L^{(-)}$ always preserves preimages ✓
- $L^{(-)}$ weakly preserves kernel pairs iff L is JID ✓

$$x \wedge \bigvee\{x_i \mid i \in I\} = \bigvee\{x \wedge x_i \mid i \in I\}$$



Finite $\mathbb{M}$-bags

2. Generalize $\mathbb{P}_\omega(X)$

- Replace boolean algebra $\mathbf{2}$ by a commutative monoid $\mathbb{M} = (M, +, 0)$

Finite $\mathbb{M}$-bags

2. Generalize $\mathbb{P}_\omega(X)$

- Replace boolean algebra $\mathbf{2}$ by a commutative monoid $\mathbb{M} = (M, +, 0)$
  - on objects: $\mathbb{M}_\omega^X = \{m_1 x_1 + \cdots m_n x_n \mid n \in \mathbb{N}, m_i \in M. x_i \in X\}$

Finite $\mathbb{M}$-bags

2. Generalize $\mathbb{P}_\omega(X)$

- Replace boolean algebra **2** by a commutative monoid $\mathbb{M} = (M, +, 0)$
  - on objects: $\mathbb{M}_\omega^X = \{m_1 x_1 + \cdots m_n x_n \mid n \in \mathbb{N}, m_i \in \mathbb{M}. x_i \in X\}$
  - on maps $f : X \to Y$: $\mathbb{M}_\omega^f(m_1 x_1 + \cdots m_n x_n) := m_1 f x_1 + \cdots m_n f x_n$

Finite $\mathbb{M}$-bags

2. Generalize $\mathbb{P}_\omega(X)$

- Replace boolean algebra $\mathbf{2}$ by a commutative monoid $\mathbb{M} = (M, +, 0)$
    - on objects: $\mathbb{M}_\omega^X = \{m_1 x_1 + \cdots m_n x_n \mid n \in \mathbb{N}, m_i \in \mathbb{M}. x_i \in X\}$
    - on maps $f: X \to Y$: $\mathbb{M}_\omega^f(m_1 x_1 + \cdots m_n x_n) := m_1 f x_1 + \cdots m_n f x_n$
- The functor $\mathbb{M}_\omega^{(-)}$ (weakly) preserves

Finite $\mathbb{M}$-bags

2. Generalize $\mathbb{P}_\omega(X)$

- Replace boolean algebra $\mathbf{2}$ by a commutative monoid $\mathbb{M} = (M, +, 0)$
  - on objects: $\mathbb{M}_\omega^X = \{m_1 x_1 + \cdots m_n x_n \mid n \in \mathbb{N}, m_i \in \mathbb{M}. x_i \in X\}$
  - on maps $f: X \to Y$: $\mathbb{M}_\omega^f(m_1 x_1 + \cdots m_n x_n) := m_1 f x_1 + \cdots m_n f x_n$
- The functor $\mathbb{M}_\omega^{(-)}$ (weakly) preserves
  - preimages iff $\mathbb{M}$ is positive

Finite $\mathbb{M}$-bags

2. Generalize $\mathbb{P}_\omega(X)$

- Replace boolean algebra **2** by a commutative monoid $\mathbb{M} = (M, +, 0)$
  - on objects: $\mathbb{M}_\omega^X = \{m_1 x_1 + \cdots m_n x_n \mid n \in \mathbb{N}, m_i \in \mathbb{M}. x_i \in X\}$
  - on maps $f: X \to Y$: $\mathbb{M}_\omega^f(m_1 x_1 + \cdots m_n x_n) := m_1 f x_1 + \cdots m_n f x_n$

positive:
$x + y = 0 \quad \Rightarrow \quad x = 0$

- The functor $\mathbb{M}_\omega^{(-)}$ (weakly) preserves
  - preimages iff $\mathbb{M}$ is positive

Finite $\mathbb{M}$-bags

2. Generalize $\mathbb{P}_\omega(X)$

- Replace boolean algebra **2** by a commutative monoid $\mathbb{M} = (M, +, 0)$
  - on objects: $\mathbb{M}_\omega^X = \{m_1 x_1 + \cdots m_n x_n \mid n \in \mathbb{N}, m_i \in \mathbb{M}. x_i \in X\}$
  - on maps $f: X \to Y$: $\mathbb{M}_\omega^f(m_1 x_1 + \cdots m_n x_n) := m_1 f x_1 + \cdots m_n f x_n$

positive:
$x + y = 0 \quad \Rightarrow \quad x = 0$

- The functor $\mathbb{M}_\omega^{(-)}$ (weakly) preserves
  - preimages iff $\mathbb{M}$ is positive ✓
  - kernel pairs iff $\mathbb{M}$ is refinable

Finite $\mathbb{M}$-bags

2. Generalize $\mathbb{P}_\omega(X)$

- Replace boolean algebra **2** by a commutative monoid $\mathbb{M} = (M, +, 0)$
  - on objects: $\mathbb{M}_\omega^X = \{m_1 x_1 + \cdots m_n x_n \mid n \in \mathbb{N}, m_i \in \mathbb{M}. x_i \in X\}$
  - on maps $f: X \to Y$: $\mathbb{M}_\omega^f(m_1 x_1 + \cdots m_n x_n) := m_1 f x_1 + \cdots m_n f x_n$
- The functor $\mathbb{M}_\omega^{(-)}$ (weakly) preserves
  - preimages iff $\mathbb{M}$ is positive ✓
  - kernel pairs iff $\mathbb{M}$ is refinable

positive:
$x + y = 0 \quad \Rightarrow \quad x = 0$

refinable:
$x_1 + x_2 = y_1 + y_2 \Rightarrow$

			x_1
			x_2
y_1	y_2		$=$

Finite $\mathbb{M}$-bags

2. Generalize $\mathbb{P}_\omega(X)$

- Replace boolean algebra **2** by a commutative monoid $\mathbb{M} = (M, +, 0)$
  - on objects: $\mathbb{M}_\omega^X = \{m_1 x_1 + \cdots m_n x_n \mid n \in \mathbb{N}, m_i \in \mathbb{M}. x_i \in X\}$
  - on maps $f: X \to Y$: $\mathbb{M}_\omega^f(m_1 x_1 + \cdots m_n x_n) := m_1 f x_1 + \cdots m_n f x_n$
- The functor $\mathbb{M}_\omega^{(-)}$ (weakly) preserves
  - preimages iff $\mathbb{M}$ is positive ✓
  - kernel pairs iff $\mathbb{M}$ is refinable ✓

positive:
$x + y = 0 \quad \Rightarrow \quad x = 0$

refinable:
$x_1 + x_2 = y_1 + y_2 \Rightarrow$

a	b	x_1
c	d	x_2
y_1	y_2	$=$

Intro

State based systems

Functors and Coalgebras

Functor properties

Weak Pullback Preservation

Functors parameterized by algebras

Free-algebra functor

Preimage preservation

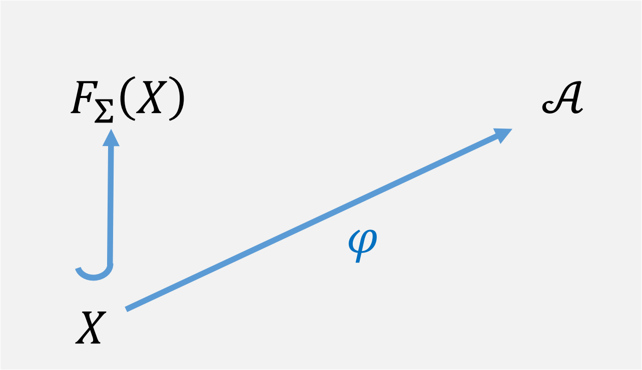
Weak kernel preservation

Conclusion

- and Breaking News

Properties of $F_\Sigma(X)$

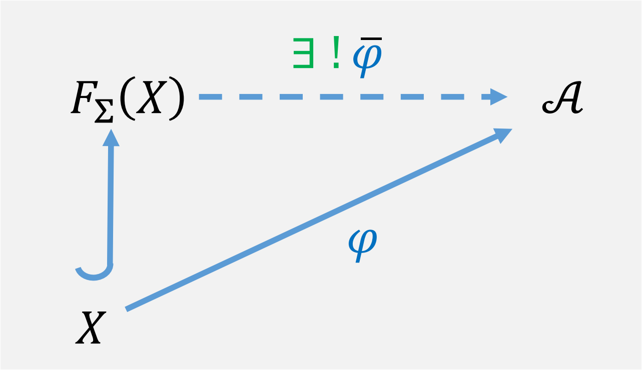
- Suppose $\mathcal{A}$ satisfies the equations Σ



Properties of $F_\Sigma(X)$

- Suppose $\mathcal{A}$ satisfies the equations Σ
- Each map $\varphi: X \to A$ has unique homomorphic extension

$$\bar{\varphi}: F_\Sigma(X) \to \mathcal{A}$$

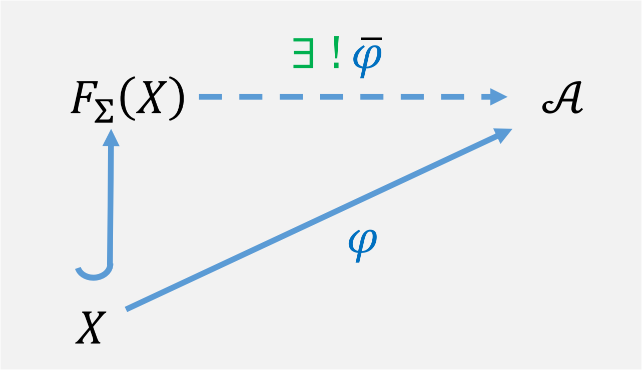


Properties of $F_\Sigma(X)$

- Suppose $\mathcal{A}$ satisfies the equations Σ
- Each map $\varphi: X \to A$ has unique homomorphic extension

$$\bar{\varphi}:\ F_\Sigma(X) \to \mathcal{A}$$

  - $\bar{\varphi}\ t(x_1, \ldots, x_n) = t^A(\varphi x_1, \ldots, \varphi x_n)$

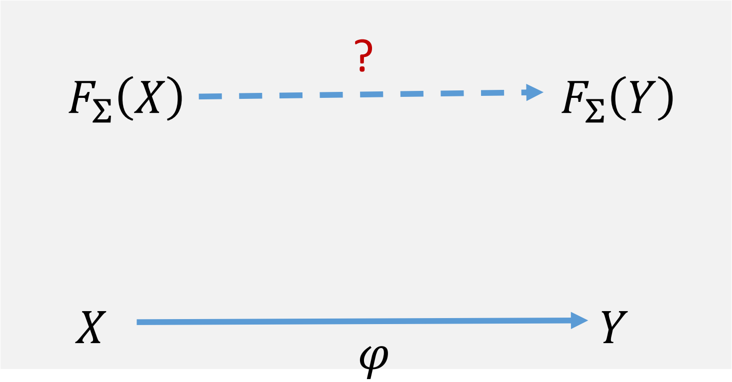


Properties of $F_\Sigma(X)$

- Suppose $\mathcal{A}$ satisfies the equations Σ
- Each map $\varphi: X \to A$ has unique homomorphic extension

$$\bar{\varphi}\colon\ F_\Sigma(X) \to \mathcal{A}$$

  - $\bar{\varphi}\ t(x_1, \dots, x_n) = t^A(\varphi x_1, \dots, \varphi x_n)$
- $F_\Sigma(-)$ is a Set-functor



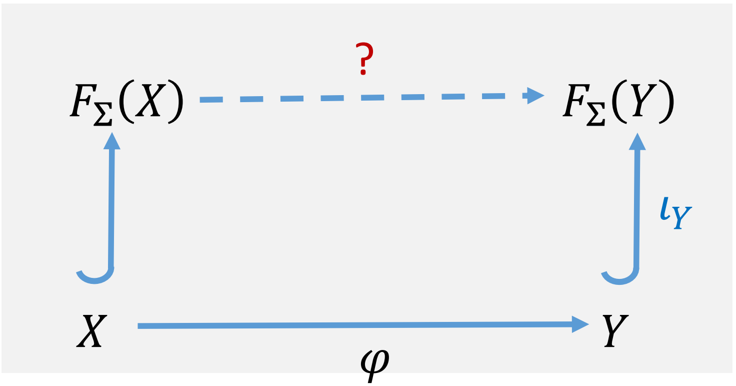
Properties of $F_\Sigma(X)$

- Suppose $\mathcal{A}$ satisfies the equations Σ
- Each map $\varphi: X \to A$ has unique homomorphic extension

$$\bar{\varphi}\colon\ F_\Sigma(X) \to \mathcal{A}$$

  - $\bar{\varphi}\ t(x_1, \dots, x_n) = t^A(\varphi x_1, \dots, \varphi x_n)$

- $F_\Sigma(-)$ is a Set-functor



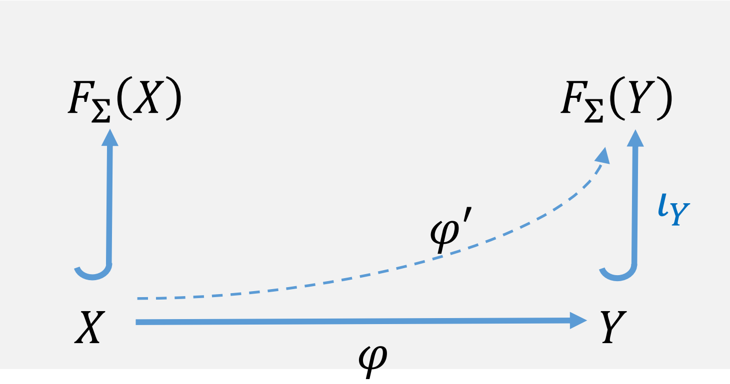
Properties of $F_\Sigma(X)$

- Suppose $\mathcal{A}$ satisfies the equations Σ
- Each map $\varphi: X \to A$ has unique homomorphic extension

$$\bar{\varphi}:\ F_\Sigma(X) \to \mathcal{A}$$

  - $\bar{\varphi}\ t(x_1, \dots, x_n) = t^A(\varphi x_1, \dots, \varphi x_n)$

- $F_\Sigma(-)$ is a Set-functor

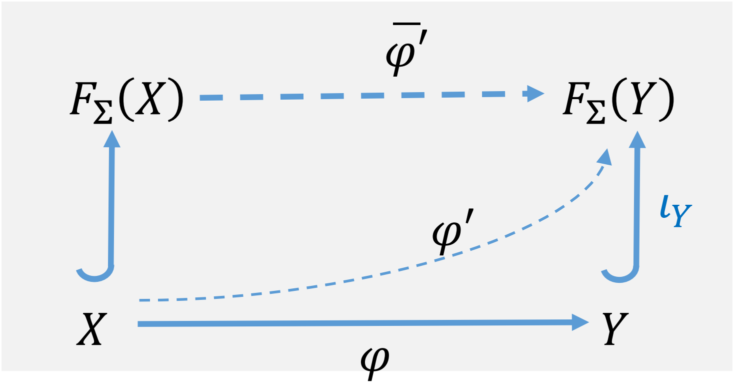


Properties of $F_\Sigma(X)$

- Suppose $\mathcal{A}$ satisfies the equations Σ
- Each map $\varphi: X \to A$ has unique homomorphic extension

$$\bar{\varphi}:\ F_\Sigma(X) \to \mathcal{A}$$

  - $\bar{\varphi}\ t(x_1, \dots, x_n) = t^A(\varphi x_1, \dots, \varphi x_n)$
- $F_\Sigma(-)$ is a Set-functor



