Coalgebras for Binary Methods

Hendrik Tews

Institute of Theoretical Computer Science, TU Dresden

tews@tcs.inf.tu-dresden.de

http://wwwtcs.inf.tu-dresden.de/∼tews/

Coalgebras can model Classes of object-oriented languages

class Point
  methods
    get_x : **Self** $\longrightarrow$ real
    get_y : **Self** $\longrightarrow$ real
    move : **Self** $\times$ real $\times$ real $\longrightarrow$ **Self**
end Point

Coalgebra $c : X \longrightarrow T_P(X)$ with

$$T(X) = \mathbb{R} \times \mathbb{R} \times (\mathbb{R} \times \mathbb{R} \Rightarrow X)$$

Abstract theory delivers

- **Bisimulation** as behavioural indistinguishability
- **Invariance**
- **Modal Operators** for an expressive logic
- **Coinduction** as definition and reasoning principle
- **Semantics for class specifications** using the final (most abstract) coalgebra

Built into the Coalgebraic Class Specification Language CCSL [RTJ01].

Endofunctors do not cover binary methods!

(Binary methods are difficult to handle in all theoretical approaches to object-oriented programming.)

Binary Methods—Example

<u>Class</u> A = {

<u>method</u> equal(x : A) = x == <u>this</u> ;

<u>method</u> memory(env : $\mathbb{N} \longrightarrow A$) = $\cdots$ env(5).equal(<u>this</u>) $\cdots$;

<u>method</u> strange(f : A $\longrightarrow$ A) = $\cdots$;

}

Assume, that X is the state space of the class

$$
\begin{array}{lcl}
\mathsf{equal} & : & X \longrightarrow (X \longrightarrow \mathsf{bool}) \\
\mathsf{memory} & : & X \longrightarrow (\mathbb{N} \longrightarrow X) \longrightarrow \boxed{\cdots} \\
\mathsf{strange} & : & X \longrightarrow (X \longrightarrow X) \longrightarrow \boxed{\cdots}
\end{array}
$$

Possible solutions

- Allow binary methods as definitional extensions only
- Use algebraic extensions of coalgebras [HK99]
- Extend underlying framework

Contents

Extending polynomial functors

Class of Functors	Property	Example
cartesian	no arguments	$\textsf{Self} \longrightarrow \boxed{\cdots}$
polynomial	constant arguments	$\textsf{Self} \times A \longrightarrow \boxed{\cdots}$
extended cartesian	no functional arguments	$\textsf{Self} \times \textsf{Self} \times \textsf{Self} \longrightarrow \boxed{\cdots}$ $\textsf{Self} \times (\textsf{Self} + A) \longrightarrow \boxed{\cdots}$
extended polynomial	functional arguments with constant domain	$\textsf{Self} \times (A \Rightarrow \textsf{Self}) \longrightarrow \boxed{\cdots}$
higher-order polynomial	arbitrary arguments	$\textsf{Self} \times (\textsf{Self} \Rightarrow A) \longrightarrow \boxed{\cdots}$ $\textsf{Self} \times (\textsf{Self} \Rightarrow \textsf{Self}) \longrightarrow \boxed{\cdots}$

Recall Polynomial Functors

Polynomial functor $\quad \mathbf{Sets} \longrightarrow \mathbf{Sets}$

$$F(X) \quad = \quad X \quad | \quad A \quad | \quad F_1(X) + F_2(X) \quad | \quad F_1(X) \times F_2(X) \quad | \quad A \Rightarrow F_1(X)$$

Invariants & Bisimulations according to Aczel & Mendler

$$\begin{tikzcd} P \subseteq X \arrow[r, "\iota"] \arrow[d, dashed] & X \arrow[d, "c"] \\ F(P) \arrow[r, "F(\iota)"] & F(X) \end{tikzcd} \qquad \begin{tikzcd} X \arrow[d, "c"'] & R \arrow[l, "\pi_1"'] \arrow[r, "\pi_2"] \arrow[d, dashed] & Y \arrow[d, "d"] \\ F(X) & F(R) \arrow[l, "F(\pi_1)"'] \arrow[r, "F(\pi_2)"] & F(Y) \end{tikzcd}$$

