Semantics for Probability and Concurrency

Ana Sokolova

IFIP WG 1.3 meeting Berlin, 4-9-17
Rigorous methods for engineering of and reasoning about reactive systems
Rigorous methods for engineering of and reasoning about reactive systems
Background big picture
Background big picture

Computer Science
Background big picture

Computer Science

Theoretical Computer Science
Background big picture

- Computer Science
- Theoretical Computer Science
- Concurrency
Background big picture

Computer Science

Theoretical Computer Science

Concurrency

Algebra and Coalgebra

Formal Methods
Background big picture

Computer Science

Theoretical Computer Science

Concurrency

Algebra and Coalgebra

Formal Methods

Probabilistic Systems

Ana Sokolova

IFIP WG 1.3, Berlin 4-9-17
Background big picture

Computer Science

Theoretical Computer Science

Concurrency

Security

Algebra and Coalgebra

Formal Methods

Probabilistic Systems

Ana Sokolova

IFIP WG 1.3, Berlin 4-9-17
Background big picture

Computer Science

Theoretical Computer Science

Concurrency

Security

Algebra and Coalgebra

Formal Methods

Probabilistic Systems

Real-Time Systems

IFIP WG 1.3, Berlin 4-9-17
Background big picture

Computer Science

Theoretical Computer Science

Concurrency

Security

Algebra and Coalgebra

Probabilistic Systems

Formal Methods

Memory Management Systems

Real-Time Systems
Background big picture

Computer Science

Theoretical Computer Science

Data Structures

Concurrency

Security

Algebra and Coalgebra

Formal Methods

Probabilistic Systems

Memory Management Systems

Real-Time Systems
Background big picture

Computer Science

- Theoretical Computer Science
- Data Structures
- Formal Methods
- Probabilistic Systems
- Algebra and Coalgebra

Security

Concurrency

Real-Time Systems

Memory Management Systems

Ana Sokolova
UNIVERSITY OF SALZBURG

IFIP WG 1.3, Berlin 4-9-17
Current favourites

Computer Science

Data Structures

Theoretical Computer Science

Security

Algebra and Coalgebra

Formal Methods

Probabilistic Systems

Concurrency

Real-Time Systems

Memory Management Systems

Ana Sokolova

IFIP WG 1.3, Berlin 4-9-17
Part I
Coalgebra/algebra + probability highlights
Part I
Coalgebra/algebra + probability highlights

Mathematical framework based on category theory for state-based systems semantics
Coalgebra/algebra + probability highlights

**Mathematical framework based on category theory for state-based systems semantics**

A. S. Probabilistic systems coalgebraically TCS’11

B. Jacobs, I. Hasuo, A. S. Generic trace semantics via coinduction LMCS’07

B. Jacobs, I. Hasuo, A. S. The microcosm principle and concurrency in coalgebra FoSSaCS’08

A. Silva, A. S. Sound and complete axiomatisation of trace semantics for probabilistic systems MFPS’11

B. Jacobs, A. Silva, A. S. Trace semantics via determinization JSS’15

A. S., H. Woracek Congruences of convex algebras JPAA’15

A. S., H. Woracek Termination in convex sets of distributions CALCO’17

F. Bonchi, A. Silva, A. S. The power of convex algebras CONCUR’17
Joint work with

Erik de Vink  
TU/e

Bart Jacobs  
Radboud University

Ichiro Hasuo  
The University of Tokyo

Harald Woracek
TU WEN

Alexandra Silva
UCL

Filippo Bonchi
ENS de Lyon

Ana Sokolova
Modelling discrete probabilistic systems

Probability distribution functor on $\textbf{Sets}$

$$\mathcal{D}X = \{ \mu : X \rightarrow [0, 1] \mid \sum_{x \in X} \mu(x) = 1 \}$$

for $f : X \rightarrow Y$ we have $\mathcal{D}f : \mathcal{D}X \rightarrow \mathcal{D}Y$ by

$$\mathcal{D}f(\mu)(y) = \sum_{x \in f^{-1}(y)} \mu(x) = \mu(f^{-1}(y))$$
Modelling discrete probabilistic systems

