Università degli Studi di Pisa
Corso di Laurea Magistrale in Informatica

Anno Accademico 2021-2022

Insegnamento di

Foundation of Computing

Pagina del corso: `http:/pages.di.unipi.it/montanari/FOC.html`

Note di

Logic Programming

a cura di
Ugo Montanari
Dipartimento di Informatica
Università degli Studi di Pisa
ugo@di.unipi.it

A coalgebraic approach to unification semantics of logic programming

Ugo Montanari (UNIPI)

Giorgio Mossa (UNIPI)

Roberto Bruni (UNIPI)

IFIP WG 1.3, Massa Marittima, Jan.14-16, 2020

Object

Can we give a categorical account of the operational semantics of logic programs?

Inspiration

S-semantics

Non-ground atoms
Answer substitutions

S-semantics by Falaschi, Levi, Palamidessi and Martelli

Operational semantics

Logic program

logic signature:
Σ operation symbols
Π predicate symbols

atoms: $A \equiv P(t_1, ..., t_n)$
goals: $G \equiv A_1, ..., A_k$

$\mathbb{P}$ a set of Horn clauses $H : -B$

Operational semantics

$$\mathbb{P} \models G \Rightarrow_{\sigma} F$$

Operational semantics

$$\frac{H \mathrel{:\!\!-} B \in \mathbb{P} \quad \sigma = \mathrm{mgu}(A, \rho(H))}{\mathbb{P} \models A \Rightarrow_{\sigma} \sigma(\rho(B))}$$

$\rho(H)$ and $\rho(B)$ have no variable in common with A

Operational semantics

$$\frac{\mathbb{P} \models G \Rightarrow_\sigma F}{\mathbb{P} \models G, G' \Rightarrow_\sigma F, \sigma(G')}$$

Canonical predicate

$\{x_1, ..., x_n\}$ canonical set of variables

$P(t_1, ..., t_n)$ as $\sigma_t(P(x_1, ..., x_n))$

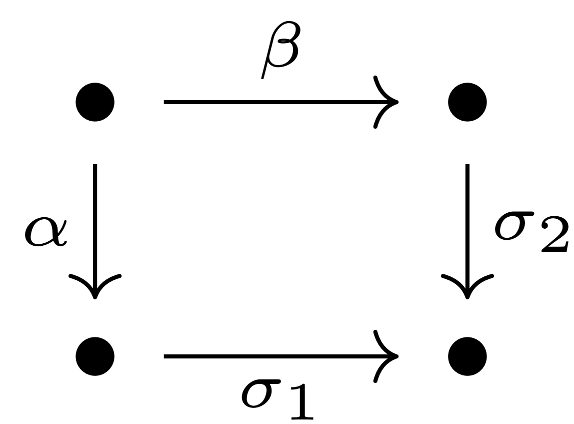
$\sigma_t : \{x_1, ..., x_n\} \rightarrow T_\Sigma(X)$ with $\sigma_t(x_i) = t_i$