Invariants & Bisimulations according to Hermida & Jacobs

- Define *predicate lifting* and *relation lifting* of F :

$$\mathrm{Pred}(F) \; : \qquad P \subseteq X \longmapsto \mathrm{Pred}(F)(P) \subseteq F(X)$$

$$\mathrm{Rel}(F) \; : \qquad R \subseteq X \times Y \longmapsto \mathrm{Rel}(F)(R) \subseteq F(X) \times F(Y)$$

- P is an *invariant*, if $\quad \forall x \in X \, . \, P(x)$ implies $\mathrm{Pred}(F)(P)(c(x))$
- R is a *bisimulation*, if $\quad \forall x \in X, y \in Y \, . \, R(x, y)$ implies $\mathrm{Rel}(F)(R)(c(x), d(y))$

Let $\mathcal{C}$ be a bicartesian closed category (i.e., containing finite products, coproducts and exponents).

Higher-order polynomial functors

$$H : \mathcal{C}^{op} \times \mathcal{C} \longrightarrow \mathcal{C}$$

$$H(Y, X) \quad = \quad \begin{cases} A \\ X \\ H_1(Y, X) \ \times \ H_2(Y, X) \\ H_1(Y, X) \ + \ H_2(Y, X) \\ H_1(X, Y) \ \Rightarrow \ H_2(Y, X) \end{cases}$$

Coalgebras

are morphisms $X \xrightarrow{\ c\ } H(X, X)$

Morphism of coalgebras

$$\begin{tikzcd} X \arrow[r, "c"] \arrow[dd, "f"'] & H(X,X) \arrow[dr, "{H(X,f)}"] & \\ & & H(X,Y) \\ Y \arrow[r, "d"] & H(Y,Y) \arrow[ur, "{H(f,Y)}"'] & \end{tikzcd}$$

Predicate lifting is an operation on predicates

$$(Q \subseteq Y, \;\; P \subseteq X) \longmapsto \mathsf{Pred}(H)(Q,P) \quad \subseteq \quad H(Y,X)$$

Defined by induction on the structure of H:

H	$\mathsf{Pred}(H)(Q,P)(a)$ for $a \in H(Y,X)$
X	$P(a)$
A	$\top$
$H_1 \times H_2$	$\mathsf{Pred}(H_1)(Q,P)(\pi_1\, a) \;\wedge\; \mathsf{Pred}(H_2)(Q,P)(\pi_2\, a)$
$H_1 + H_2$	$\begin{cases} \text{if } a = \kappa_1\, a_1 & \mathsf{Pred}(H_1)(Q,P)(a_1) \\ \text{if } a = \kappa_2\, a_2 & \mathsf{Pred}(H_2)(Q,P)(a_2) \end{cases}$
$H_1 \Rightarrow H_2$	$\forall u \in \mathsf{Pred}(H_1)(P,Q)\;.\;\mathsf{Pred}(H_2)(Q,P)(a(u))$

Relation lifting (for a h.-o. pol. functor H) is an operation on relations

$$\big(S \subseteq U \times V, \quad R \subseteq X \times Y\big) \longmapsto \mathsf{Rel}(H)(S,R) \quad \subseteq \quad H(U,X) \times H(V,Y)$$

Defined by induction on the structure of H:

H	$\mathrm{Rel}(H)$	$a \in H(U,X) \quad b \in H(V,Y)$
X	R	$R(a,b)$
A	$=_A$	$a =_A b$
$H_1 \times H_2$	$\mathrm{Rel}(H_1) \times_{\mathrm{R}} \mathrm{Rel}(H_2)$	$\mathsf{Rel}(H_1)(S,R)(\pi_1\, a, \pi_1\, b) \;\wedge\; \mathsf{Rel}(H_2)(S,R)(\pi_2\, a, \pi_2\, b)$
$H_1 + H_2$	$\mathrm{Rel}(H_1) +_{\mathrm{R}} \mathrm{Rel}(H_2)$	$\begin{cases} \text{if } (a,b) = (\kappa_1\, a_1, \kappa_1\, b_1) & \mathsf{Rel}(H_1)(S,R)(a_1,b_1) \\ \text{if } (a,b) = (\kappa_2\, a_2, \kappa_2\, b_2) & \mathsf{Rel}(H_2)(S,R)(a_2,b_2) \\ \text{otherwise} & \bot \end{cases}$
$H_1 \Rightarrow H_2$	$\mathrm{Rel}(H_1) \Rightarrow_{\mathrm{R}} \mathrm{Rel}(H_2)$	$\forall (u,v) \in \mathsf{Rel}(H_1)(R,S) \;.\; \mathsf{Rel}(H_2)(S,R)(a(u), b(v))$