Probability distribution functor on \textbf{Sets}

\[ \mathcal{D} X = \{ \mu : X \rightarrow [0, 1] \mid \sum_{x \in X} \mu(x) = 1 \} \]

and its variants

\[ \mathcal{D}_{\leq 1} X = \{ \mu : X \rightarrow [0, 1] \mid \sum_{x \in X} \mu(x) \leq 1 \} \]

\[ \mathcal{D}_f X = \{ \mu : X \rightarrow [0, 1] \mid \sum_{x \in X} \mu(x) = 1, \text{supp}(\mu) \text{ is finite} \} \]
Modelling discrete probabilistic systems

Almost all known probabilistic systems can be modelled as coalgebras on \( \textbf{Sets} \) for functors given by the following grammar:

\[
F: = - | A | \mathcal{D} | \mathcal{P} | F^A | F + F | F \circ F | F \times F
\]

in all cases concrete and coalgebraic bisimilarity (and behavioural equivalence) coincide.
Almost all known probabilistic systems can be modelled as coalgebras on \textbf{Sets} for functors given by the following grammar:

\[ F : = \emptyset \mid A \mid \mathcal{D} \mid \mathcal{P} \mid F^A \mid F + F \mid F \circ F \mid F \times F \]

in all cases concrete and coalgebraic bisimilarity (and behavioural equivalence) coincide.
Examples
Examples

NFA

\[ 2 \times (\mathcal{P}(-))^A \cong \mathcal{P} (1 + A \times (-)) \]
Examples

\[ 2 \times (P(-))^A \cong P(1 + A \times (-)) \]

\[ D(1 + A \times (-)) \]

NFA

Generative PTS
Examples

NFA

\[ 2 \times (\mathcal{P}(-))^A \equiv \mathcal{P} (1 + A \times (-)) \]

Generative PTS

\[ \mathcal{D} (1 + A \times (-)) \]

Simple PA

\[ \mathcal{P} (A \times \mathcal{D}(-)) \]
Examples

NFA

\[ 2 \times (\mathcal{P}(-))^A \cong \mathcal{P}(1 + A \times (-)) \]

\[
\begin{array}{c}
\xrightarrow{a} x_1 \xrightarrow{a} x_3 \xleftarrow{b} \\
\downarrow \ast \\
\end{array}
\]

Generative PTS

\[ \mathcal{D}(1 + A \times (-)) \]

\[
\begin{array}{c}
\xrightarrow{a, \frac{1}{2}} x_1 \xrightarrow{a, \frac{1}{4}} \\
\xrightarrow{b, \frac{1}{3}} x_4 \xleftarrow{1} \downarrow \ast \\
\xrightarrow{c, \frac{1}{2}} x_5 \downarrow 1 \\
\end{array}
\]

Simple PA

\[ \mathcal{P}(A \times \mathcal{D}(-)) \]

\[
\begin{array}{c}
\xleftarrow{\frac{1}{2}} x_2 \xleftarrow{\frac{1}{3}} x_3 \xleftarrow{\frac{1}{2}} x_4 \\
\xrightarrow{a} x_1 \xrightarrow{a} \\
\xleftarrow{a} \xrightarrow{\frac{1}{2}} x_2 \xrightarrow{b} x_3 \xrightarrow{b} x_4 \\
\end{array}
\]

Here \( \mathcal{D} \) for \( \mathcal{D}_{\leq 1} \)
Expressiveness hierarchy

F. Bartels, A.S., E. de Vink ’03/’04
Expressiveness hierarchy

F. Bartels, A.S., E. de Vink ’03/’04

MG

PZ

Var

Bun

Seg

SSeg

React

LTS

DLTS

Gen

Alt

Str

MC

translación que preserva y refleja la bisimilaridad
Traces ?
Traces?

Generative PTS

\[ \varnothing (1 + A \times (-)) \]

\[
\begin{array}{c}
 a, \frac{1}{2} & x_1 & a, \frac{1}{4} \\
 x_2 & & x_3 \\
b, \frac{1}{3} & y & c, \frac{1}{2} \\
x_4 & y & x_5 \\
1 & y & 1 \\
* & y & * \\
\end{array}
\]
Traces ?

Generative PTS

\[ \mathcal{D} (1 + A \times (-)) \]

\[
\begin{array}{c|c}
  a, \frac{1}{2} & x_1 \\
  b, \frac{1}{3} & x_2 \\
  x_4 & \vdash c, \frac{1}{2} \\
  1 & \vdash 1 \\
  * & * \\
\end{array}
\]

\[
\text{tr}(x_1)(ab) = \frac{1}{6} \quad \text{tr}(x_1)(ac) = \frac{1}{8}
\]
Traces?