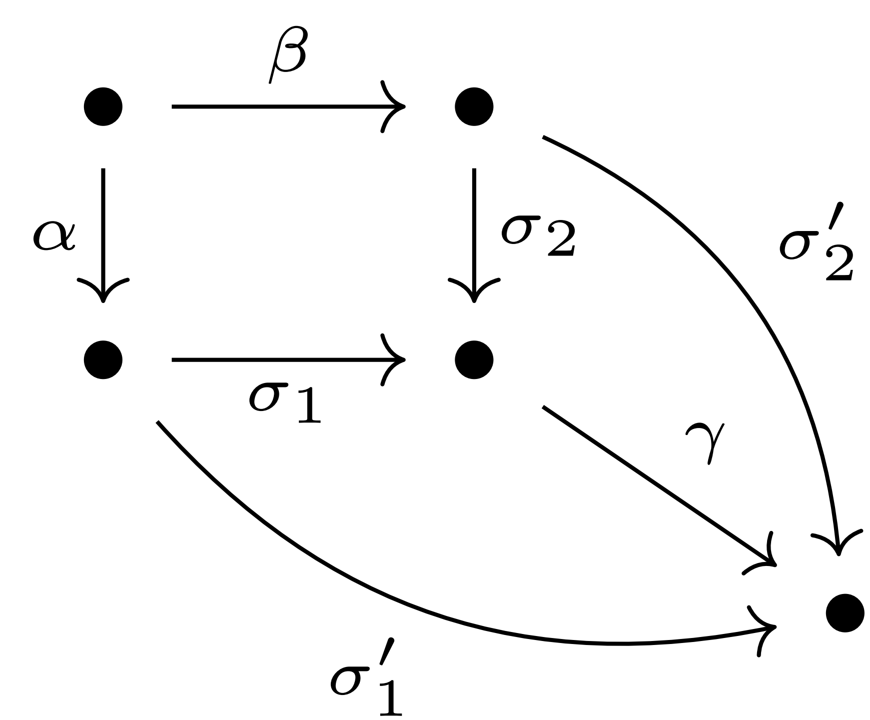
we write P for $P(x_1, ..., x_n)$

Unification as a pushout

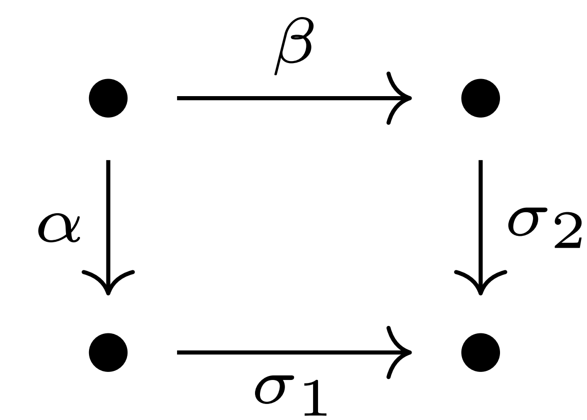
$$\mathrm{mgu}(\alpha(P), \beta(Q))?$$

$$P = Q$$





Operational semantics



$$\frac{\alpha(P) :\!\text{-}\, B \in \mathbb{P} \quad (\alpha, \beta, \sigma_1, \sigma_2) \text{ is a pushout}}{\mathbb{P} \models \beta(P) \Rightarrow_{\sigma_2} \sigma_1(B)}$$