Intro

State based systems
Functors and Coalgebras

Functor properties

Weak Pullback Preservation
Functors parameterized by algebras

Free-algebra functor

Preimage preservation
Weak kernel preservation

Conclusion

- and Breaking News

Independence

- $p(x, v_1, \dots, v_n)$ *weakly independent* of x if

$$p(x, v_1, \dots, v_n) \approx q(y) \qquad \text{where } x \neq y$$

Independence

- $p(x, v_1, \ldots, v_n)$ *weakly independent* of x if

  $p(x, v_1, \ldots, v_n) \approx q(y)$ where $x \neq y$

- $p(x, z_1, \ldots, z_n)$ *independent* of x, if

  $p(x, z_1, \ldots, z_n) \approx p(y, z_1, \ldots, z_n)$ where $x, y, z_1, \ldots, z_n$ mutually different

Independence

- $p(x, v_1, \ldots, v_n)$ *weakly independent* of x if

  $p(x, v_1, \ldots, v_n) \approx q(y)$ where $x \neq y$

- $p(x, z_1, \ldots, z_n)$ *independent* of x, if

  $p(x, z_1, \ldots, z_n) \approx p(y, z_1, \ldots, z_n)$ where $x, y, z_1, \ldots, z_n$ mutually different

Derivative of Σ:

$$\Sigma' \coloneqq \{p(x, \vec{z}) \approx p(y, \vec{z}) \mid p(x, \vec{v}) \text{ weakly independent of } x\}$$

Preservation of preimages

- Thm: $F_{\Sigma}(-)$ preserves preimages iff $\Sigma \vdash \Sigma'$

Preservation of preimages

- Thm: $F_{\Sigma}(-)$ preserves preimages iff $\Sigma \vdash \Sigma'$
- Proof($\Rightarrow$)

Preservation of preimages

- Thm: $F_\Sigma(-)$ preserves preimages iff $\Sigma \vdash \Sigma'$

- Proof($\Rightarrow$)

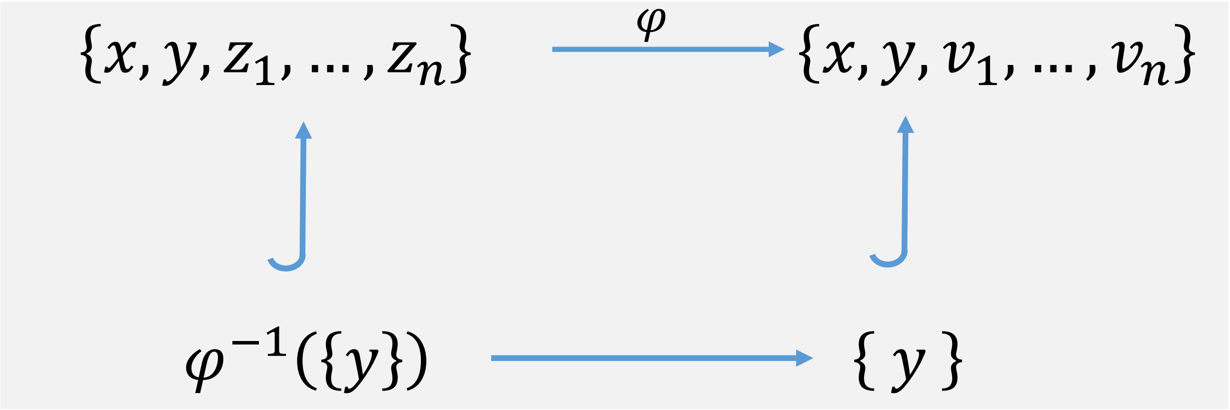
$$p(x, v_1, \dots, v_n) \approx q(y) \quad \vdash \quad p(x, z_1, \dots, z_n) \approx p(y, z_1, \dots, z_n)$$

Preservation of preimages

- <u>Thm</u>: $F_{\Sigma}(-)$ preserves preimages iff $\Sigma \vdash \Sigma'$

- Proof($\Rightarrow$)

$$p(x, v_1, \ldots, v_n) \approx q(y) \quad \vdash \quad p(x, z_1, \ldots, z_n) \approx p(y, z_1, \ldots, z_n)$$

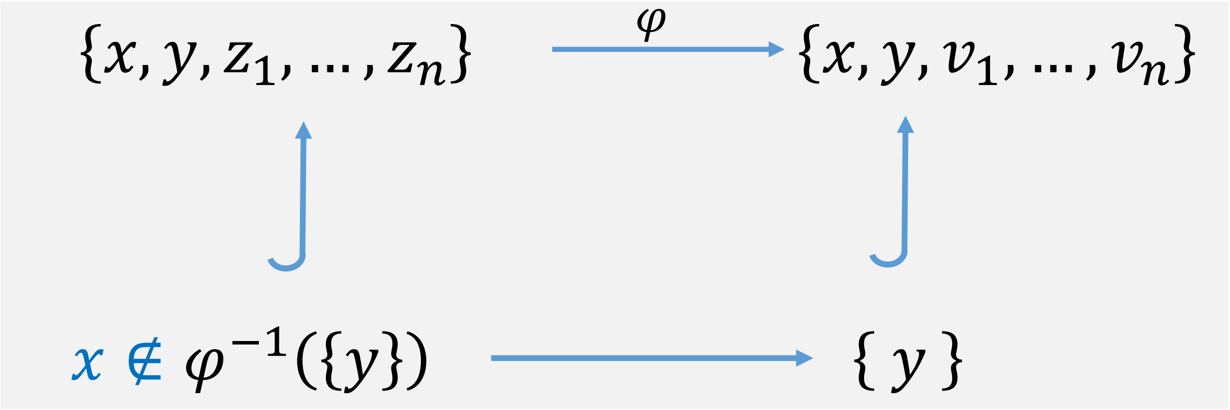


Preservation of preimages

- Thm: $F_\Sigma(-)$ preserves preimages iff $\Sigma \vdash \Sigma'$

- Proof($\Rightarrow$)

$$p(x, v_1, \dots, v_n) \approx q(y) \quad \vdash \quad p(x, z_1, \dots, z_n) \approx p(y, z_1, \dots, z_n)$$



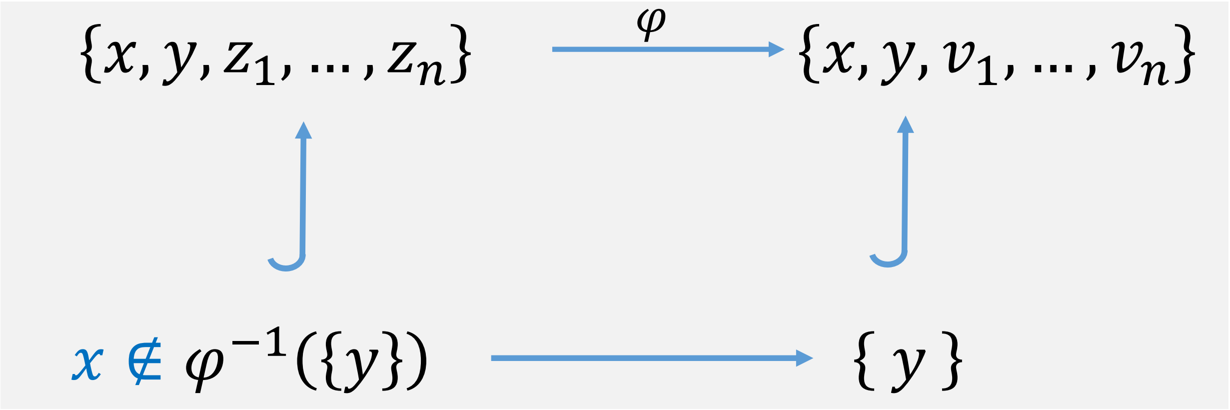
Preservation of preimages

- <u>Thm</u>: $F_\Sigma(-)$ preserves preimages iff $\Sigma \vdash \Sigma'$

- Proof($\Rightarrow$)

$$p(x, v_1, \dots, v_n) \approx q(y) \quad \vdash \quad p(x, z_1, \dots, z_n) \approx p(y, z_1, \dots, z_n)$$

$$\bar{\varphi}p(x, z_1, \dots, z_n) \approx p(\varphi x, \varphi z_1, \dots, \varphi z_n)$$



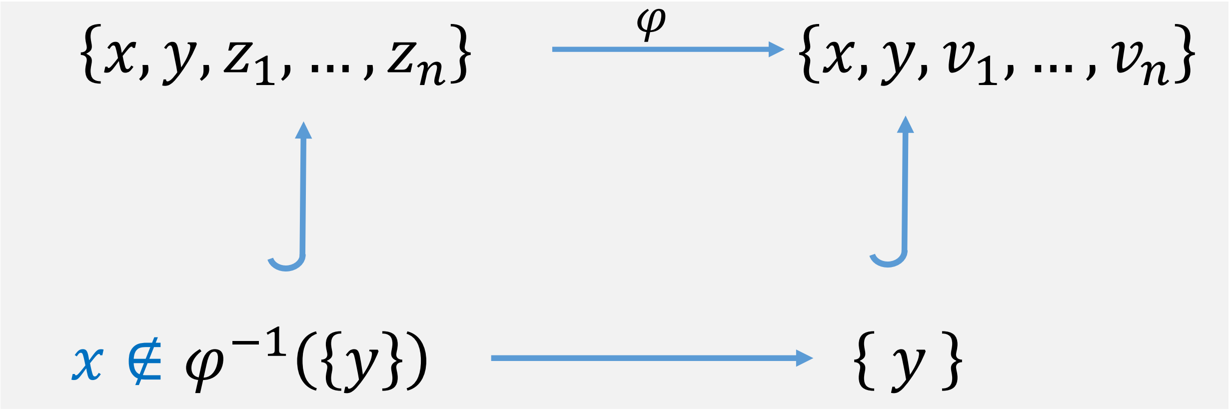
Preservation of preimages

- Thm: $F_\Sigma(-)$ preserves preimages iff $\Sigma \vdash \Sigma'$

- Proof($\Rightarrow$)

$$p(x, v_1, \dots, v_n) \approx q(y) \quad \vdash \quad p(x, z_1, \dots, z_n) \approx p(y, z_1, \dots, z_n)$$

$$\begin{aligned}\bar{\varphi}p(x, z_1, \dots, z_n) &\approx p(\varphi x, \varphi z_1, \dots, \varphi z_n) \\ &\approx p(x, v_1, \dots, v_n)\end{aligned}$$



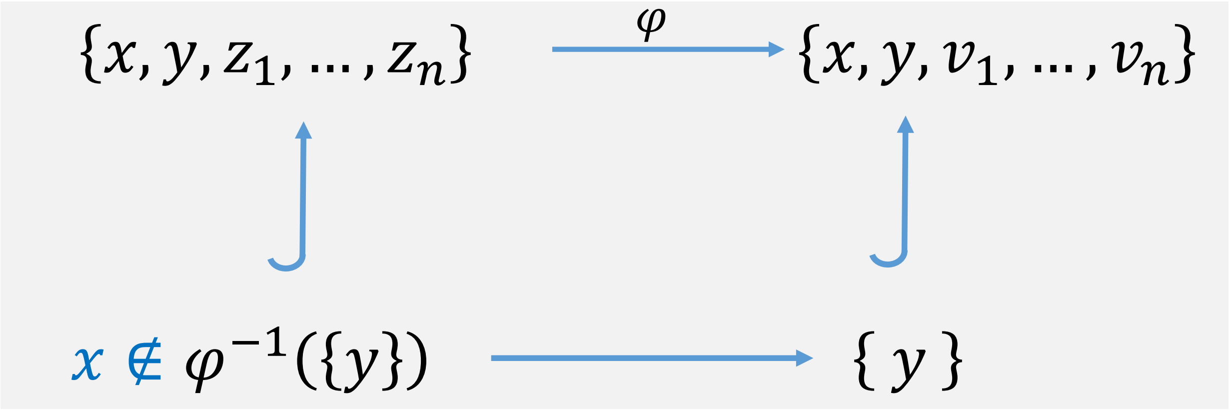
Preservation of preimages

- <u>Thm</u>: $F_\Sigma(-)$ preserves preimages iff $\Sigma \vdash \Sigma'$

- Proof($\Rightarrow$)

$$p(x, v_1, \dots, v_n) \approx q(y) \quad \vdash \quad p(x, z_1, \dots, z_n) \approx p(y, z_1, \dots, z_n)$$

$$\begin{aligned}\bar{\varphi}p(x, z_1, \dots, z_n) &\approx p(\varphi x, \varphi z_1, \dots, \varphi z_n)\\ &\approx p(x, v_1, \dots, v_n)\\ &\approx q(y)\end{aligned}$$



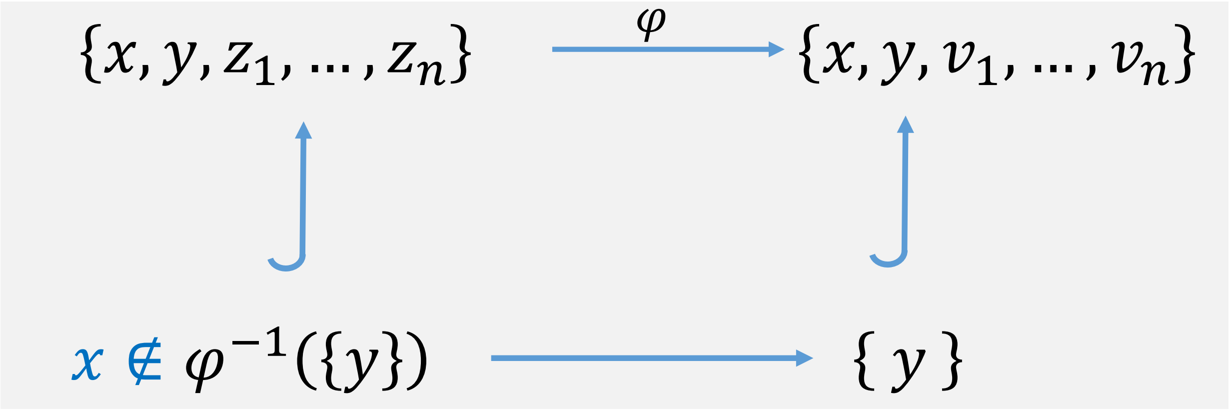
Preservation of preimages

- <u>Thm</u>: $F_\Sigma(-)$ preserves preimages iff $\Sigma \vdash \Sigma'$

- Proof($\Rightarrow$)

$$p(x, v_1, \dots, v_n) \approx q(y) \quad \vdash \quad p(x, z_1, \dots, z_n) \approx p(y, z_1, \dots, z_n)$$

$$\begin{aligned}\bar{\varphi} p(x, z_1, \dots, z_n) &\approx p(\varphi x, \varphi z_1, \dots, \varphi z_n) \\ &\approx p(x, v_1, \dots, v_n) \\ &\approx q(y) \qquad \in F_\Sigma(\{y\})\end{aligned}$$