Consider a higher-order polynomial functor H and a coalgebras $c : X \longrightarrow H(X,X)$.

Invariants according to Hermida & Jacobs

- P is an invariant for c, if

$$\forall x \in X \,.\, P(x) \quad \text{implies} \quad \mathrm{Pred}(H)(P,P)(c(x))$$

Invariants according to Aczel & Mendler

$$\begin{tikzcd}
P \subseteq X \arrow[rr, "\iota"] \arrow[d] & & X \arrow[d, "c"] \\
H(P,P) \arrow[dr, "H(P,\iota)"'] & & H(X,X) \arrow[dl, "H(\iota,X)"] \\
& H(P,X) &
\end{tikzcd}$$

- Both notions of invariant are different
- Aczel/Mendler invariants are not closed under taking successor states.

Consider a higher-order polynomial functor H and two coalgebras $c : X \longrightarrow H(X,X)$ and $d : Y \longrightarrow H(Y,Y)$.

Bisimulations according to Hermida & Jacobs

- R is a bisimulation for c and d, if

$$\forall x \in X, y \in Y \;.\; R(x,y) \quad \text{implies} \quad \mathrm{Rel}(H)(R,R)(c(x), d(y))$$

Bisimulations according to Aczel & Mendler

$$\begin{tikzcd}
X \arrow[d, "c"'] & & R \arrow[ll, "\pi_1"'] \arrow[rr, "\pi_2"] \arrow[d] & & Y \arrow[d, "d"] \\
H(X,X) \arrow[dr, "H(\pi_1, X)"'] & & H(R,R) \arrow[dl, "H(R,\pi_1)"] \arrow[dr, "H(R,\pi_2)"'] & & H(Y,Y) \arrow[dl, "H(\pi_2, Y)"] \\
& H(R,X) & & H(R,Y) &
\end{tikzcd}$$

- Both notions of bisimulation are different
- Aczel/Mendler bisimulations are not closed under taking successor states.

Proposition 1 *If R is a bisimulation then so is R^{op}.*

Proposition 2 *Equality $=$ is a bisimulation for any coalgebra.*

Proposition 3 *The truth predicate is an invariant for any coalgebra.*

Proposition 4 *If a class signature $\mathcal{A}$ contains a binary method m, then the associated signature functor $H_{\mathcal{A}}$ has a final coalgebra if and only if $H_{\mathcal{A}}(\mathbf{1}, \mathbf{1})$ is a singleton.*

Proof

1. Assume H has a final coalgebra, say $z : Z \longrightarrow H(Z, Z)$.
2. Construct a new coalgebra $z' : Z + 1 \longrightarrow H(Z + 1, Z + 1)$ such that
   - the inclusion $\iota : Z \longrightarrow Z + 1$ is a coalgebra morphism *and*
   - for all $y \in Z$: $m_{z'}(\bot, \iota\, y) \neq m_z(y, y)$.
3. Note, that if $f : Z + 1 \longrightarrow Z$ is a coalgebra morphism $z' \longrightarrow z$, then $f \circ \iota = \mathrm{id}_Z$.
4. Conclude, that there is no such morphism $z' \longrightarrow z$ by construction:

$$m_z(\circledast, \circledast) \quad = \quad m_z(\circledast, (f \circ \iota)\circledast) \quad = \quad m_{z'}(\bot, \iota\, \circledast) \qquad \square$$

Negative Results

In general for coalgebras of higher-order polynomial functors it does **not** hold, that