no need of creating new variables

Coalgebras

Coalgebras

$$F : \mathbf{C} \to \mathbf{C}$$

$$\langle X, \alpha : X \to F(X) \rangle$$

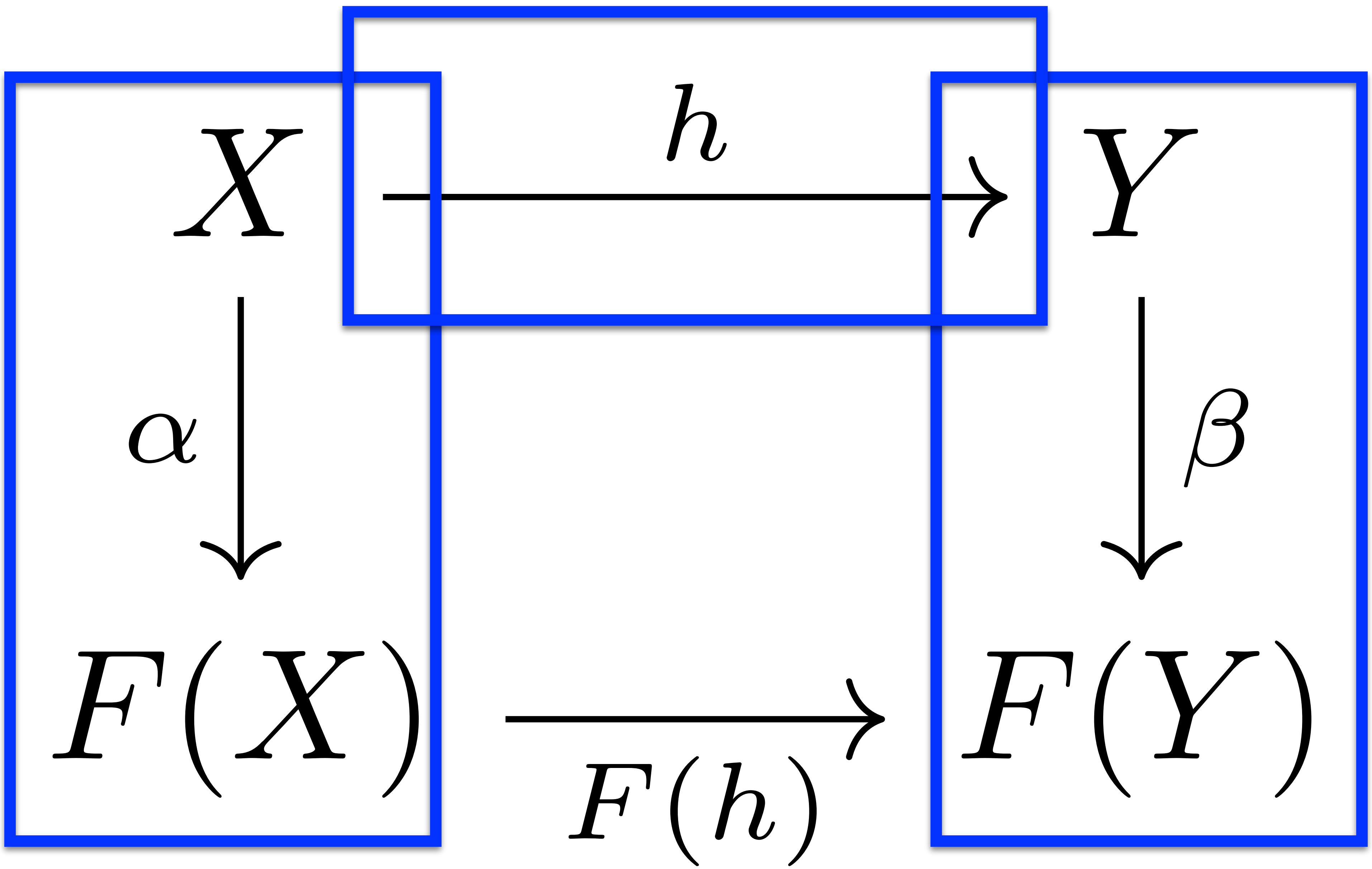
$$\wp_L : \mathbf{Set} \to \mathbf{Set}$$

$$\wp_L(X) = \wp(L \times X)$$

$$\alpha : S \to \wp(L \times S)$$

$$\alpha(s) = \{(\ell, t) \mid s \xrightarrow{\ell} t\}$$

Coalg(F)



Final coalgebra

$$\langle T, \tau \rangle \qquad \forall \langle X, \alpha \rangle .\ \exists! t : \langle X, \alpha \rangle \to \langle T, \tau \rangle$$

$$p \sim q \Leftrightarrow t(p) = t(q)$$

Algebraic structure of states

The theory of monoids

a sort M

$\cdot : M \times M \to M$

$\square :\to M$

$$x \cdot (y \cdot z) = (x \cdot y) \cdot z$$

$$x \cdot \square = x = \square \cdot x$$

The theory of substitutions

a countable set of sorts $\{\underline{n} \mid n \in \mathbb{N}\}$
$\underline{\sigma} : \underline{n} \to \underline{m}$ for every $\sigma : \{x_1, ..., x_n\} \to T_\Sigma(\{x_1, ..., x_m\})$

$$\underline{\tau}(\underline{\sigma}(x)) = \underline{\tau \circ \sigma}(x)$$

$$\underline{\mathrm{id}_n}(x) = x$$

The theory of substitutive monoids

Just a product theory!

a countable set of sorts $\{\underline{n} \mid n \in \mathbb{N}\}$

$\underline{\sigma} : \underline{n} \to \underline{m}$ for every $\sigma : \{x_1, ..., x_n\} \to T_\Sigma(\{x_1, ..., x_m\})$