Preservation of preimages

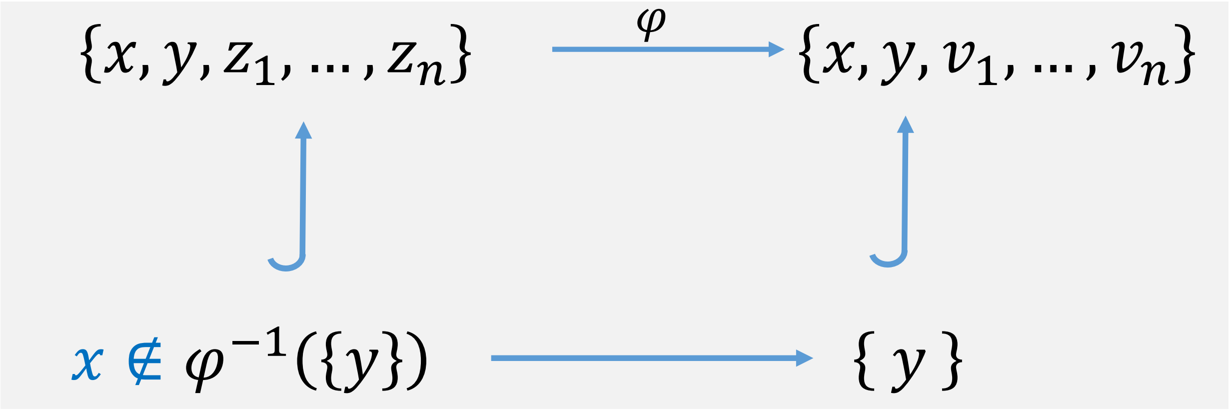
- Thm: $F_\Sigma(-)$ preserves preimages iff $\Sigma \vdash \Sigma'$

- Proof($\Rightarrow$)

$$p(x, v_1, \dots, v_n) \approx q(y) \quad \vdash \quad p(x, z_1, \dots, z_n) \approx p(y, z_1, \dots, z_n)$$

$$\begin{aligned} \bar{\varphi} p(x, z_1, \dots, z_n) &\approx p(\varphi x, \varphi z_1, \dots, \varphi z_n) \\ &\approx p(x, v_1, \dots, v_n) \\ &\approx q(y) \qquad \in F_\Sigma(\{y\}) \end{aligned}$$

$$\Rightarrow p(x, z_1, \dots, z_n) \in \bar{\varphi}^{-1}(F_\Sigma(\{y\}))$$



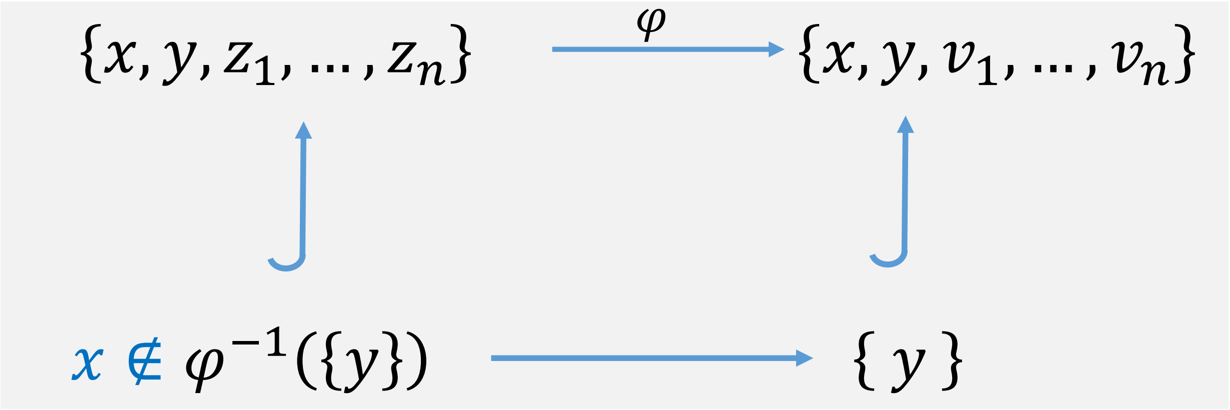
Preservation of preimages

- <u>Thm</u>: $F_\Sigma(-)$ preserves preimages iff $\Sigma \vdash \Sigma'$

- Proof($\Rightarrow$)

$$p(x, v_1, \dots, v_n) \approx q(y) \quad \vdash \quad p(x, z_1, \dots, z_n) \approx p(y, z_1, \dots, z_n)$$

$$\begin{aligned} \bar{\varphi} p(x, z_1, \dots, z_n) &\approx p(\varphi x, \varphi z_1, \dots, \varphi z_n) \\ &\approx p(x, v_1, \dots, v_n) \\ &\approx q(y) \quad \in F_\Sigma(\{y\}) \end{aligned}$$



$$\begin{aligned} \Rightarrow p(x, z_1, \dots, z_n) &\in \bar{\varphi}^{-1}(F_\Sigma(\{y\})) \\ &= F_\Sigma(\varphi^{-1}\{y\}) \end{aligned}$$

Preservation of preimages

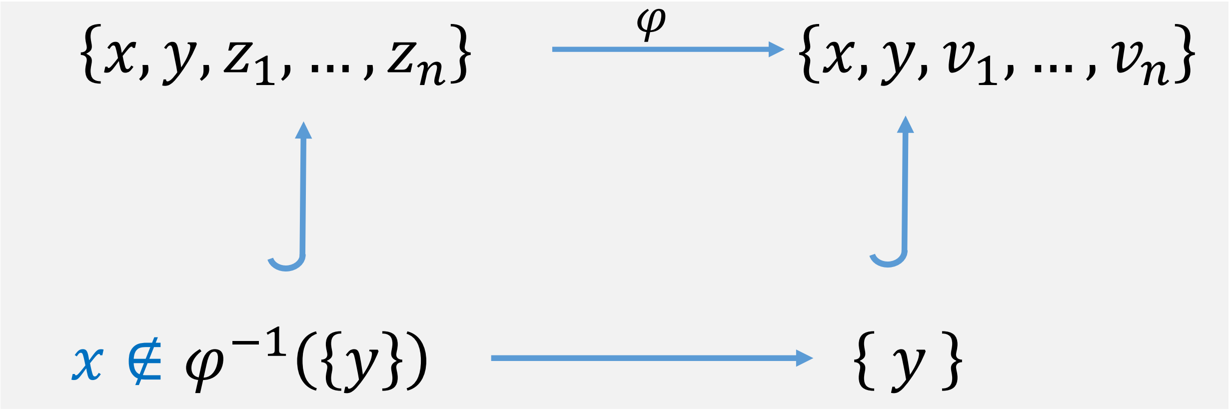
- Thm: $F_\Sigma(-)$ preserves preimages iff $\Sigma \vdash \Sigma'$

- Proof($\Rightarrow$)

$$p(x, v_1, \dots, v_n) \approx q(y) \quad \vdash \quad p(x, z_1, \dots, z_n) \approx p(y, z_1, \dots, z_n)$$

$$\begin{aligned} \bar{\varphi} p(x, z_1, \dots, z_n) &\approx p(\varphi x, \varphi z_1, \dots, \varphi z_n) \\ &\approx p(x, v_1, \dots, v_n) \\ &\approx q(y) \quad \in F_\Sigma(\{y\}) \end{aligned}$$

$$\begin{aligned} \Rightarrow p(x, z_1, \dots, z_n) &\in \bar{\varphi}^{-1}(F_\Sigma(\{y\})) \\ &= F_\Sigma(\varphi^{-1}\{y\}) \\ &\subseteq F_\Sigma(\{y, z_1, \dots, z_n\}) \end{aligned}$$



Preservation of preimages

- Thm: $F_\Sigma(-)$ preserves preimages iff $\Sigma \vdash \Sigma'$

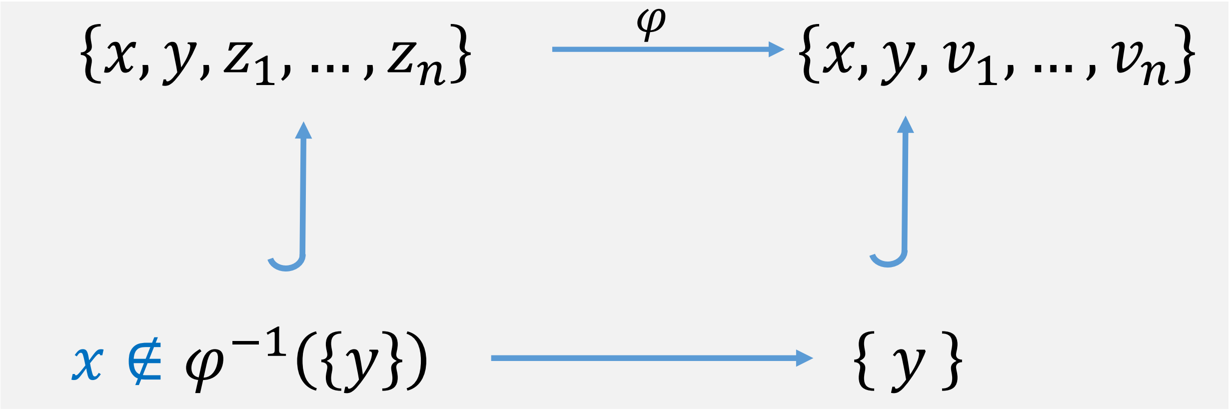
- Proof($\Rightarrow$)

$$p(x, v_1, \dots, v_n) \approx q(y) \quad \vdash \quad p(x, z_1, \dots, z_n) \approx p(y, z_1, \dots, z_n)$$

$$\begin{aligned} \bar{\varphi} p(x, z_1, \dots, z_n) &\approx p(\varphi x, \varphi z_1, \dots, \varphi z_n) \\ &\approx p(x, v_1, \dots, v_n) \\ &\approx q(y) \quad \in F_\Sigma(\{y\}) \end{aligned}$$

$$\begin{aligned} \Rightarrow p(x, z_1, \dots, z_n) &\in \bar{\varphi}^{-1}(F_\Sigma(\{y\})) \\ &= F_\Sigma(\varphi^{-1}\{y\}) \\ &\subseteq F_\Sigma(\{y, z_1, \dots, z_n\}) \end{aligned}$$

$$\Rightarrow \quad p(x, z_1, \dots, z_n) \approx r(y, z_1, \dots, z_n)$$



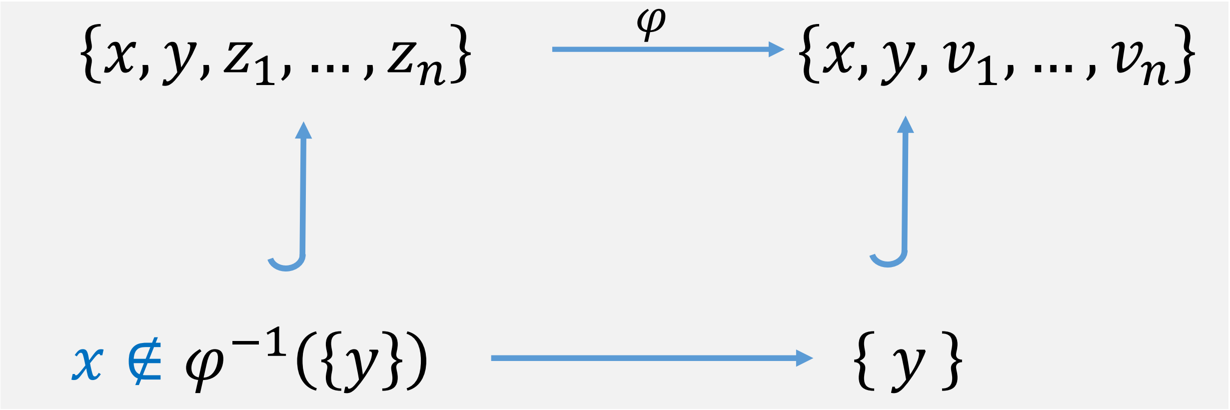
Preservation of preimages

- <u>Thm</u>: $F_\Sigma(-)$ preserves preimages iff $\Sigma \vdash \Sigma'$

- Proof($\Rightarrow$)

$$p(x, v_1, \dots, v_n) \approx q(y) \quad \vdash \quad p(x, z_1, \dots, z_n) \approx p(y, z_1, \dots, z_n)$$

$$\begin{aligned}
\bar{\varphi} p(x, z_1, \dots, z_n) &\approx p(\varphi x, \varphi z_1, \dots, \varphi z_n) \\
&\approx p(x, v_1, \dots, v_n) \\
&\approx q(y) \qquad \in F_\Sigma(\{y\})
\end{aligned}$$



$$\begin{aligned}
\Rightarrow p(x, z_1, \dots, z_n) \in \; &\bar{\varphi}^{-1}(F_\Sigma(\{y\})) \\
&= F_\Sigma(\varphi^{-1}\{y\}) \\
&\subseteq F_\Sigma(\{y, z_1, \dots, z_n\})
\end{aligned}$$

$$\Rightarrow \quad p(x, z_1, \dots, z_n) \approx r(y, z_1, \dots, z_n) \approx r(x, z_1, \dots, z_n)$$

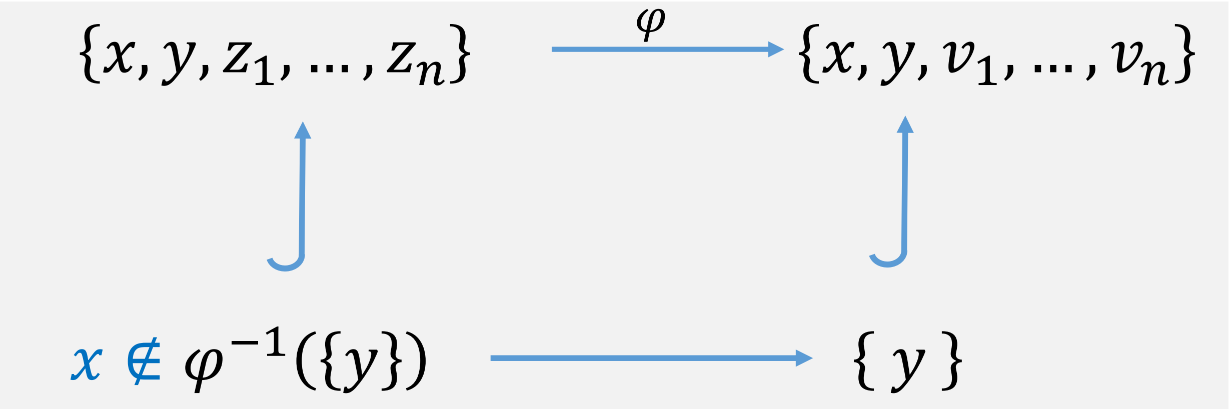
Preservation of preimages

- <u>Thm</u>: $F_\Sigma(-)$ preserves preimages iff $\Sigma \vdash \Sigma'$

- Proof($\Rightarrow$)

$$p(x, v_1, \dots, v_n) \approx q(y) \quad \vdash \quad p(x, z_1, \dots, z_n) \approx p(y, z_1, \dots, z_n)$$

$$\begin{aligned}\bar{\varphi} p(x, z_1, \dots, z_n) &\approx p(\varphi x, \varphi z_1, \dots, \varphi z_n) \\ &\approx p(x, v_1, \dots, v_n) \\ &\approx q(y) \quad \in F_\Sigma(\{y\})\end{aligned}$$



$$\begin{aligned}\Rightarrow p(x, z_1, \dots, z_n) &\in \bar{\varphi}^{-1}(F_\Sigma(\{y\})) \\ &= F_\Sigma(\varphi^{-1}\{y\}) \\ &\subseteq F_\Sigma(\{y, z_1, \dots, z_n\})\end{aligned}$$

$$\Rightarrow \quad p(x, z_1, \dots, z_n) \approx r(y, z_1, \dots, z_n) \approx r(x, z_1, \dots, z_n) \approx p(y, z_1, \dots, z_n) \checkmark$$

Preservation of preimages

- Thm: $F_{\Sigma}(-)$ preserves preimages iff $\Sigma \vdash \Sigma'$
- Proof($\Leftarrow$)