Generative PTS

\[\mathcal{D} (1 + A \times (-))\]

\[
\begin{align*}
& a, \frac{1}{2} \quad x_1 \quad a, \frac{1}{4} \\
& x_2 \quad x_3 \\
& b, \frac{1}{3} \quad x_4 \quad c, \frac{1}{2} \\
& x_5 \\
& 1 \quad 1 \\
& * \quad * \\
\end{align*}
\]

\[
\text{tr}(x_1)(ab) = \frac{1}{6} \quad \text{tr}(x_1)(ac) = \frac{1}{8}
\]

\[
\text{tr} : X \rightarrow DA^*
\]
Traces?

Generative PTS

\( \mathcal{D} (1 + A \times (-)) \)

\[
\begin{align*}
& x_1 \quad a, \frac{1}{4} \\
& x_2 \quad \frac{1}{2} \\
& x_3 \quad b, \frac{1}{3} \\
& x_4 \quad \psi \\
& x_5 \quad c, \frac{1}{2} \\
& 1 \quad \psi \\
& * \quad * 
\end{align*}
\]

\( \text{tr} (x_1)(ab) = \frac{1}{6} \quad \text{tr} (x_1)(ac) = \frac{1}{8} \)

\( \text{tr}: X \rightarrow \mathcal{D}A^* \)

\( \mathcal{D} \) is a monad
Traces?

Generative PTS

\[ \mathcal{D} (1 + A \times (-)) \]

\[
\begin{align*}
  a, \frac{1}{2} & \quad x_1 & a, \frac{1}{4} \\
  x_2 & & x_3 \\
  b, \frac{1}{3} & \Psi & \Psi c, \frac{1}{2} \\
  x_4 & & x_5 \\
  1 & \Psi & \Psi 1 \\
  * & & * \\
\end{align*}
\]

\[
\text{tr}(x_1)(ab) = \frac{1}{6} \quad \text{tr}(x_1)(ac) = \frac{1}{8}
\]

\( \text{tr}: X \to \mathcal{DA}^* \)

\( \mathcal{D} \) is a monad

arrow in \( \mathcal{KL}(\mathcal{D}) \)
Traces?

Generative PTS

\[ \mathcal{D} (1 + A \times (-)) \]

\[
\begin{array}{cccccc}
    a, \frac{1}{2} & \xrightarrow{x_1} & a, \frac{1}{4} \\
    x_2 & \xrightarrow{x_3} & x_3 \\
    b, \frac{1}{3} & \xrightarrow{x_4} & q, \frac{1}{2} \\
    x_4 & \xrightarrow{x_5} & x_5 \\
    1 & \xrightarrow{1} & 1 \\
    * & \xrightarrow{*} & *
\end{array}
\]

\[ \text{tr}(x_1)(ab) = \frac{1}{6} \quad \text{tr}(x_1)(ac) = \frac{1}{8} \]

\[ \text{tr}: X \rightarrow \mathcal{D}A^* \]

\[ \mathcal{D} \text{ is a monad} \]

\[ \text{lifts to } \mathcal{Kl}(\mathcal{D}) \text{ via a distributive law} \]

\[ \mathcal{D} \text{ is a monad} \]

\[ \text{arrow in } \mathcal{Kl}(\mathcal{D}) \]
Traces?

Generative PTS

\( \mathcal{D} (1 + A \times (-)) \)

\[
\begin{align*}
  a, \frac{1}{2} & \quad x_1 & \quad a, \frac{1}{4} \\
  x_2 & \quad x_3 & \\
  b, \frac{1}{3} & \quad c, \frac{1}{2} & \\
  x_4 & \quad x_5 \\
  1 & \quad 1 \\
  * & \quad *
\end{align*}
\]

lifts to \( \mathcal{Kl}(\mathcal{D}) \) via a distributive law

\[
\begin{align*}
  \text{tr}(x_1)(ab) &= \frac{1}{6} \\
  \text{tr}(x_1)(ac) &= \frac{1}{8}
\end{align*}
\]

\( \text{tr} : X \rightarrow \mathcal{D}A^* \)