$\cdot_{\underline{n}} : \underline{n} \times \underline{n} \to \underline{n}$

$\square_{\underline{n}} : \to \underline{n}$

$$\underline{\tau}(\underline{\sigma}(x)) = \underline{\tau \circ \sigma}(x) \qquad \underline{\mathrm{id}_n}(x) = x$$

$$x \cdot_{\underline{n}} (y \cdot_{\underline{n}} z) = (x \cdot_{\underline{n}} y) \cdot_{\underline{n}} z \qquad x \cdot_{\underline{n}} \square_{\underline{n}} = \square_{\underline{n}} \cdot_{\underline{n}} x = x$$

$$\underline{\sigma}(x \cdot_{\underline{n}} y) = \underline{\sigma}(x) \cdot_{\underline{m}} \underline{\sigma}(y) \qquad \underline{\sigma}(\square_{\underline{n}}) = \square_{\underline{m}}$$

The theory of Π-substitutive monoids

Add to the theory of substitutive monoids

$$P :\to \underline{n} \text{ for every } P \in \Pi_n$$

Initiality of the goal algebra

$$A_1, ..., A_k$$

$$\underline{\sigma_1}(P_1) \cdot_{\underline{n}} (\cdots \cdot_{\underline{n}} (\underline{\sigma_{k-1}}(P_{k-1}) \cdot_{\underline{n}} \underline{\sigma_k}(P_k)) \ldots)$$

A syntax for states

Coalgebraic semantics

De Simone rules

$$\frac{\sigma(P) :- B \in \mathbb{P} \quad \gamma \text{ iso}}{P \xrightarrow{\gamma \circ \sigma} \underline{\gamma}(B)}$$

$$\frac{G \xrightarrow{\sigma} B \quad (\tau, \sigma, \sigma', \tau') \text{ is a pushout}}{\underline{\tau}(G) \xrightarrow{\sigma'} \underline{\tau'}(B)}$$

$$\frac{G \xrightarrow{\sigma} B}{G \cdot G' \xrightarrow{\sigma} B \cdot \underline{\sigma}(G')} \qquad \frac{G \xrightarrow{\sigma} B}{G' \cdot G \xrightarrow{\sigma} \underline{\sigma}(G') \cdot B}$$

Induced coalgebra

(terms of Π-substitutive monoids)

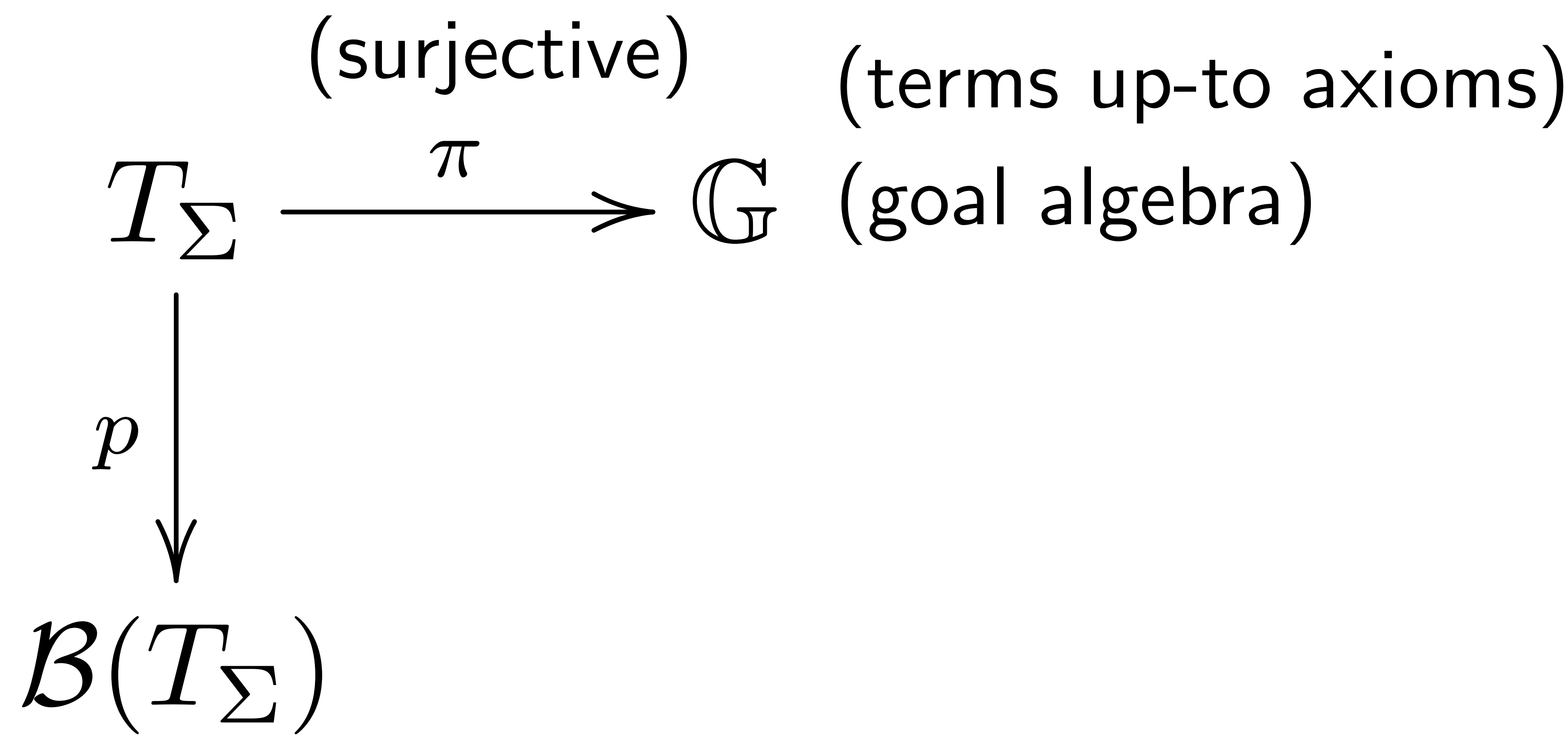
(no axioms)