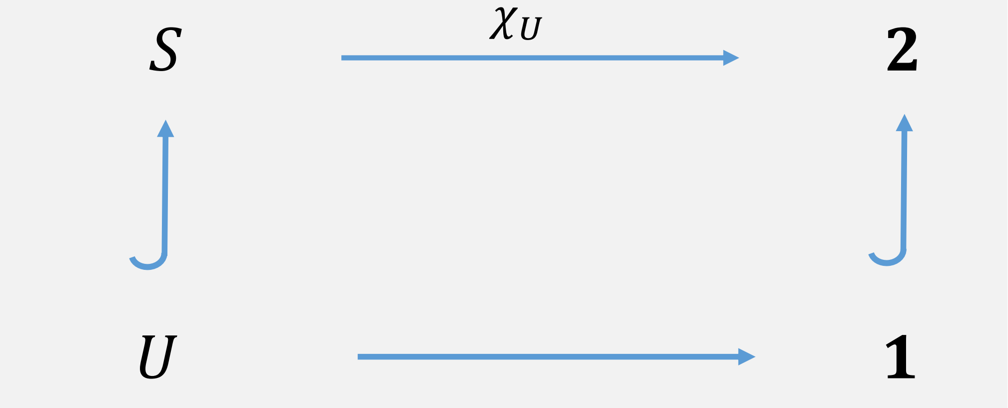
Preservation of preimages

- Thm: $F_{\Sigma}(-)$ preserves preimages iff $\Sigma \vdash \Sigma'$
- Proof($\Leftarrow$) Enough to consider classifying preimages

Preservation of preimages

- <u>Thm</u>: $F_\Sigma(-)$ preserves preimages iff $\Sigma \vdash \Sigma'$

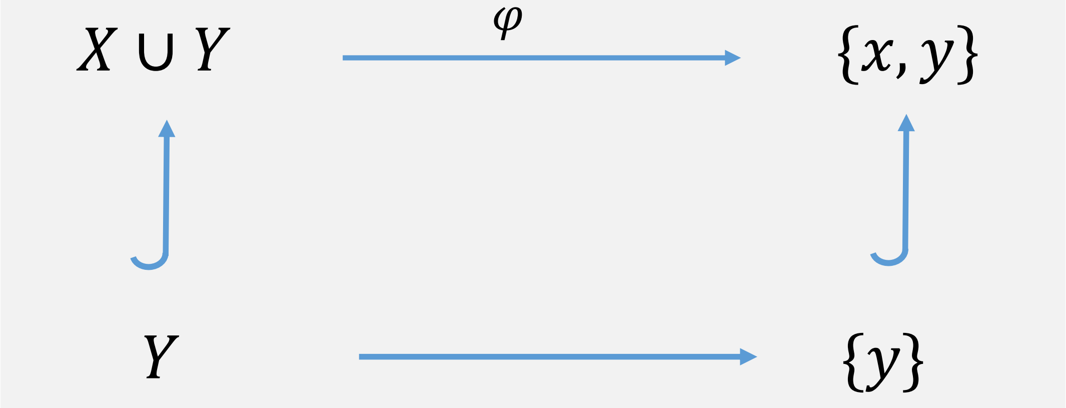
- Proof($\Leftarrow$) Enough to consider classifying preimages



Preservation of preimages

- <u>Thm</u>: $F_{\Sigma}(-)$ preserves preimages iff $\Sigma \vdash \Sigma'$

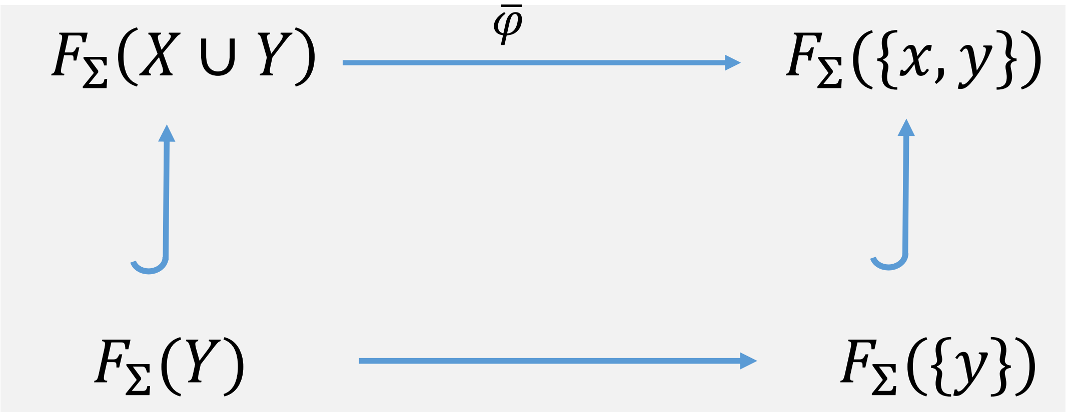
- Proof($\Leftarrow$) Enough to consider classifying preimages



Preservation of preimages

- Thm: $F_\Sigma(-)$ preserves preimages iff $\Sigma \vdash \Sigma'$

- Proof($\Leftarrow$) Enough to consider classifying preimages

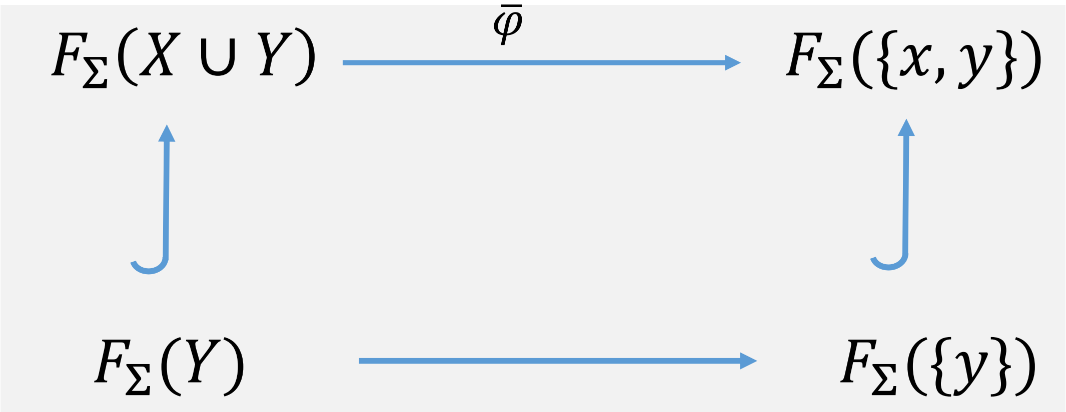


Preservation of preimages

- Thm: $F_\Sigma(-)$ preserves preimages iff $\Sigma \vdash \Sigma'$

- Proof($\Leftarrow$) Enough to consider classifying preimages

  Given $p \in F_\Sigma(X \cup Y)$ with $\bar{\varphi}p \in F_\Sigma(\{y\})$ show $p \in F_\Sigma(Y)$



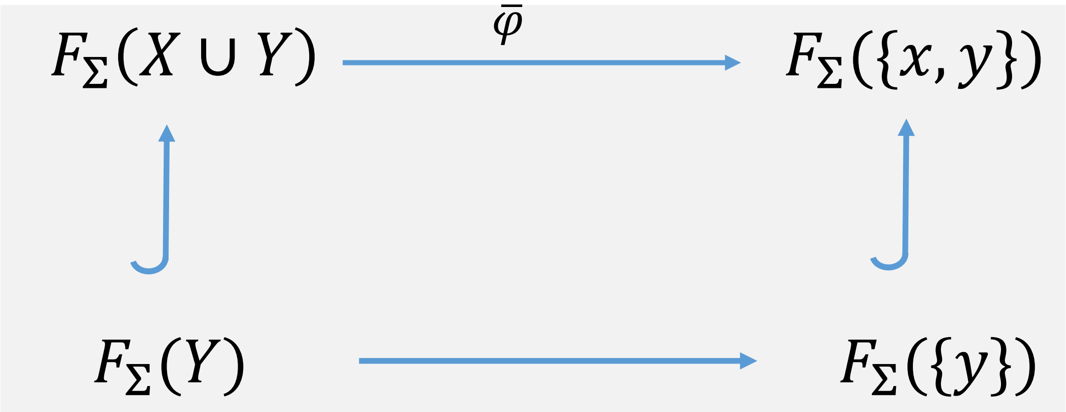
Preservation of preimages

- <u>Thm</u>: $F_\Sigma(-)$ preserves preimages iff $\Sigma \vdash \Sigma'$

- Proof($\Leftarrow$) Enough to consider classifying preimages

  Given $p \in F_\Sigma(X \cup Y)$ with $\bar{\varphi}p \in F_\Sigma(\{y\})$ show $p \in F_\Sigma(Y)$

$$p \quad = \quad p(x_1, \dots, x_n, y_1, \dots, y_m)$$



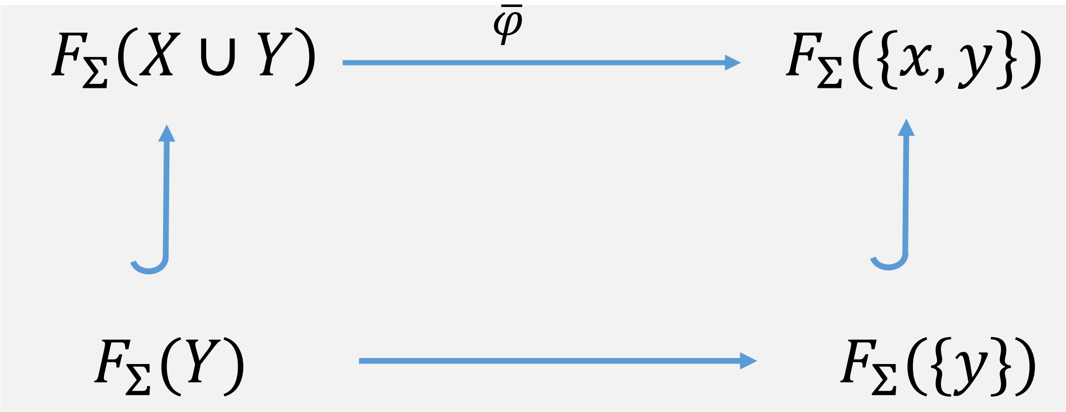
Preservation of preimages

- <u>Thm</u>: $F_\Sigma(-)$ preserves preimages iff $\Sigma \vdash \Sigma'$

- Proof($\Leftarrow$) Enough to consider classifying preimages

  Given $p \in F_\Sigma(X \cup Y)$ with $\bar{\varphi}p \in F_\Sigma(\{y\})$ show $p \in F_\Sigma(Y)$

$$\bar{\varphi}p \quad = \bar{\varphi}p(x_1, \dots, x_n, y_1, \dots, y_m)$$



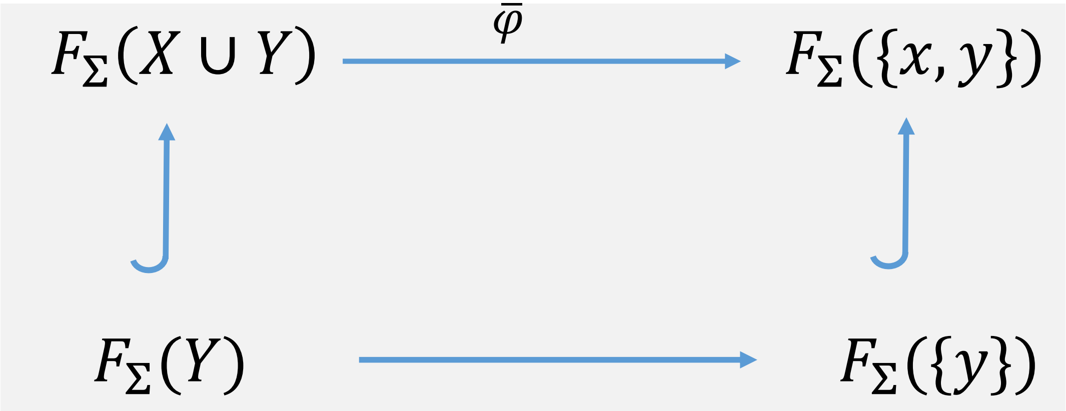
Preservation of preimages

- Thm: $F_\Sigma(-)$ preserves preimages iff $\Sigma \vdash \Sigma'$

- Proof($\Leftarrow$) Enough to consider classifying preimages

  Given $p \in F_\Sigma(X \cup Y)$ with $\bar{\varphi} p \in F_\Sigma(\{y\})$ show $p \in F_\Sigma(Y)$

$$\bar{\varphi} p = \bar{\varphi} p(x_1, \dots, x_n, y_1, \dots, y_m) = p(x, \dots, x, y, \dots, y)$$



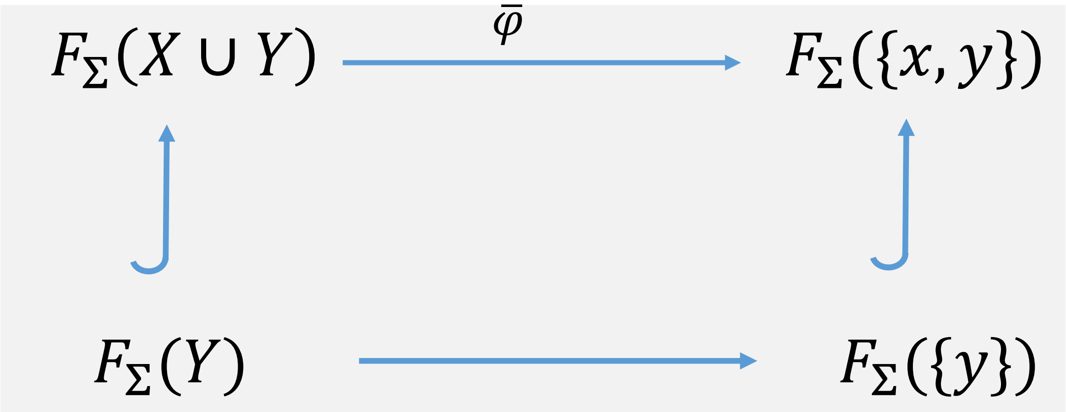
Preservation of preimages

- Thm: $F_{\Sigma}(-)$ preserves preimages iff $\Sigma \vdash \Sigma'$

- Proof($\Leftarrow$) Enough to consider classifying preimages

  Given $p \in F_{\Sigma}(X \cup Y)$ with $\bar{\varphi}p \in F_{\Sigma}(\{y\})$ show $p \in F_{\Sigma}(Y)$

$$\begin{aligned} \bar{\varphi}p \quad &= \bar{\varphi}p(x_1, \dots, x_n, y_1, \dots, y_m) \\ &= p(x, \dots, x, y, \dots, y) \approx q(y) \end{aligned}$$



Preservation of preimages

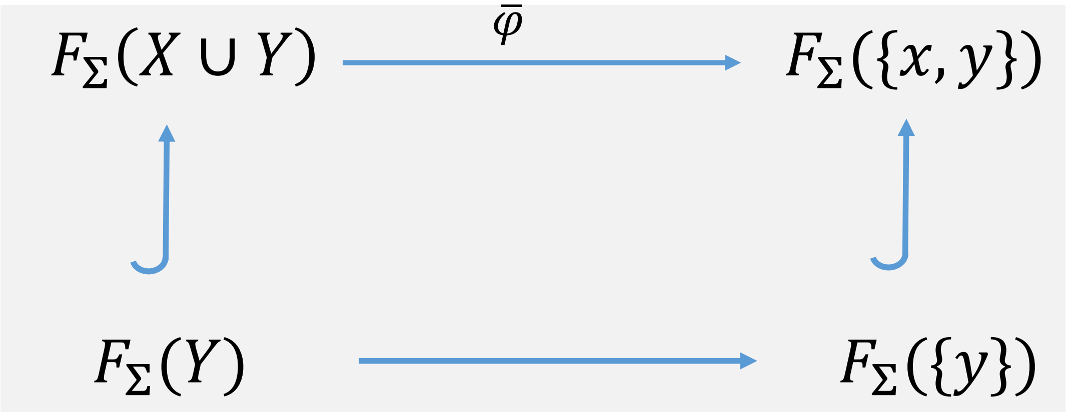
- <u>Thm</u>: $F_\Sigma(-)$ preserves preimages iff $\Sigma \vdash \Sigma'$