\( \mathcal{D} \) is a monad

\[
X \rightarrow \mathcal{D}(1 + A \times X) \rightarrow \mathcal{D}(1 + A \times \mathcal{D}(1 + A \times X)) \rightarrow \mathcal{D}^2(1 + A \times (1 + A \times X)) \rightarrow \mathcal{D}(1 + A \times X + A^2 \times X) \cdots
\]
Traces via determinisation
Traces via determinisation

Generative PTS

\[ \emptyset (1 + A \times (-)) \]

- \( a, \frac{1}{2} \) \( \xrightarrow{} \) \( x_1 \) \( \xrightarrow{} \) \( a, \frac{1}{4} \)
- \( b, \frac{1}{3} \) \( \xrightarrow{} \) \( x_2 \)
- \( x_4 \) \( \xrightarrow{} \) \( x_3 \) \( \xrightarrow{} \) \( c, \frac{1}{2} \)
- \( 1 \) \( \xrightarrow{} \) \( x_5 \)
- \( \ast \) \( \xrightarrow{} \) \( 1 \)
- \( \ast \) \( \xrightarrow{} \) \( \ast \)
Traces via determinisation

Generative PTS

\( \mathcal{D} (1 + A \times (-)) \)

\[
\begin{align*}
  a, \frac{1}{2} & \quad x_1 & a, \frac{1}{4} \\
  x_2 & \quad x_3 \\
  b, \frac{1}{3} & \quad c, \frac{1}{2} \\
  x_4 & \quad x_5 \\
  1 & \quad 1 \\
  * & \quad *
\end{align*}
\]

\[
\text{tr}(x_1)(ab) = \frac{1}{6} \quad \text{tr}(x_1)(ac) = \frac{1}{8}
\]
Traces via determinisation

Generative PTS

\( \mathcal{D} (1 + A \times (-)) \)

\[
\begin{align*}
&\begin{array}{c}
a, \frac{1}{2} \\
b, \frac{1}{3} \\
1
\end{array} \rightarrow
\begin{array}{c}
x_2 \\
x_4 \\
*
\end{array} &
\begin{array}{c}
a, \frac{1}{4} \\
\uparrow c, \frac{1}{2} \\
\uparrow 1
\end{array} \rightarrow
\begin{array}{c}
x_3 \\
x_5 \\
*
\end{array}
\end{align*}
\]

\[
\text{tr}(x_1)(ab) = \frac{1}{6} \quad \text{tr}(x_1)(ac) = \frac{1}{8}
\]

\( \text{tr}: X \rightarrow DA^* \)
Traces via determinisation

Generative PTS

\[ \mathcal{D} (1 + A \times (-)) \]

\[
\begin{array}{cccc}
a, \frac{1}{2} & x_1 & a, \frac{1}{4} \\
b, \frac{1}{3} & x_2 & & x_3 \leftarrow c, \frac{1}{2} \\
\downarrow & x_4 & & x_5 \\
1 & \downarrow & & 1 \\
* & & & *
\end{array}
\]

\[ \text{tr}(x_1)(ab) = \frac{1}{6} \quad \text{tr}(x_1)(ac) = \frac{1}{8} \]

\( \text{tr}: X \rightarrow \mathcal{DA}^* \)

trace = bisimilarity after determinisation
Traces via determinisation
Traces via determinisation

Generative PTS

\[ \mathcal{D} (1 + A \times (-)) \]

\[
\begin{align*}
  &a, \frac{1}{2} & x_1 & a, \frac{1}{4} \\
  &b, \frac{1}{3} & \psi & x_2 & x_3 & \psi, c, \frac{1}{2} \\
  &1 & \psi & x_4 & x_5 & \psi, 1 \\
  &* & & * & & *
\end{align*}
\]
Traces via determinisation

Generative PTS

\( \emptyset (1 + A \times (-)) \)

\[
\begin{array}{cccc}
  a, \frac{1}{2} & \rightarrow & x_1 & \rightarrow & a, \frac{1}{4} \\
  x_2 & \rightarrow & x_3 \\
  \downarrow b, \frac{1}{3} & \rightarrow & \downarrow c, \frac{1}{2} \\
  x_4 & \rightarrow & x_5 \\
  \downarrow 1 & \rightarrow & \downarrow 1 \\
  * & \rightarrow & * \\
\end{array}
\]
Traces via determinisation