$$\begin{array}{c} T_\Sigma \\ \Big\downarrow p \\ \mathcal{B}(T_\Sigma) \end{array}$$

$$G \xrightarrow{\sigma} G'$$
$$\Leftrightarrow$$
$$\mathbb{P} \models G \Rightarrow_\sigma G'$$

$$p(t) = \Big\{(\sigma, t') \colon t \xrightarrow{\sigma} t' \text{ is a derivable sequent}\Big\}$$

Homomorphism



Lifting

$$
\begin{array}{ccc}
T_\Sigma & \xrightarrow{\ \pi\ } & \mathbb{G} \\
\Big\downarrow{\scriptstyle p} & & \\
\mathcal{B}(T_\Sigma) & \xrightarrow[\mathcal{B}(\pi)]{} & \mathcal{B}(\mathbb{G})
\end{array}
$$

Structured coalgebra

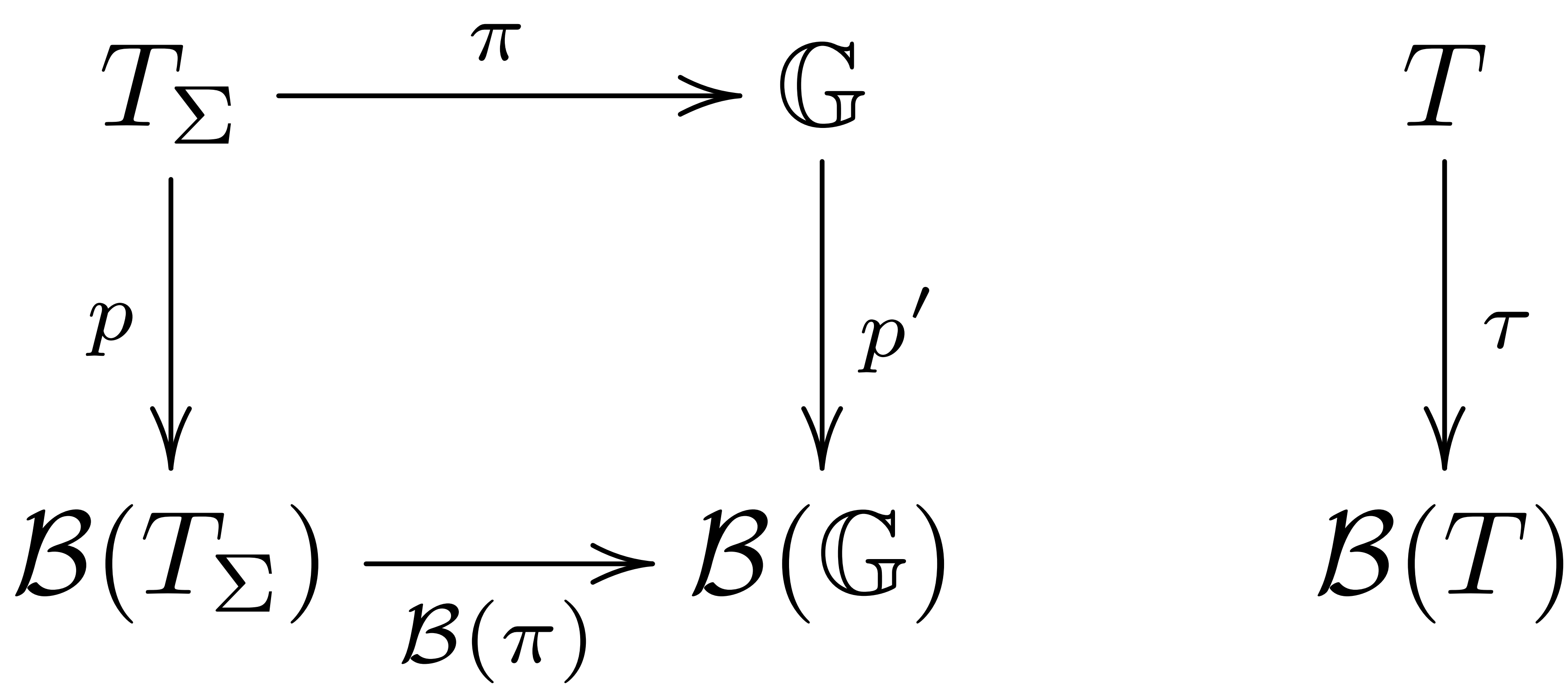
$$
\begin{array}{ccc}
T_\Sigma & \xrightarrow{\quad \pi \quad} & \mathbb{G} \\
{\scriptstyle p}\big\downarrow & & \big\downarrow{\scriptstyle p'} \\
\mathcal{B}(T_\Sigma) & \xrightarrow[\mathcal{B}(\pi)]{\qquad} & \mathcal{B}(\mathbb{G})
\end{array}
$$

(axioms bisimulate)

implies

(unique coalgebra)

Finality



$\mathbf{Coalg}(\mathcal{B})$ has a final object

Examples

Example

$$P(x,y,z) :\text{-}\ Q(x,y), R(y,z)$$
$$Q(a,c) :\text{-}\ \square$$
$$R(c,b) :\text{-}\ \square$$

$$S(x,y,z) :\text{-}\ T(x,y,z)$$