- Proof($\Leftarrow$) Enough to consider classifying preimages

  Given $p \in F_\Sigma(X \cup Y)$ with $\bar{\varphi}p \in F_\Sigma(\{y\})$ show $p \in F_\Sigma(Y)$

$$\bar{\varphi}p(x_1, \dots, x_n, y_1, \dots, y_m)$$
$$= p(x, \dots, x, y, \dots, y) \approx q(y)$$

$$p(x_1, x_2, \dots, x_n, y_1, \dots, y_m)$$
$$\approx$$
$$\approx$$
$$\approx$$
$$\approx$$



Preservation of preimages

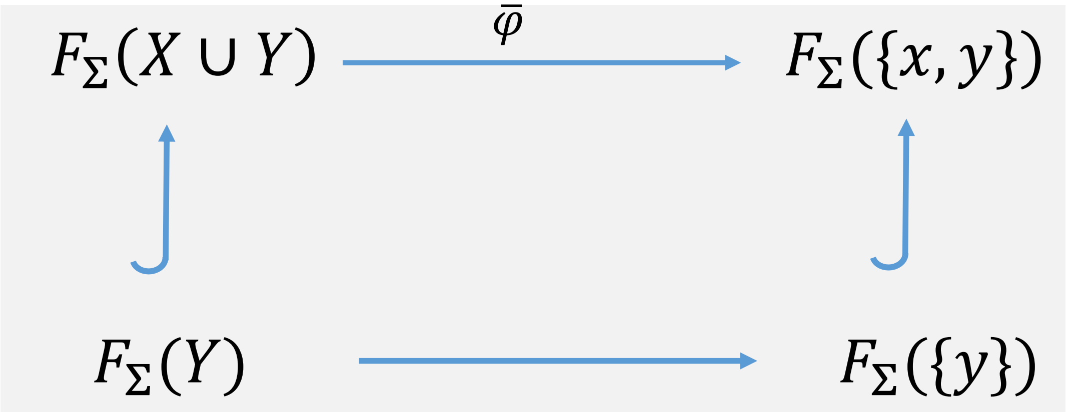
- <u>Thm</u>: $F_\Sigma(-)$ preserves preimages iff $\Sigma \vdash \Sigma'$

- Proof($\Leftarrow$) Enough to consider classifying preimages

  Given $p \in F_\Sigma(X \cup Y)$ with $\bar{\varphi} p \in F_\Sigma(\{y\})$ show $p \in F_\Sigma(Y)$

$$\bar{\varphi} p(x_1, \dots, x_n, y_1, \dots, y_m)$$
$$= p(x, \dots, x, y, \dots, y) \approx q(y)$$

$$p(x_1, x_2, \dots, x_n, y_1, \dots, y_m)$$
$$\approx p(z_1, x_2, \dots, x_n, y_1, \dots, y_m)$$
$$\approx$$
$$\approx$$
$$\approx$$



Preservation of preimages

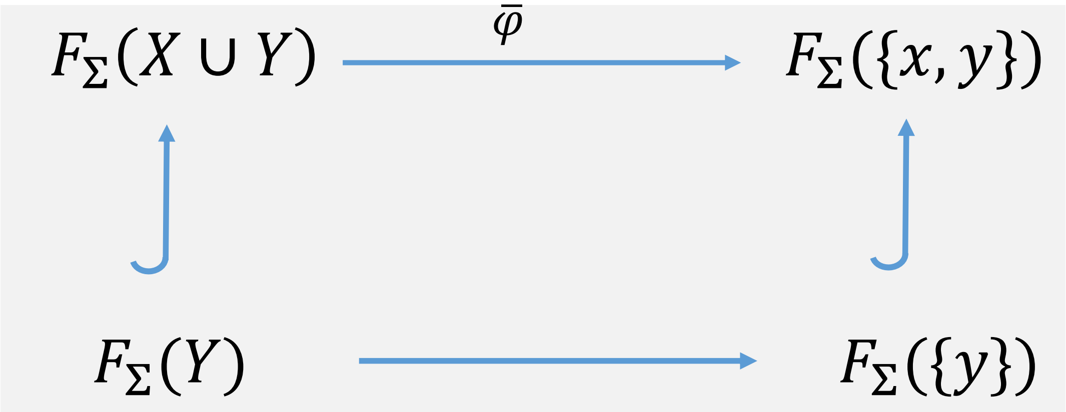
- <u>Thm</u>: $F_\Sigma(-)$ preserves preimages iff $\Sigma \vdash \Sigma'$

- Proof($\Leftarrow$) Enough to consider classifying preimages

  Given $p \in F_\Sigma(X \cup Y)$ with $\bar{\varphi}p \in F_\Sigma(\{y\})$ show $p \in F_\Sigma(Y)$

$$\bar{\varphi}p(x_1, \dots, x_n, y_1, \dots, y_m)$$
$$= p(x, \dots, x, y, \dots, y) \approx q(y)$$

$$p(x_1, x_2, \dots, x_n, y_1, \dots, y_m)$$
$$\approx p(z_1, x_2, \dots, x_n, y_1, \dots, y_m)$$
$$\approx p(z_1, z_2, \dots, x_n, y_1, \dots, y_m)$$
$$\approx$$
$$\approx$$



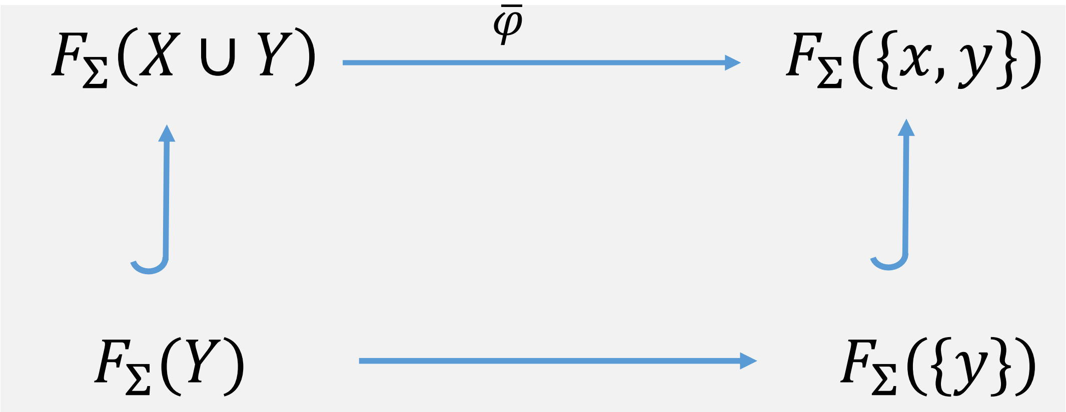
Preservation of preimages

- <u>Thm</u>: $F_\Sigma(-)$ preserves preimages iff $\Sigma \vdash \Sigma'$

- Proof($\Leftarrow$) Enough to consider classifying preimages

  Given $p \in F_\Sigma(X \cup Y)$ with $\bar{\varphi}p \in F_\Sigma(\{y\})$ show $p \in F_\Sigma(Y)$

$$\bar{\varphi}p(x_1, \dots, x_n, y_1, \dots, y_m)$$
$$= p(x, \dots, x, y, \dots, y) \approx q(y)$$



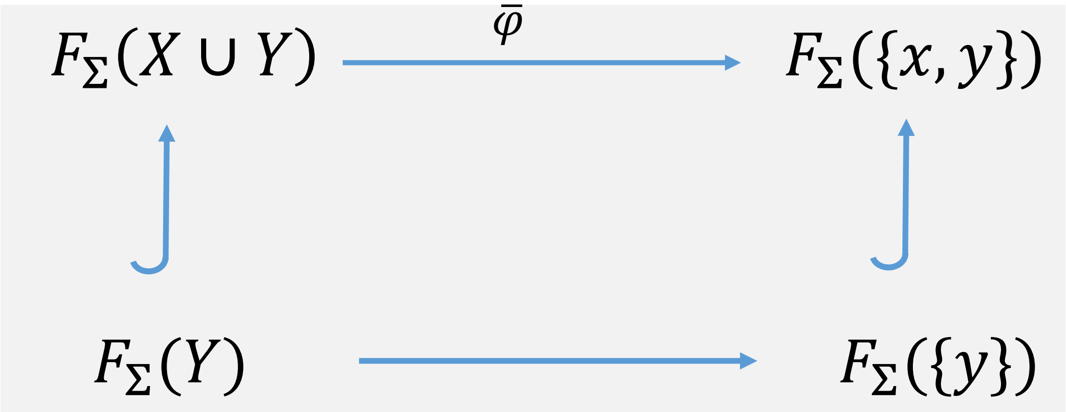
$$p(x_1, x_2, \dots, x_n, y_1, \dots, y_m)$$
$$\approx p(z_1, x_2, \dots, x_n, y_1, \dots, y_m)$$
$$\approx p(z_1, z_2, \dots, x_n, y_1, \dots, y_m)$$
$$\approx \;\cdots$$
$$\approx p(z_1, z_2, \dots, z_n, y_1, \dots, y_m)$$

Preservation of preimages

- Thm.: $F_\Sigma(-)$ preserves preimages iff $\Sigma \vdash \Sigma'$

- Proof(⇐) Enough to consider classifying preimages

  Given $p \in F_\Sigma(X \cup Y)$ with $\bar{\varphi}p \in F_\Sigma(\{y\})$ show $p \in F_\Sigma(Y)$



$$\bar{\varphi}p(x_1, \ldots, x_n, y_1, \ldots, y_m)$$
$$= p(x, \ldots, x, y, \ldots, y) \approx q(y)$$

$$p(x_1, x_2, \ldots, x_n, y_1, \ldots, y_m)$$
$$\approx p(z_1, x_2, \ldots, x_n, y_1, \ldots, y_m)$$
$$\approx p(z_1, z_2, \ldots, x_n, y_1, \ldots, y_m)$$
$$\approx \cdots$$
$$\approx p(z_1, z_2, \ldots, z_n, y_1, \ldots, y_m) \approx p(y, y, \ldots, y, y_1, \ldots, y_m)$$

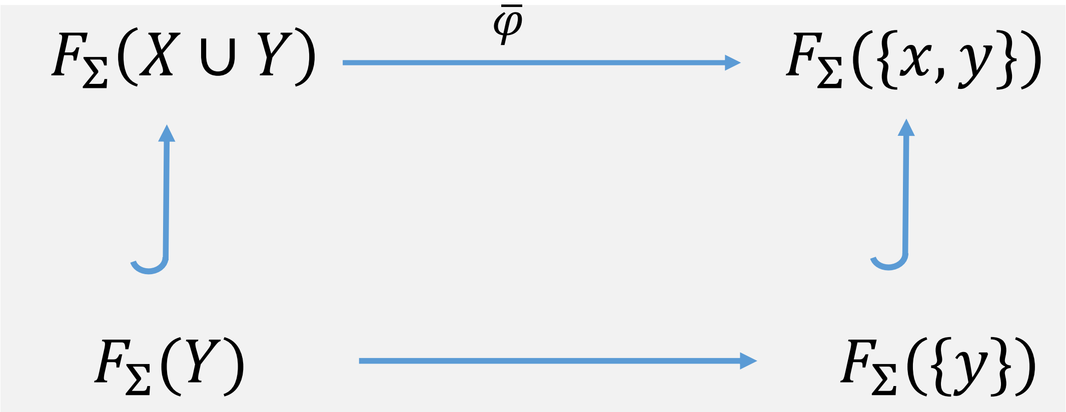
Preservation of preimages

- <u>Thm</u>: $F_\Sigma(-)$ preserves preimages iff $\Sigma \vdash \Sigma'$

- Proof($\Leftarrow$) Enough to consider classifying preimages

  Given $p \in F_\Sigma(X \cup Y)$ with $\bar{\varphi} p \in F_\Sigma(\{y\})$ show $p \in F_\Sigma(Y)$

$$\bar{\varphi}p(x_1, \dots, x_n, y_1, \dots, y_m) = p(x, \dots, x, y, \dots, y) \approx q(y)$$



$$\begin{aligned} p(x_1, x_2, \dots, x_n, y_1, \dots, y_m) &\approx p(z_1, x_2, \dots, x_n, y_1, \dots, y_m) \\ &\approx p(z_1, z_2, \dots, x_n, y_1, \dots, y_m) \\ &\approx \cdots \\ &\approx p(z_1, z_2, \dots, z_n, y_1, \dots, y_m) \approx p(y, y, \dots, y, y_1, \dots, y_m) \in F_\Sigma(Y) \checkmark \end{aligned}$$

Intro

State based systems

Functors and Coalgebras

Functor properties

Weak Pullback Preservation

Functors parameterized by algebras

Free-algebra functor

Preimage preservation

Weak kernel preservation

Conclusion

- and Breaking News

Mal‘cev term

- Variety $\mathcal{V}(\Sigma)$ = all algebras satisfying Σ

Mal‘cev term

А. И. Мальцев
1909-1967

- Variety $\mathcal{V}(\Sigma)$ = all algebras satisfying Σ

- *Mal‘cev variety* $:\Leftrightarrow \exists m.$

$$x = m(x, y, y)$$
$$m(x, x, y) = y$$

Mal‘cev term

А. И. Мальцев
1909-1967

- Variety $\mathcal{V}(\Sigma)$ = all algebras satisfying Σ

- *Mal‘cev variety* $:\Leftrightarrow \exists m.$

$$x = m(x,y,y)$$
$$m(x,x,y) = y$$

Groups:
$$m(x,y,z) = x * y^{-1} * z$$
Quasigroups:
$$m(x,y,z) = (x/(y\backslash y)) * (y\backslash z)$$
Rings:
$$m(x,y,z) = x - y + z$$
…

Mal‘cev term

А. И. Мальцев
1909-1967

- Variety $\mathcal{V}(\Sigma)$ = all algebras satisfying Σ

- *Mal‘cev variety* :⇔ $\exists m$.

$$x = m(x,y,y)$$
$$m(x,x,y) = y$$

- *n −permutable variety* :⇔ $\exists k. \exists m_1, \ldots, m_k$.