Generative PTS

\[ \mathcal{D} \left( 1 + A \times (-) \right) \]

DFA

\[ [0, 1] \times (-)^{A} \text{ states } \mathcal{D}(-) \]
Traces via determinisation

Generative PTS
\[ \mathcal{D} (1 + A \times (-)) \]

\[
\begin{align*}
& a, \frac{1}{2} & x_1 & a, \frac{1}{4} \\
& b, \frac{1}{3} & x_2 & x_3 & c, \frac{1}{2} \\
& 1 & x_4 & x_5 & 1 \\
& * & * & * & * 
\end{align*}
\]

DFA
\[ [0, 1] \times (-)^A \]

states \[ \mathcal{D}(-) \]

\[
\begin{align*}
& x_1 & \downarrow a \\
& b & \frac{1}{2} x_2 + \frac{1}{4} x_3 & c \\
& \frac{1}{6} x_4 & \frac{1}{6} & \frac{1}{8} x_5 & \frac{1}{8} \\
& * & * & * & * 
\end{align*}
\]

trace = bisimilarity after determinisation

Ana Sokolova
Traces via determinisation

Generative PTS
\[ \mathcal{D} (1 + A \times (-)) \]

- \( a, \frac{1}{2} \) \( \uparrow \) \( x_1 \)
- \( a, \frac{1}{4} \) \( \uparrow \) \( x_1 \)
- \( x_2 \)
- \( b, \frac{1}{3} \) \( \uparrow \) \( x_4 \)
- \( 1 \) \( \uparrow \) *
- \( * \)

- \( x_3 \)
- \( \frac{1}{2} x_2 + \frac{1}{4} x_3 \)
- \( \frac{1}{6} x_4 \)
- \( \frac{1}{8} x_5 \)
- \( * \)

DFA
\[ [0,1] \times (-)^A \]

states \( \mathcal{D}(-) \)

trace = bisimilarity after determinisation

Happens in \( \mathcal{EM}(\mathcal{D}) \)
Traces via determinisation

Generative PTS

\[ \mathcal{D} (1 + A x (-)) \]

DFA

\[ [0,1] \times (-)^A \]

\begin{align*}
DFA\ states & \mathcal{D}(-) \\
& \\\ntrace & = \text{bisimilarity after determinisation}
\end{align*}
Traces via determinisation

Generative PTS
\[ \mathcal{O} (1 + A \times (-)) \]

DFA
\[ [0,1] \times (-)^A \]

trace = bisimilarity after determinisation

Happens in \( \mathcal{E}M(\mathcal{D}) \)

(positive) convex algebras

we recently generalised this to PA too
Trace axioms for generative PTS

Axioms for bisimilarity

\[ p \cdot a \cdot (p_1 E_1 \oplus p_2 E_2) \equiv p_1 \cdot a \cdot p E_1 \oplus p_2 \cdot a \cdot p E_2 \quad (D) \]
Trace axioms for generative PTS

Axioms for bisimilarity

\[ p \cdot a \cdot (p_1E_1 \oplus p_2E_2) \equiv p_1 \cdot a \cdot pE_1 \oplus p_2 \cdot a \cdot pE_2 \quad (D) \]

soundness and completeness !?
Trace axioms for generative PTS

Axioms for bisimilarity

\[ p \cdot a \cdot (p_1 E_1 \oplus p_2 E_2) \equiv p_1 \cdot a \cdot p E_1 \oplus p_2 \cdot a \cdot p E_2 \quad (D) \]

soundness and completeness !?

Happens in \( EM(D) \)

Ana Sokolova

IFIP WG 1.3, Berlin 4-9-17
Trace axioms for generative PTS

Axioms for bisimilarity

\[ p \cdot a \cdot (p_1 E_1 \oplus p_2 E_2) \equiv p_1 \cdot a \cdot pE_1 \oplus p_2 \cdot a \cdot pE_2 \ (D) \]

Happens in \( E_M(D) \)

(positive) convex algebras

soundness and completeness !?
Trace axioms for generative PTS
Trace axioms for generative PTS