$$T(a,c,z) :\text{-}\ V(z)$$
$$T(x,c,b) :\text{-}\ U(x)$$

$$V(b) :\text{-}\ \square$$
$$U(a) :\text{-}\ \square$$

$$P \sim S$$

isomorphic LTSs

Example

$P(f(x))$:- $P(x)$ $\qquad$ $Q(f(x))$:- $R(x)$ $\qquad$ $R(f(x))$:- $Q(x)$

$$P \sim Q \sim R$$

perpetual processes

Some References

Corradini, A., Montanari, U.: An algebraic semantics for structured transition systems and its applications to logic programs. Theor. Comput. Sci. **103**(1), 51–106 (1992)

Komendantskaya, E., Power, J.: Logic programming: laxness and saturation. J. Log. Algebr. Meth. Program. **101**, 1–21 (2018)

Bruni, R., Montanari, U., Rossi, F.: An interactive semantics of logic programming. Theory Pract. Logic Program. **1**(6), 647–690 (2001)

Bonchi, F., Zanasi, F.: Bialgebraic semantics for logic programming. Logical Methods Comput. Sci. **11**(1), 1–47 (2015)

M. S. Alvim et al. (Eds.): Palamidessi Festschrift, LNCS 11760, pp. 223–240, 2019.
https://doi.org/10.1007/978-3-030-31175-9_13

Conclusion

Algebraic structure of goals: substitutive monoids
Operational semantics: structured coalgebra
Bisimilarity is a congruence