$$x = m_1(x,y,y),$$
$$m_i(x,x,y) = m_{i+1}(x,y,y)$$
$$m_k(x,x,y) = y$$

Groups:

$$m(x,y,z) = x * y^{-1} * z$$

Quasigroups:

$$m(x,y,z) = (x/(y\backslash y)) * (y\backslash z)$$

Rings:

$$m(x,y,z) = x - y + z$$

…

Mal‘cev term

А. И. Мальцев
1909-1967

- Variety $\mathcal{V}(\Sigma)$ = all algebras satisfying Σ

- *Mal‘cev variety* $:\Leftrightarrow \exists m.$

$$x = m(x, y, y)$$
$$m(x, x, y) = y$$

- n *−permutable variety* $:\Leftrightarrow \exists k. \exists m_1, \ldots, m_k.$

$$x = m_1(x, y, y),$$
$$m_i(x, x, y) = m_{i+1}(x, y, y)$$
$$m_k(x, x, y) = y$$

Groups:

$$m(x, y, z) = x * y^{-1} * z$$

Quasigroups:

$$m(x, y, z) = (x/(y\backslash y)) * (y\backslash z)$$

Rings:

$$m(x, y, z) = x - y + z$$

…

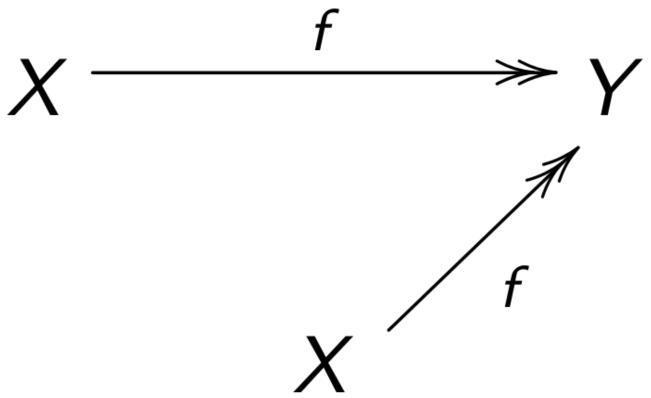
… the above, and also …

- *implication* algebras
- all *congruence regular* algebras

…

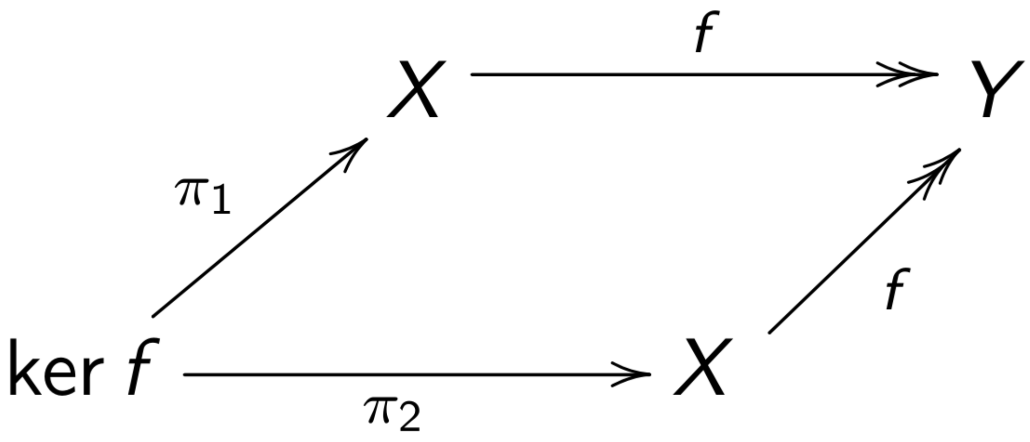
When does $F_\Sigma(-)$ preserve kernel pairs

- Thm: If $\mathcal{V}$ is Mal'cev, then $F_\Sigma(-)$ weakly preserves kernel pairs



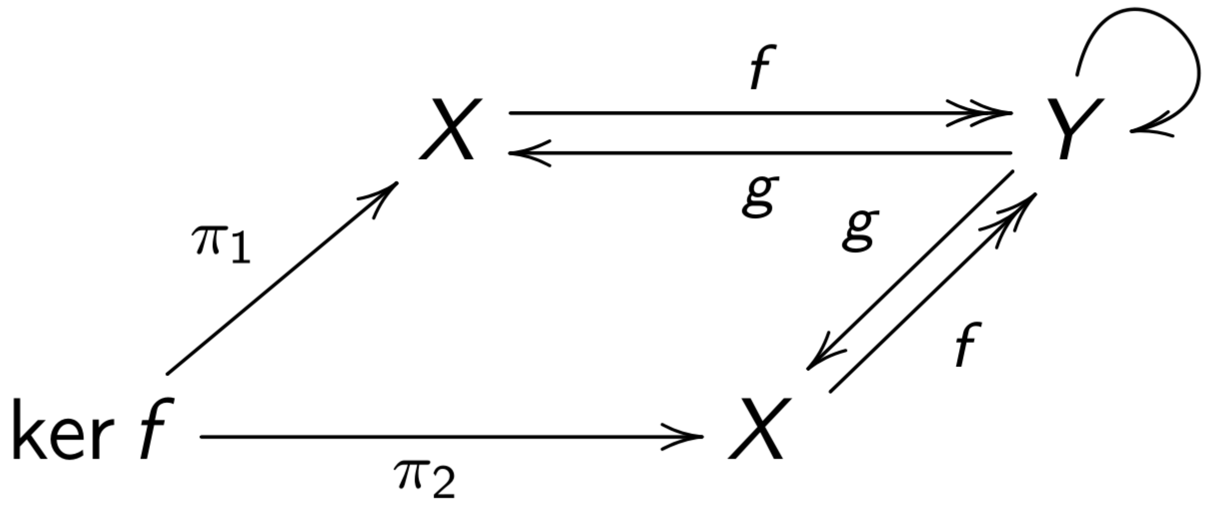
When does $F_\Sigma(-)$ preserve kernel pairs

- Thm: If $\mathcal{V}$ is Mal'cev, then $F_\Sigma(-)$ weakly preserves kernel pairs



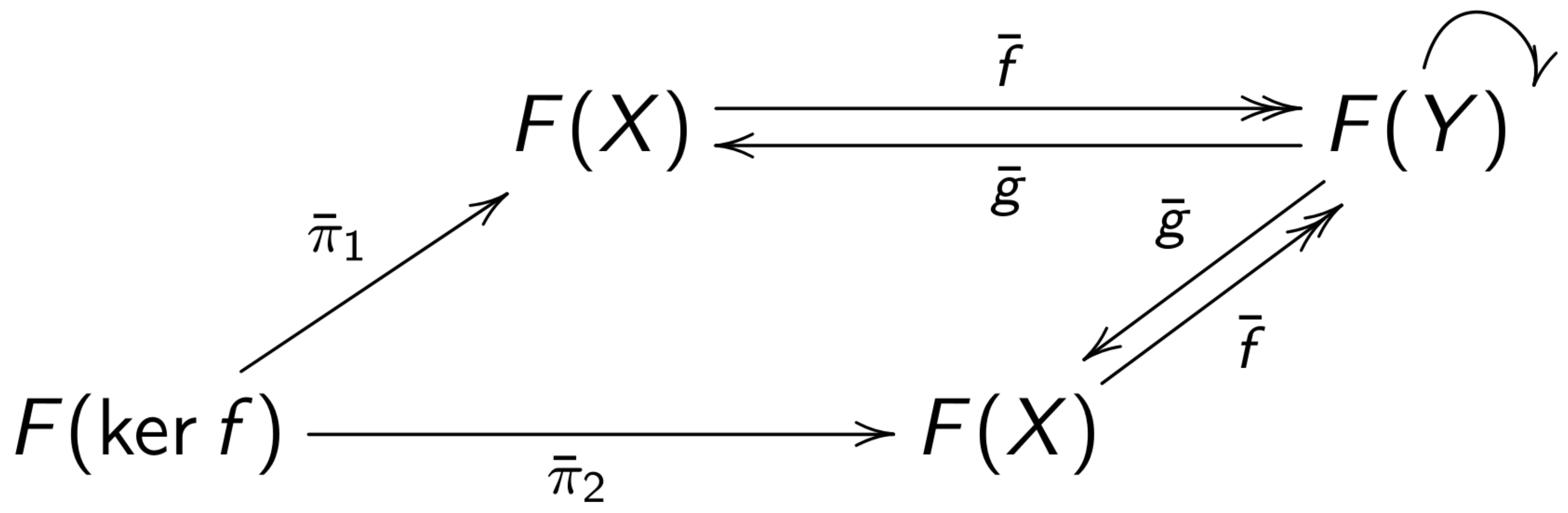
When does $F_\Sigma(-)$ preserve kernel pairs

- <u>Thm</u>: If $\mathcal{V}$ is Mal'cev, then $F_\Sigma(-)$ weakly preserves kernel pairs



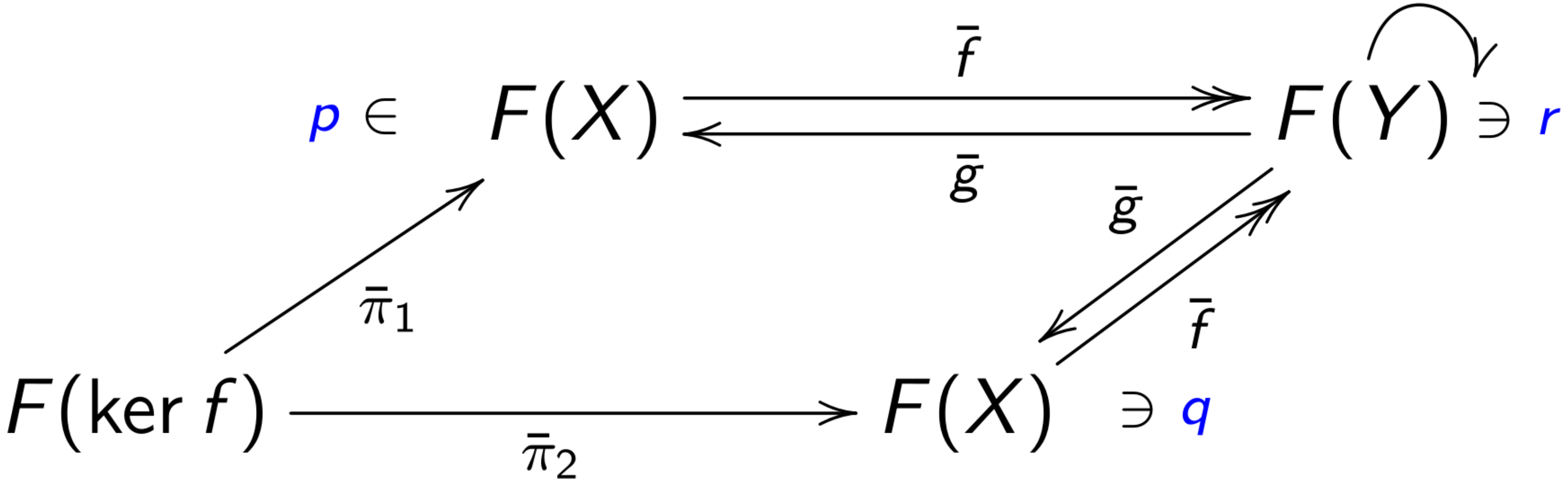
When does $F_\Sigma(-)$ preserve kernel pairs

- Thm: If $\mathcal{V}$ is Mal'cev, then $F_\Sigma(-)$ weakly preserves kernel pairs



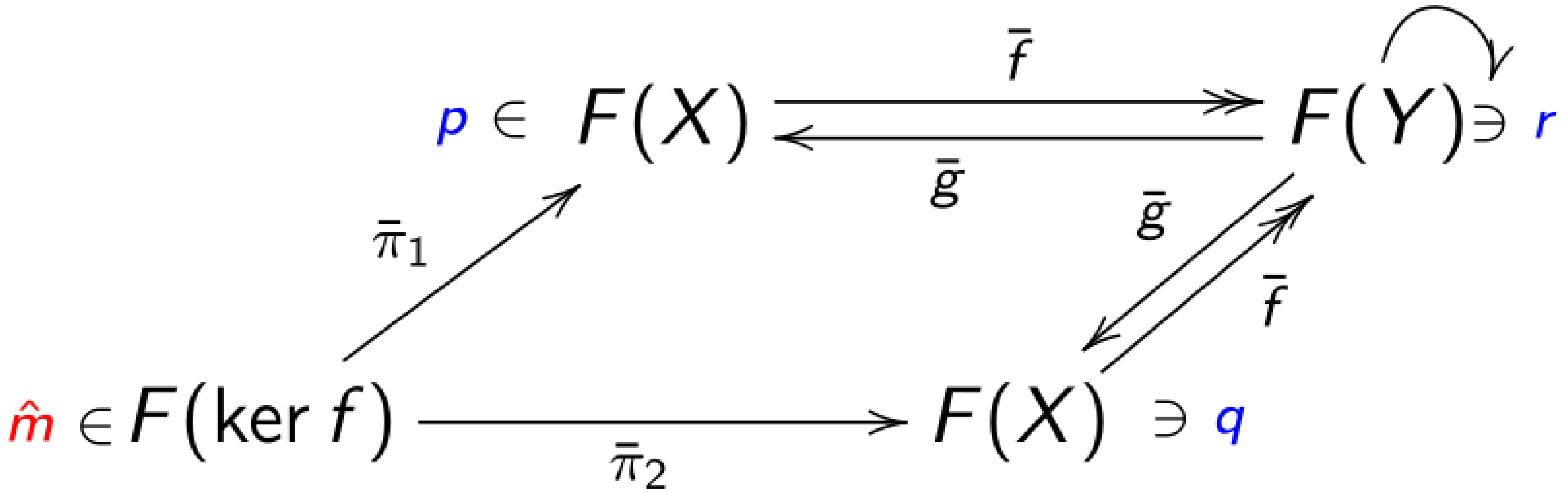
When does $F_\Sigma(-)$ preserve kernel pairs

- <u>Thm</u>: If $\mathcal{V}$ is Mal'cev, then $F_\Sigma(-)$ weakly preserves kernel pairs



When does $F_\Sigma(-)$ preserve kernel pairs

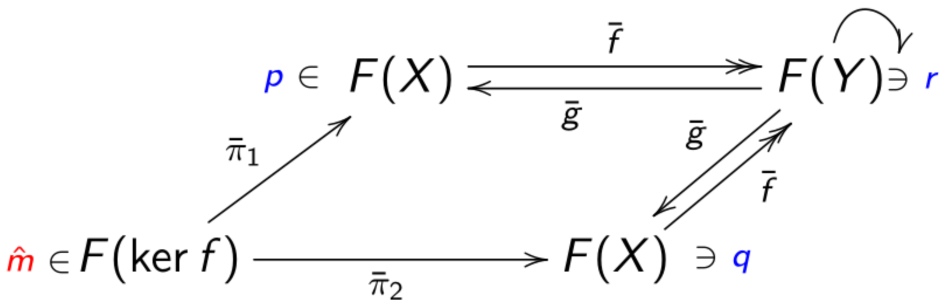
- Thm: If $\mathcal{V}$ is Mal'cev, then $F_\Sigma(-)$ weakly preserves kernel pairs



Mal‘cev ⇒ weak kernel preservation

- <u>Thm</u>: If $\mathcal{V}$ is Mal‘cev, then $F_\Sigma(-)$ weakly preserves kernel pairs

$\hat{m} := m(\hat{p}, \hat{r}, \hat{q})$ where
$\hat{p} := p(\langle x_1, gfx_1\rangle, \ldots, \langle x_n, gfx_n\rangle)$
$\hat{r} = r(\langle gy_1, gy_1\rangle, \ldots, \langle gy_k, gy_k\rangle)$,
$\hat{q} := q(\langle gfx'_1, x'_1\rangle, \ldots, \langle gfx'_m, x'_m\rangle)$

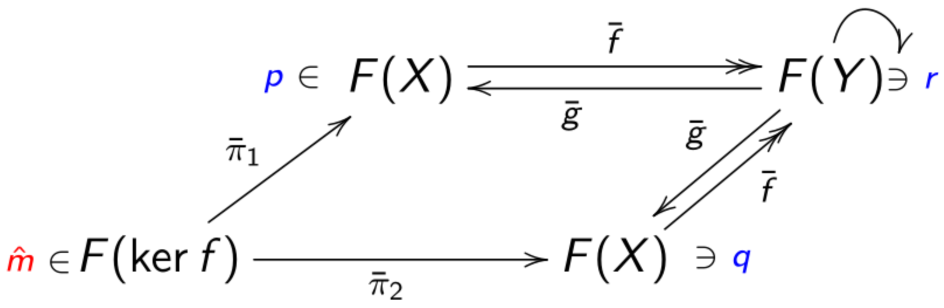


$$\bar{\pi}_1(\hat{m}) =$$
$$=$$
$$=$$
$$=$$
$$=$$
$$= \qquad = p$$

Mal'cev ⇒ weak kernel preservation

- <u>Thm</u>: If $\mathcal{V}$ is Mal'cev, then $F_\Sigma(-)$ weakly preserves kernel pairs