- bisimulations/invariants are closed under union/intersection
- bisimulations are closed under composition
- there exists greatest bisimulations/invariants
- the image of a morphism is an invariant
- invariants correspond to subcoalgebras
- $\coprod_{\pi_1} R = \{x \mid \exists y \,.\, x\, R\, y\}$ is an invariant for a bisimulation R
- $\pi_1^* P \cap R = \{(x, y) \mid x\, R\, y \wedge P\, x\}$ is a bisimulation for an invariant P and a bisimulation R
- $\coprod_{\delta} P = \{(x, x) \mid P\, x\}$ is a bisimulation for an invariant P
- the graph and the kernel of a morphism are bisimulations

Example: Intersection of Bisimulations

Functor $T(Y, X) \stackrel{\text{def}}{=} (X \Rightarrow Y) \Rightarrow X$

related functions $R \subseteq X \times Y$, $S \subseteq U \times V$ $f : X \longrightarrow U$, $g : Y \longrightarrow V$:

f and g are (R, S)–related if $\forall x \in X, y \in Y \,.\, x\, R\, y \Longrightarrow (f\, x)\, S\, (g\, y)$

Relation lifting for T $\mathrm{Rel}(T)(S, R) \subseteq T(U, X) \times T(V, Y)$

$$\mathrm{Rel}(T)(S, R) = \Big\{(a, b) \mid a : (X \Rightarrow U) \longrightarrow X, b : (Y \Rightarrow V) \longrightarrow Y \text{ such that for all } (R, S)\text{–related functions } f \text{ and } g : (a\, f)\, R\, (b\, g)\Big\}$$

T–bisimulation $R \subseteq X \times Y$, $c : X \longrightarrow (X \Rightarrow X) \Rightarrow X$, $d : Y \longrightarrow (Y \Rightarrow Y) \Rightarrow Y$

$$\forall x : X,\ y : Y \,.\, f : X \longrightarrow X, g : Y \longrightarrow Y \,.\, x\, R\, y \wedge f, g\ (R, R)\text{–related} \Longrightarrow (c\, x\, f)\, R\, (d\, y\, g)$$

Defining T–coalgebras $X \stackrel{\text{def}}{=} \{x_1, x_2, x_3, x_4\}$ $Y \stackrel{\text{def}}{=} \{y_1, y_2, y_3, y_4\}$

$$\begin{array}{lll} f & : & X \longrightarrow X \\ g & : & Y \longrightarrow Y \\ c & : & X \longrightarrow (X \Rightarrow X) \Rightarrow X \\ d & : & Y \longrightarrow (Y \Rightarrow Y) \Rightarrow Y \end{array} \qquad \begin{array}{lll} f\, x & = & \text{if } x = x_1 \text{ then } x_1 \text{ else } x_4 \text{ endif} \\ g\, y & = & \text{if } y = y_1 \text{ then } y_1 \text{ else } y_4 \text{ endif} \\ c\, x\, h & = & \text{if } h = f \text{ then } x_4 \text{ else } x_1 \text{ endif} \\ d\, y\, k & = & \text{if } k = g \text{ then } y_4 \text{ else } y_1 \text{ endif} \end{array}$$

- f and g are neither (R, R) nor (S, S) related $\Longrightarrow$ R and S are bisimulations
- f and g are $(R \cap S, R \cap S)$–related $\Longrightarrow$ $R \cap S$ is not a bisimulation

Let $\mathcal{C}$ be a bicartesian closed category (i.e., containing finite products, coproducts and exponents).

Polynomial functors $F : \mathcal{C} \longrightarrow \mathcal{C}$

$$F(X) \quad = \quad \begin{cases} A \\ X \\ F_1(X) \ \times \ F_2(X) \\ F_1(X) \ + \ F_2(X) \\ A \ \Rightarrow \ F_1(X) \end{cases}$$

Extended polynomial functors $G : \mathcal{C}^{op} \times \mathcal{C} \longrightarrow \mathcal{C}$

$$G(Y, X) \quad = \quad \begin{cases} A \\ X \\ G_1(Y, X) \ \times \ G_2(Y, X) \\ G_1(Y, X) \ + \ G_2(Y, X) \\ F(Y) \ \Rightarrow \ G_1(Y, X) \end{cases}$$

Let $\mathcal{C}$ be a bicartesian closed category (i.e., containing finite products, coproducts and exponents).