Generative PTS

\[\mathcal{D} (1 + A \times (-))\]
Trace axioms for generative PTS

Generative PTS

\[ \mathcal{D}(1 + A \times (-)) \]
Trace axioms for generative PTS

Generative PTS
\[ \mathcal{D} (1 + A \times (-)) \]

\[
\begin{align*}
\left( \frac{1}{2} & \cdot a \cdot \frac{1}{3} \cdot b \cdot 1 \cdot * \right) \oplus \left( \frac{1}{4} & \cdot a \cdot \frac{1}{2} \cdot c \cdot 1 \cdot * \right) \quad (Cong) \\
\equiv & \quad \left( \frac{1}{2} \cdot a \cdot \frac{1}{3} \cdot b \cdot 1 \cdot * \right) \oplus \left( \frac{1}{2} \cdot a \cdot \frac{1}{4} \cdot c \cdot 1 \cdot * \right) \\
\quad (D) & \quad \equiv \quad \frac{1}{2} \cdot a \cdot \left( \frac{1}{3} \cdot b \cdot 1 \cdot * \oplus \frac{1}{4} \cdot c \cdot 1 \cdot * \right)
\end{align*}
\]
Trace axioms for generative PTS

Generative PTS

\[ \mathcal{D} (1 + A \times (-)) \]

\[
\begin{align*}
\frac{1}{4} & \cdot a \cdot \frac{1}{2} \cdot c \cdot 1 \cdot * \\
\end{align*}
\]

\[
\begin{align*}
\frac{1}{2} & \cdot a \cdot \frac{1}{3} \cdot b \cdot 1 \cdot * \\
\end{align*}
\]

\[
\begin{align*}
\frac{1}{2} & \cdot a \cdot \frac{1}{3} \cdot b \cdot 1 \cdot * \\
\end{align*}
\]

\[
\begin{align*}
\frac{1}{2} & \cdot a \cdot \frac{1}{3} \cdot b \cdot 1 \cdot * \\
\end{align*}
\]

\[
\begin{align*}
\frac{1}{2} & \cdot a \cdot \frac{1}{3} \cdot b \cdot 1 \cdot * \\
\end{align*}
\]

\[
\begin{align*}
\frac{1}{2} & \cdot a \cdot \frac{1}{3} \cdot b \cdot 1 \cdot * \\
\end{align*}
\]

\[
\begin{align*}
\frac{1}{2} & \cdot a \cdot \frac{1}{3} \cdot b \cdot 1 \cdot * \\
\end{align*}
\]
The quest for completeness

Inspired lots of new research:

• A. S., H. Woracek *Congruences of convex algebras* JPAA’15

• S. Milius *Proper functors* CALCO’17
The quest for completeness

Inspired lots of new research:

• A. S., H. Woracek *Congruences of convex algebras* JPAA’15

• S. Milius *Proper functors* CALCO’17

\[ \text{f.p.} = \text{f.g. for (positive) convex algebras} \]
The quest for completeness

Inspired lots of new research:

• A. S., H. Woracek *Congruences of convex algebras* JPAA’15

• S. Milius *Proper functors* CALCO’17

\[ f.p. = f.g. \text{ for (positive) convex algebras} \]

if f.p. = f.g. and then completeness
The quest for completeness

Inspired lots of new research:

• A. S., H. Woracek *Congruences of convex algebras* JPAA’15

• S. Milius *Proper functors* CALCO’17

\[ f.p. = f.g. \text{ for (positive) convex algebras} \]

if \( f.p. = f.g. \) and then completeness does not hold
The quest for completeness

Inspired lots of new research:

- A. S., H. Woracek *Congruences of convex algebras* JPAA’15

- S. Milius *Proper functors* CALCO’17

Our axiomatisation would be proven complete if only one particular functor $\hat{G}$ on $\mathcal{EM}(D)$ were proper.

If $f.p. = f.g.$ and then completeness does not hold.

f.p. = f.g. for (positive) convex algebras
Part II
Proper convex functors
Part II
Proper convex functors

the trace axioms can be proven complete!
Part II
Proper convex functors

the trace axioms can be proven complete!

very new nontrivial results!
joint work with

Harald Woracek
A functor $F$ on an algebraic category is proper, if

- for any two $F$-coalgebras with free f.g. carriers $TX \longrightarrow FTX$ and $TY \longrightarrow FTY$,
- for any two points $x$ in $TX$, $y$ in $TY$ with $\eta(x) \sim \eta(y)$

there is a zigzag of $F$-coalgebras with free f.g. carriers that relates $x$ and $y$.