$\hat{m} := m(\hat{p}, \hat{r}, \hat{q})$ where
$\hat{p} := p(\langle x_1, gfx_1\rangle, \ldots, \langle x_n, gfx_n\rangle)$
$\hat{r} = r(\langle gy_1, gy_1\rangle, \ldots, \langle gy_k, gy_k\rangle)$,
$\hat{q} := q(\langle gfx'_1, x'_1\rangle, \ldots, \langle gfx'_m, x'_m\rangle)$

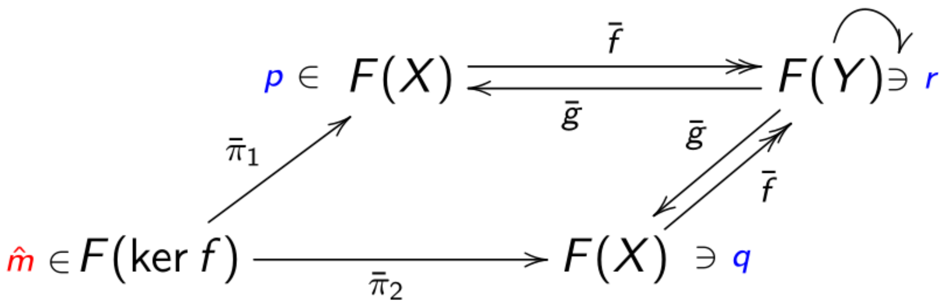


$$\bar{\pi}_1(\hat{m}) = \bar{\pi}_1(m(\hat{p}, \hat{r}, \hat{q}))$$
$=$
$=$
$=$
$=$
$= \qquad = p$

Mal'cev ⇒ weak kernel preservation

- Thm: If $\mathcal{V}$ is Mal'cev, then $F_\Sigma(-)$ weakly preserves kernel pairs

$\hat{m} := m(\hat{p}, \hat{r}, \hat{q})$ where
$\hat{p} := p(\langle x_1, gfx_1\rangle, \ldots, \langle x_n, gfx_n\rangle)$
$\hat{r} = r(\langle gy_1, gy_1\rangle, \ldots, \langle gy_k, gy_k\rangle)$,
$\hat{q} := q(\langle gfx'_1, x'_1\rangle, \ldots, \langle gfx'_m, x'_m\rangle)$

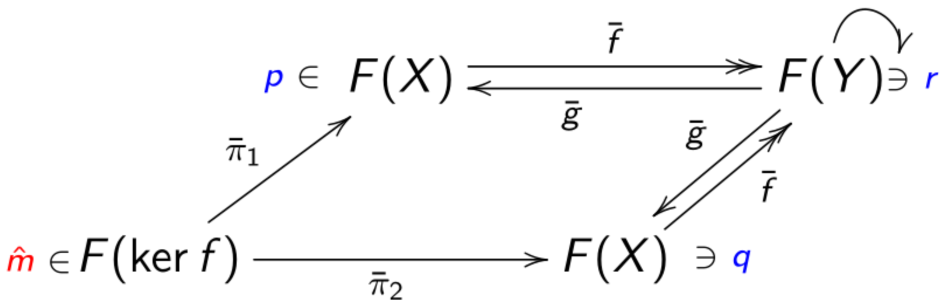


$$
\begin{aligned}
\bar{\pi}_1(\hat{m}) &= \bar{\pi}_1(m(\hat{p}, \hat{r}, \hat{q})) \\
&= m(\bar{\pi}_1\hat{p}, \bar{\pi}_1\hat{r}, \bar{\pi}_1\hat{q}) \\
&= \\
&= \\
&= \\
&= \qquad\qquad = p
\end{aligned}
$$

Mal'cev ⇒ weak kernel preservation

- <u>Thm</u>: If $\mathcal{V}$ is Mal'cev, then $F_\Sigma(-)$ weakly preserves kernel pairs

$\hat{m} := m(\hat{p}, \hat{r}, \hat{q})$ where
$\hat{p} := p(\langle x_1, gfx_1\rangle, \ldots, \langle x_n, gfx_n\rangle)$
$\hat{r} = r(\langle gy_1, gy_1\rangle, \ldots, \langle gy_k, gy_k\rangle),$
$\hat{q} := q(\langle gfx'_1, x'_1\rangle, \ldots, \langle gfx'_m, x'_m\rangle)$

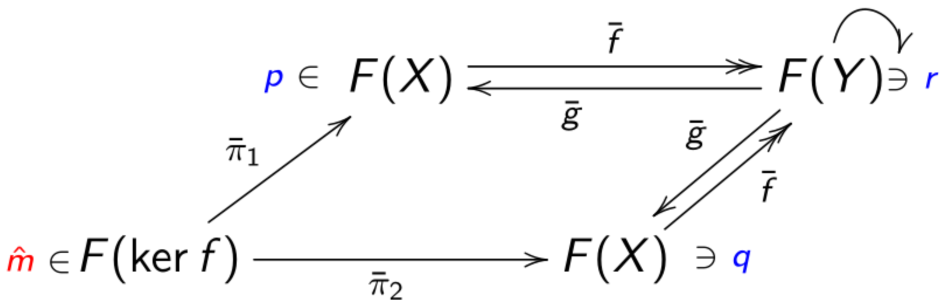


$$
\begin{aligned}
\bar{\pi}_1(\hat{m}) &= \bar{\pi}_1(m(\hat{p}, \hat{r}, \hat{q})) \\
&= m(\bar{\pi}_1\hat{p}, \bar{\pi}_1\hat{r}, \bar{\pi}_1\hat{q}) \\
&= m(p(x_1 \ldots, x_n), r(gy_1, \ldots, gy_k), q(gfx'_m, \ldots, gfx'_m)) \\
&= \\
&= \\
&= \qquad\qquad\qquad = p
\end{aligned}
$$

Mal‘cev ⇒ weak kernel preservation

- Thm: If $\mathcal{V}$ is Mal‘cev, then $F_\Sigma(-)$ weakly preserves kernel pairs

$\hat{m} := m(\hat{p}, \hat{r}, \hat{q})$ where
$\hat{p} := p(\langle x_1, gfx_1\rangle, \ldots, \langle x_n, gfx_n\rangle)$
$\hat{r} = r(\langle gy_1, gy_1\rangle, \ldots, \langle gy_k, gy_k\rangle)$,
$\hat{q} := q(\langle gfx'_1, x'_1\rangle, \ldots, \langle gfx'_m, x'_m\rangle)$

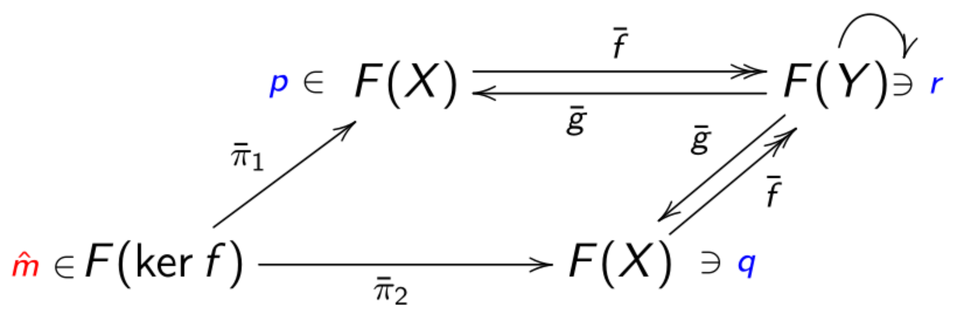


$$\begin{aligned}
\bar{\pi}_1(\hat{m}) &= \bar{\pi}_1(m(\hat{p}, \hat{r}, \hat{q})) \\
&= m(\bar{\pi}_1\hat{p}, \bar{\pi}_1\hat{r}, \bar{\pi}_1\hat{q}) \\
&= m(p(x_1 \ldots, x_n), r(gy_1, \ldots, gy_k), q(gfx'_m, \ldots, gfx'_m)) \\
&= m(p, \bar{g}r(y_1, \ldots, y_k), \bar{g}q(fx'_m, \ldots, fx'_m)) \\
&= \\
&= \qquad\qquad = p
\end{aligned}$$

Mal‘cev ⇒ weak kernel preservation

- Thm: If $\mathcal{V}$ is Mal‘cev, then $F_\Sigma(-)$ weakly preserves kernel pairs

$\hat{m} := m(\hat{p}, \hat{r}, \hat{q})$ where
$\hat{p} := p(\langle x_1, gfx_1\rangle, \ldots, \langle x_n, gfx_n\rangle)$
$\hat{r} = r(\langle gy_1, gy_1\rangle, \ldots, \langle gy_k, gy_k\rangle)$,
$\hat{q} := q(\langle gfx_1', x_1'\rangle, \ldots, \langle gfx_m', x_m'\rangle)$

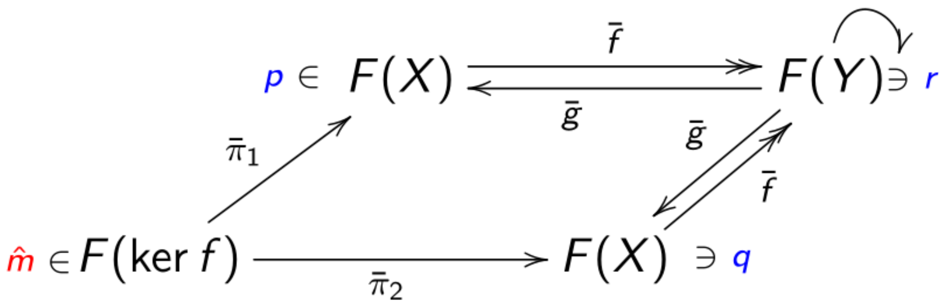


$$
\begin{aligned}
\bar\pi_1(\hat{m}) &= \bar\pi_1(m(\hat{p}, \hat{r}, \hat{q})) \\
&= m(\bar\pi_1\hat{p}, \bar\pi_1\hat{r}, \bar\pi_1\hat{q}) \\
&= m(p(x_1 \ldots, x_n), r(gy_1, \ldots, gy_k), q(gfx_m', \ldots, gfx_m')) \\
&= m(p, \bar{g}r(y_1, \ldots, y_k), \bar{g}q(fx_m', \ldots, fx_m')) \\
&= m(p, \bar{g}r, \bar{g}\bar{f}q(x_m', \ldots, x_m')) \\
&= \qquad\qquad = p
\end{aligned}
$$

Mal‘cev ⇒ weak kernel preservation

- <u>Thm</u>: If $\mathcal{V}$ is Mal‘cev, then $F_\Sigma(-)$ weakly preserves kernel pairs

$$
\begin{aligned}
\hat{m} &:= m(\hat{p}, \hat{r}, \hat{q}) \text{ where} \\
\hat{p} &:= p(\langle x_1, gfx_1\rangle, \ldots, \langle x_n, gfx_n\rangle) \\
\hat{r} &= r(\langle gy_1, gy_1\rangle, \ldots, \langle gy_k, gy_k\rangle), \\
\hat{q} &:= q(\langle gfx'_1, x'_1\rangle, \ldots, \langle gfx'_m, x'_m\rangle)
\end{aligned}
$$



$$
\begin{aligned}
\bar{\pi}_1(\hat{m}) &= \bar{\pi}_1(m(\hat{p}, \hat{r}, \hat{q})) \\
&= m(\bar{\pi}_1\hat{p}, \bar{\pi}_1\hat{r}, \bar{\pi}_1\hat{q}) \\
&= m(p(x_1 \ldots, x_n), r(gy_1, \ldots, gy_k), q(gfx'_m, \ldots, gfx'_m)) \\
&= m(p, \bar{g}r(y_1, \ldots, y_k), \bar{g}q(fx'_m, \ldots, fx'_m)) \\
&= m(p, \bar{g}r, \bar{g}\bar{f}q(x'_m, \ldots, x'_m)) \\
&= m(p, \bar{g}r, \bar{g}r) = p \qquad \checkmark
\end{aligned}
$$

n-permutable + wkp $\Leftrightarrow$ Mal‘cev

- Lemma: If F_Σ weakly preserves kernel pairs then for any terms p, q

$$p(x,x,y) = q(x,y,y) \Leftrightarrow \quad \exists\, s.$$

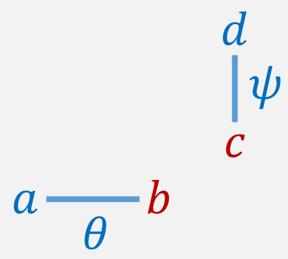
$$p(x,y,z) = s(x,y,z,z)$$
$$q(x,y,z) = s(x,x,y,z)$$

n-permutable + wkp $\Leftrightarrow$ Mal'cev

- <u>Lemma</u>: If F_Σ weakly preserves kernel pairs then for any terms p, q

$$p(x,x,y) = q(x,y,y) \Leftrightarrow \quad \exists\, s.$$

$$p(x,y,z) = s(x,y,z,z)$$
$$q(x,y,z) = s(x,x,y,z)$$



n-permutable + wkp $\Leftrightarrow$ Mal'cev

- <u>Lemma</u>: If F_Σ weakly preserves kernel pairs then for any terms p, q

$$p(x,x,y) = q(x,y,y) \Leftrightarrow \quad \exists\, s.$$

$$\begin{aligned} p(x,y,z) &= s(x,y,z,z) \\ q(x,y,z) &= s(x,x,y,z) \end{aligned}$$

d

ψ

c

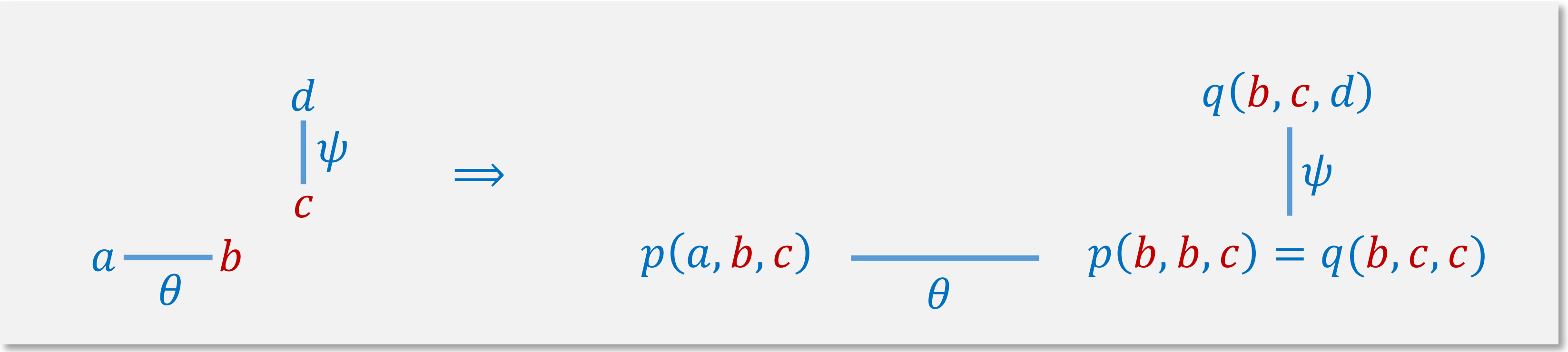
$a \overset{}{\underset{\theta}{\text{———}}} b \quad \Longrightarrow \quad p(a,b,c) \qquad q(b,c,d) \qquad p(b,b,c) = q(b,c,c)$