extends the notion of a proper semiring of Ésik and Maletti

A semiring $S$ is proper iff $S \times (-)^A$ is proper
Proper functors

A functor $\mathcal{F}$ on an algebraic category is proper, if

- for any two $\mathcal{F}$-coalgebras with free f.g. carriers $TX \xrightarrow{} FTX$ and $TY \xrightarrow{} FTY$
- for any two points $x$ in $TX$, $y$ in $TY$ with $\eta(x) \sim \eta(y)$ there is a zigzag of $\mathcal{F}$-coalgebras with free f.g. carriers that relates $x$ and $y$

extends the notion of a proper semiring of Ésik and Maletti

a semiring $S$ is proper iff $S \times (-)^A$ is proper
Proper functors

A functor F on an algebraic category is proper, if

- for any two F-coalgebras with free f.g. carriers $TX \xrightarrow{} FTX$ and $TY \xrightarrow{} FTY$
- for any two points $x$ in $TX$, $y$ in $TY$ with $\eta(x) \sim \eta(y)$

there is a zigzag of F-coalgebras with free f.g. carriers that relates $x$ and $y$
Proper functors

A functor $F$ on an algebraic category is proper, if

1. for any two $F$-coalgebras with free f.g. carriers $TX \to FTX$ and $TY \to FTY$
2. for any two points $x$ in $TX$, $y$ in $TY$ with $\eta(x) \sim \eta(y)$

there is a zigzag of $F$-coalgebras with free f.g. carriers that relates $x$ and $y$.

extends the notion of a proper semiring of Ésik and Maletti

a semiring $S$ is proper iff $S \times (-)^A$ is proper.
Proper functors

A functor $F$ on an algebraic category is proper, if

- for any two $F$-coalgebras with free f.g. carriers $TX \xrightarrow{\eta} FTX$ and $TY \xrightarrow{\eta} FTY$
- for any two points $x$ in $TX$, $y$ in $TY$ with $\eta(x) \sim \eta(y)$

there is a zigzag of $F$-coalgebras with free f.g. carriers that relates $x$ and $y$

extends the notion of a proper semiring of Ésik and Maletti

A semiring $S$ is proper iff $S \times (-)^A$ is proper
Proper functors

extends the notion of a proper semiring of Ésik and Maletti

a semiring
S is proper iff $S \times (-)^A$

is proper
Proper functors

Known

• any Noetherian semiring is proper, hence \( \mathbb{Z}, \mathbb{R} \) are proper
• \( \mathbb{N} \) is proper

extends the notion of a proper semiring of Ésik and Maletti

a semiring \( S \) is proper iff \( S \times (-)^A \) is proper
Proper functors

Known

- any Noetherian semiring is proper, hence \( \mathbb{Z}, \mathbb{R} \) are proper
- \( \mathbb{N} \) is proper

Open

- If \( \mathbb{R}^+ \) is proper
- If \([0,1] \times (-)^A\) is proper on \((P)CA\)
- If \( \hat{G} \) on PCA is proper

extends the notion of a proper semiring of Ésik and Maletti

a semiring

\( S \) is proper iff \( S \times (-)^A \) is proper
Proper functors

Known

- any Noetherian semiring is proper, hence \(\mathbb{Z}, \mathbb{R}\) are proper
- \(\mathbb{N}\) is proper

Open

- If \(\mathbb{R}^+\) is proper
- If \([0,1] \times (-)^A\) is proper on (P)CA
- If \(\hat{G}\) on PCA is proper

extends the notion of a proper semiring of Ésik and Maletti

we have just proven all these proper

a semiring
\(S\) is proper iff \(S \times (-)^A\) is proper
Proper functors

**Known**
- any Noetherian semiring is proper, hence \( \mathbb{Z}, \mathbb{R} \) are proper
- \( \mathbb{N} \) is proper

**Open**
- If \( \mathbb{R}^+ \) is proper
- If \([0,1] \times (-)^A\) is proper on (P)CA
- If \( \hat{G} \) on PCA is proper

Extends the notion of a proper semiring of Ésik and Maletti

A semiring \( S \) is proper iff \( S \times (-)^A \) is proper

We have just proven all these proper nontrivial all
Proper functors

Known
• any Noetherian semiring is proper, hence \( \mathbb{Z}, \mathbb{R} \) are proper
• \( \mathbb{N} \) is proper