n-permutable + wkp ⇔ Mal'cev

- Lemma: If F_Σ weakly preserves kernel pairs then for any terms p, q

$$p(x,x,y) = q(x,y,y) \Leftrightarrow \quad \exists\, s.$$

$$p(x,y,z) = s(x,y,z,z)$$
$$q(x,y,z) = s(x,x,y,z)$$

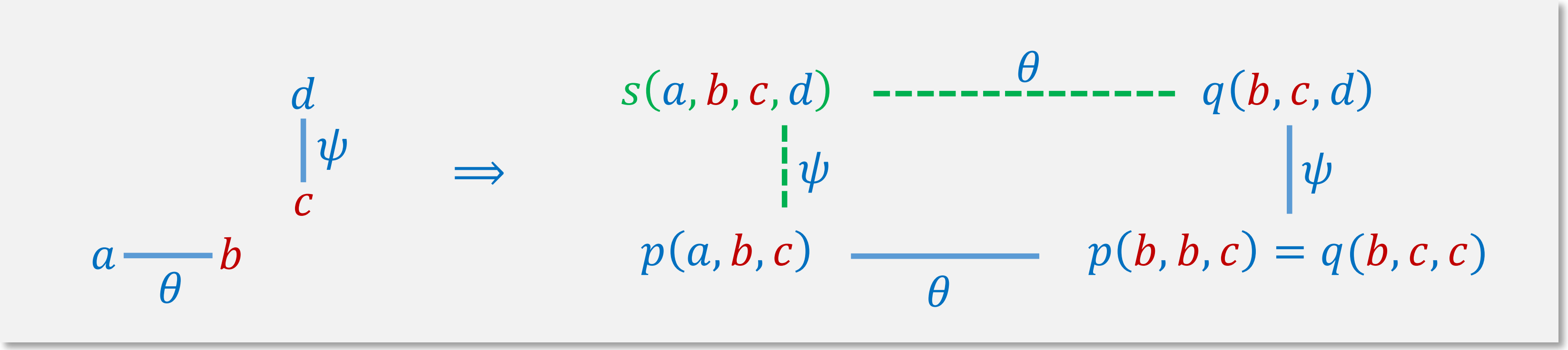


n-permutable + wkp ⇔ Mal'cev

- Lemma: If F_Σ weakly preserves kernel pairs then for any terms p, q

$$p(x,x,y) = q(x,y,y) \Leftrightarrow \quad \exists\, s.$$

$$p(x,y,z) = s(x,y,z,z)$$
$$q(x,y,z) = s(x,x,y,z)$$



n-permutable + wkp ⇔ Mal‘cev

- Lemma: If F_Σ weakly preserves kernel pairs then for any terms p, q

$$p(x,x,y) = q(x,y,y) \Leftrightarrow \quad \exists\, s.$$

$$p(x,y,z) = s(x,y,z,z)$$
$$q(x,y,z) = s(x,x,y,z)$$

- Thm: Mal‘cev ⇔ $n-permutable$ + F_Σ weakly preserves kernel pairs

$x = m(x,y,y)$ $m(x,x,y) = y$	⇔	*WKP*	+	$x = m_1(x,y,y),$ $m_i(x,x,y) = m_{i+1}(x,y,y)$ $m_k(x,x,y) = y$

n-permutable + wkp $\Leftrightarrow$ Mal'cev

- Lemma: If F_Σ weakly preserves kernel pairs then for any terms p, q

$$p(x,x,y) = q(x,y,y) \Leftrightarrow \quad \exists\, s.$$

$$p(x,y,z) = s(x,y,z,z)$$
$$q(x,y,z) = s(x,x,y,z)$$

- Thm: Mal'cev $\Leftrightarrow$ n $-permutable$ + F_Σ weakly preserves kernel pairs

Proof $\Leftarrow$:

$$x = m_1(x,y,y),$$
$$\dots$$
$$m_i\,(x,x,y) \quad = m_{i+1}(x,y,y)$$
$$\dots$$
$$m_k(x,x,y) = y$$

n-permutable + wkp $\Leftrightarrow$ Mal'cev

- Lemma: If F_Σ weakly preserves kernel pairs then for any terms p, q

$$p(x,x,y) = q(x,y,y) \Leftrightarrow \quad \exists\, s.$$

$$p(x,y,z) = s(x,y,z,z)$$
$$q(x,y,z) = s(x,x,y,z)$$

- Thm: Mal'cev $\Leftrightarrow$ n *−permutable* + F_Σ weakly preserves kernel pairs

Proof $\Leftarrow$:

$$x = m_1(x,y,y),$$
$$\dots$$
$$m_{i-1}(x,x,y) = m_i(x,y,y)$$
$$m_i(x,x,y) = m_{i+1}(x,y,y)$$
$$m_{i+1}(x,x,y) = m_{i+2}(x,y,y)$$
$$\dots$$
$$m_k(x,x,y) = y$$

n-permutable + wkp ⇔ Mal'cev

- Lemma: If F_Σ weakly preserves kernel pairs then for any terms p, q

$$p(x,x,y) = q(x,y,y) \Leftrightarrow \quad \exists\, s.$$

$$p(x,y,z) = s(x,y,z,z)$$
$$q(x,y,z) = s(x,x,y,z)$$

- Thm: Mal'cev ⇔ n −*permutable* + F_Σ weakly preserves kernel pairs

Proof ⇐ :

$$x = m_1(x,y,y),$$
$$\cdots$$
$$m_{i-1}(x,x,y) = m_i(x,y,y)$$
$$m_i(x,x,y) = m_{i+1}(x,y,y)$$
$$m_{i+1}(x,x,y) = m_{i+2}(x,y,y)$$
$$\cdots$$
$$m_k(x,x,y) = y$$

n-permutable + wkp $\Leftrightarrow$ Mal‘cev

- Lemma: If F_Σ weakly preserves kernel pairs then for any terms p, q

$$p(x,x,y) = q(x,y,y) \Leftrightarrow \quad \exists\, s.$$

$$p(x,y,z) = s(x,y,z,z)$$
$$q(x,y,z) = s(x,x,y,z)$$

- Thm: Mal‘cev $\Leftrightarrow$ n *–permutable* + F_Σ weakly preserves kernel pairs

Proof $\Leftarrow$:

$$x = m_1(x,y,y),$$
$$\cdots$$
$$m_{i-1}(x,x,y) = m_i(x,y,y)$$
$$m_i(x,x,y) = m_{i+1}(x,y,y)$$
$$m_{i+1}(x,x,y) = m_{i+2}(x,y,y)$$
$$\cdots$$
$$m_k(x,x,y) = y$$

$$m_i(x,y,z) = s(x,y,z,z)$$
$$m_{i+1}(x,y,z) = s(x,x,y,z)$$

n-permutable + wkp ⇔ Mal‘cev

- Lemma: If F_Σ weakly preserves kernel pairs then for any terms p, q

$$p(x,x,y) = q(x,y,y) \Leftrightarrow \quad \exists\, s.$$

$$p(x,y,z) = s(x,y,z,z)$$
$$q(x,y,z) = s(x,x,y,z)$$

- Thm: Mal‘cev ⇔ n −*permutable* + F_Σ weakly preserves kernel pairs

Proof ⇐ :

$$x = m_1(x,y,y),$$
$$\dots$$
$$m_{i-1}(x,x,y) = m_i(x,y,y)$$
$$m_i(x,x,y) = m_{i+1}(x,y,y)$$
$$m_{i+1}(x,x,y) = m_{i+2}(x,y,y)$$
$$\dots$$
$$m_k(x,x,y) = y$$

$$m_i(x,y,z) = s(x,y,z,z)$$
$$m_{i+1}(x,y,z) = s(x,x,y,z)$$

$$m(x,y,z) := s(x,y,y,z)$$

n-permutable + wkp ⇔ Mal'cev

- Lemma: If F_Σ weakly preserves kernel pairs then for any terms p, q

$$p(x,x,y) = q(x,y,y) \Leftrightarrow \quad \exists\, s.$$

$$p(x,y,z) = s(x,y,z,z)$$
$$q(x,y,z) = s(x,x,y,z)$$

- Thm: Mal'cev ⇔ n *−permutable* + F_Σ weakly preserves kernel pairs

Proof ⇐ :

$$x = m_1(x,y,y),$$
$$\dots$$
$$m_{i-1}(x,x,y) = m_i(x,y,y)$$
$$m_i(x,x,y) = m_{i+1}(x,y,y)$$
$$m_{i+1}(x,x,y) = m_{i+2}(x,y,y)$$
$$\dots$$
$$m_k(x,x,y) = y$$

$$m_i(x,y,z) = s(x,y,z,z)$$
$$m_{i+1}(x,y,z) = s(x,x,y,z)$$

$$m(x,y,z) := s(x,y,y,z)$$

$$m(x,y,y) = s(x,y,y,y) = m_i(x,y,y)$$

n-permutable + wkp ⇔ Mal'cev

- Lemma: If F_Σ weakly preserves kernel pairs then for any terms p, q

$$p(x,x,y) = q(x,y,y) \Leftrightarrow \quad \exists\, s.$$

$$p(x,y,z) = s(x,y,z,z)$$
$$q(x,y,z) = s(x,x,y,z)$$

- Thm: Mal'cev ⇔ n *−permutable* + F_Σ weakly preserves kernel pairs

Proof ⇐ :

$$x = m_1(x,y,y),$$
$$\dots$$
$$m_{i-1}(x,x,y) = m_i(x,y,y) = m(x,y,y)$$
$$m_i(x,x,y) = m_{i+1}(x,y,y)$$
$$m_{i+1}(x,x,y) = m_{i+2}(x,y,y)$$
$$\dots$$
$$m_k(x,x,y) = y$$

$$m_i(x,y,z) = s(x,y,z,z)$$
$$m_{i+1}(x,y,z) = s(x,x,y,z)$$

$$m(x,y,z) := s(x,y,y,z)$$

n-permutable + wkp ⇔ Mal‘cev

- Lemma: If F_Σ weakly preserves kernel pairs then for any terms p, q

$$p(x,x,y) = q(x,y,y) \Leftrightarrow \quad \exists\, s.$$

$$\begin{aligned} p(x,y,z) &= s(x,y,z,z) \\ q(x,y,z) &= s(x,x,y,z) \end{aligned}$$

- Thm: Mal‘cev ⇔ n *−permutable* + F_Σ weakly preserves kernel pairs

Proof ⇐ :

$$x = m_1(x,y,y),$$
$$\dots$$
$$m_{i-1}(x,x,y) = m_i(x,y,y) = m(x,y,y)$$
$$m_i(x,x,y) = m_{i+1}(x,y,y)$$
$$m_{i+1}(x,x,y) = m_{i+2}(x,y,y)$$
$$\dots$$
$$m_k(x,x,y) = y$$

$$\begin{aligned} m_i(x,y,z) &= s(x,y,z,z) \\ m_{i+1}(x,y,z) &= s(x,x,y,z) \end{aligned}$$

$$m(x,y,z) := s(x,y,y,z)$$

$$\begin{aligned} m(x,x,y) &= s(x,x,x,y) \\ &= m_{i+1}(x,x,y) \end{aligned}$$

n-permutable + wkp ⇔ Mal‘cev

- Lemma: If F_Σ weakly preserves kernel pairs then for any terms p, q

$$p(x,x,y) = q(x,y,y) \Leftrightarrow \quad \exists\, s.$$

$$\begin{aligned} p(x,y,z) &= s(x,y,z,z) \\ q(x,y,z) &= s(x,x,y,z) \end{aligned}$$

- Thm: Mal‘cev ⇔ n *–permutable* + F_Σ weakly preserves kernel pairs

Proof $\Leftarrow$:

$$x = m_1(x,y,y),$$

$$\dots$$

$$m_{i-1}(x,x,y) = m_i(x,y,y) = m(x,y,y)$$

$$m_i\,(x,x,y) = m_{i+1}(x,y,y)$$

$$m(x,y,y) = m_{i+1}(x,x,y) = m_{i+2}(x,y,y)$$

$$\dots$$

$$m_k(x,x,y) = y$$

$$\begin{aligned} m_i\,(x,y,z) &= s(x,y,z,z) \\ m_{i+1}(x,y,z) &= s(x,x,y,z) \end{aligned}$$

$$m(x,y,z) \quad := s(x,y,y,z)$$

$$\begin{aligned} m(x,x,y) &= s(x,x,x,y) \\ &= m_{i+1}(x,x,y) \end{aligned}$$

n-permutable + wkp ⇔ Mal'cev

- Lemma: If F_Σ weakly preserves kernel pairs then for any terms p, q

$$p(x,x,y) = q(x,y,y) \Leftrightarrow \quad \exists\, s.$$

$$p(x,y,z) = s(x,y,z,z)$$
$$q(x,y,z) = s(x,x,y,z)$$

- Thm: Mal'cev ⇔ n *−permutable* + F_Σ weakly preserves kernel pairs

Proof ⇐ :

$$x = m_1(x,y,y),$$
$$\dots$$
$$m_{i-1}(x,x,y) = m_i(x,y,y) = m(x,y,y)$$
$$m_i(x,x,y) = m_{i+1}(x,y,y)$$
$$m(x,y,y) = m_{i+1}(x,x,y) = m_{i+2}(x,y,y)$$
$$\dots$$
$$m_k(x,x,y) = y$$

$$m_i(x,y,z) = s(x,y,z,z)$$
$$m_{i+1}(x,y,z) = s(x,x,y,z)$$

$$m(x,y,z) := s(x,y,y,z)$$

n-permutable + wkp ⇔ Mal'cev

- Lemma: If F_Σ weakly preserves kernel pairs then for any terms p, q

$$p(x,x,y) = q(x,y,y) \Leftrightarrow \quad \exists\, s.$$

$$p(x,y,z) = s(x,y,z,z)$$
$$q(x,y,z) = s(x,x,y,z)$$

- Thm: Mal'cev ⇔ n −*permutable* + F_Σ weakly preserves kernel pairs

Proof ⇐ :

$$x = m_1(x,y,y),$$
$$\ldots$$
$$m_{i-1}(x,x,y) \;= m(x,y,y)$$

$$m(x,x,y) = m_{i+2}(x,y,y)$$
$$\ldots$$
$$m_k(x,x,y) = y$$

$$m_i\;(x,y,z) \;= s(x,y,z,z)$$
$$m_{i+1}(x,y,z) = s(x,x,y,z)$$

$$m(x,y,z) \;:= s(x,y,y,z)$$

Conclusion

- F_{Σ} preserves
  - preimages $\Leftrightarrow$ weak independence implies independence
  - kernel pairs $\Leftarrow \mathcal{V}(\Sigma)$ is Mal‘cev
    $\Rightarrow$ (n-permutable $\Rightarrow$ Mal‘cev)

  Open: $\Leftrightarrow$???

Distributive and modular varieties

- If $F_{\mathcal{V}}(X)$ preserves kernel pairs then
  - $\mathcal{V}$ is congruence distributive iff
    $\exists k \in \mathbb{N}. \exists m. m$ is k–ary majority term, i.e.
    $$m(x, \dots, x, y) = \cdots = m(x, \dots, x, y, x, \dots, x) = \cdots = m(y, x, \dots, x) = x$$
  - $\mathcal{V}$ is congruence modular iff
    $\exists k \in \mathbb{N}. \exists m. \exists q.$
    $$m(x, \dots, x, y) = \cdots = m(x, \dots, x, y, x, \dots, x) = \cdots = m(x, y, x, \dots, x) = x$$
    $$m(y, x, \dots, x) = q(x, x, y)$$
    $$q(x, y, y) = x$$