Open
• If \( \mathbb{R}^+ \) is proper
• If \([0,1] \times (-)^A\) is proper on (P)CA
• If \( \hat{G} \) on PCA is proper

difficult case

we have just proven all these proper

nontrivial all

extends the notion of a proper semiring of Ésik and Maletti

\[ S \text{ is proper iff } S \times (-)^A \text{ is proper} \]
Proper functors

extends the notion of a proper semiring of Ésik and Maletti

A semiring
S is proper iff $S \times (\cdot)^A$
is proper
Proper functors

Our results prove

- \( \mathbb{N} \) is proper
- \( \mathbb{R}^+ \) is proper
- \([0,1] \times (-)^A\) is proper on PCA

extends the notion of a proper semiring of Ésik and Maletti

A semiring
- \( S \) is proper iff \( S \times (-)^A \) is proper
Proper functors

Our results prove

- \( \mathbb{N} \) is proper
- \( \mathbb{R}^+ \) is proper
- \([0,1] \times (-)^A\) is proper on PCA
Proper functors

Our results prove:

- $\mathbb{N}$ is proper
- $\mathbb{R}^+$ is proper
- $[0,1] \times (-)^A$ is proper on PCA

via “scalar extensions”
Proper functors

Our results prove:

- \( \mathbb{N} \) is proper
- \( \mathbb{R}^+ \) is proper
- \([0,1] \times (-)^A\) is proper on PCA

via “scalar extensions”

extension to \( \mathbb{Z} \)
Proper functors

Our results prove

- $\mathbb{N}$ is proper
- $\mathbb{R}^+$ is proper
- $[0,1] \times (-)^A$ is proper on PCA

via “scalar extensions”

extension to $\mathbb{Z}$

extension to $\mathbb{R}$
Proper functors

Our results prove:

- \( \mathbb{N} \) is proper
- \( \mathbb{R}^+ \) is proper
- \([0,1] \times (-)^A\) is proper on PCA

Extension to \( \mathbb{Z} \)

Extension to \( \mathbb{R} \)

Via "scalar extensions"

Extension to \( \mathbb{R} \times (-)^A \)
Proper functors

- \(\mathbb{N}\) is proper
- \(\mathbb{R}^+\) is proper
- \([0,1] \times (-)^A\) is proper on PCA

Our results prove:

- Extension to \(\mathbb{Z}\)
- Extension to \(\mathbb{R}\)
- Extension to \(\mathbb{R} \times (-)^A\)

via "scalar extensions"

In all cases one span suffices.
Proper functors

Our results prove:

- $\mathbb{N}$ is proper
- $\mathbb{R}^+$ is proper
- $[0,1] \times (-)^A$ is proper on PCA

As well as the difficult case:

- $\hat{G}$ on PCA is proper

Extension to $\mathbb{Z}$

Extension to $\mathbb{R}$

Extension to $\mathbb{R} \times (-)^A$

Via “scalar extensions”

In all cases one span suffices
Proper functors

Our results prove:

- \( \mathbb{N} \) is proper
- \( \mathbb{R}^+ \) is proper
- \([0,1] \times (-)\) is proper on PCA

As well as the difficult case:

- \( \hat{\mathcal{G}} \) on PCA is proper

\[\begin{align*}
\text{extension to } \mathbb{Z} \\
\text{extension to } \mathbb{R} \\
\text{via "scalar extensions"} \\
\text{extension to } \mathbb{R} \times (-)^A \\
\text{via Kakutani fixed-point theorem}
\end{align*}\]

in all cases one span suffices
Proper functors

Our results prove:

- \( \mathbb{N} \) is proper
- \( \mathbb{R}^+ \) is proper
- \([0,1] \times (-)^A\) is proper on PCA

As well as the difficult case:

- \( \hat{G} \) on PCA is proper

via "scalar extensions" and via Kakutani fixed-point theorem

In all cases one span suffices.

Here a zigzag is needed.
Our results prove

- convexity matters for various results in semantics / analysis of probabilistic systems
- Taking a general approach pays off
Proper functors

Our results prove

- convexity matters for various results in semantics / analysis of probabilistic systems
- Taking a general approach pays off

no concrete proof of completeness found
Proper functors

Our results prove

- convexity matters for various results in semantics / analysis of probabilistic systems
- Taking a general approach pays off

no concrete proof of completeness found

Thank You!