Polynomial functors $\quad F : \mathcal{C} \qquad \mathcal{C}$

$$F(X) \quad = \quad \begin{cases} A \\ X \\ F_1(X) \;\times\; F_2(X) \\ F_1(X) \;+\; F_2(X) \\ A \;\Rightarrow\; F_1(X) \end{cases}$$

Extended polynomial functors $\quad G : \mathcal{C}^{op} \times \mathcal{C} \qquad \mathcal{C}$

$$G(Y, X) \quad = \quad \begin{cases} A \\ X \\ G_1(Y, X) \;\times\; G_2(Y, X) \\ G_1(Y, X) \;+\; G_2(Y, X) \\ F(Y) \;\Rightarrow\; G_1(Y, X) \end{cases}$$

Let $\mathcal{C}$ be a bicartesian closed category (i.e., containing finite products, coproducts and exponents).

Cartesian functors $\quad K : \mathcal{C} \qquad \mathcal{C}$

$$K(X) \quad = \quad \begin{cases} A \\ X \\ K_1(X) \ \times \ K_2(X) \\ K_1(X) \ + \ K_2(X) \\ A \ \Rightarrow \ F_1(X) \end{cases}$$

Extended cartesian functors $\quad G : \mathcal{C}^{op} \times \mathcal{C} \qquad \mathcal{C}$

$$G(Y, X) \quad = \quad \begin{cases} A \\ X \\ G_1(Y, X) \ \times \ G_2(Y, X) \\ G_1(Y, X) \ + \ G_2(Y, X) \\ K(Y) \ \Rightarrow \ G_1(Y, X) \end{cases}$$

Lemma 5 *Assume* $(R_i \subseteq X \times Y)_{i \in I}$, $(S_i \subseteq U \times V)_{i \in I}$

$$\bigcap_i \mathrm{Rel}(G)(S_i, R_i) \quad \subseteq \quad \mathrm{Rel}(G)\big(\bigcap_i S_i, \bigcap_i R_i\big)$$

Proof

- for polynomial functors F: $\bigcap_i \mathrm{Rel}(F)(R_i) \quad = \quad \mathrm{Rel}(F)(\bigcap_i R_i)$ $(*)$
- main result by induction, for instance

$$\begin{aligned}
\bigcap_i \mathrm{Rel}(F \Rightarrow G_1)(S_i, R_i) = &\quad \bigcap_i \big(\mathrm{Rel}(F)(S_i) \Rightarrow_{\mathrm{R}} \mathrm{Rel}(G_1)(S_i, R_i)\big) \\
\subseteq &\quad \bigcap_i \mathrm{Rel}(F)(S_i) \Rightarrow_{\mathrm{R}} \bigcap_i \mathrm{Rel}(G_1)(S_i, R_i) \\
= &\quad \mathrm{Rel}(F)(\bigcap_i S_i) \Rightarrow_{\mathrm{R}} \bigcap_i \mathrm{Rel}(G_1)(S_i, R_i) && \text{by } (*) \\
\subseteq &\quad \mathrm{Rel}(F)(\bigcap_i S_i) \Rightarrow_{\mathrm{R}} \mathrm{Rel}(G_1)(\bigcap_i S_i, \bigcap_i R_i) && \text{by Ind.} \\
= &\quad \mathrm{Rel}(F \Rightarrow G_1)(\bigcap_i S_i, \bigcap_i R_i)
\end{aligned}$$

Proposition 6 *Bisimulations are closed under arbitrary nonempty intersections.*

Proof

- Assume two coalgebras $c : X \longrightarrow G(X, X)$, $d : Y \longrightarrow G(Y, Y)$ and a nonempty family $(R_i \subseteq X \times Y)_{i \in I}$ of bisimulations.
- $\forall i \in I \,.\quad (c\,x,\, d\,y) \;\in\; \mathrm{Rel}(G)(R_i, R_i)$
- $(c\,x,\, d\,y) \;\in\; \bigcap_i \mathrm{Rel}(G)(R_i, R_i)$
- Lemma 5: $(c\,x,\, d\,y) \in \mathrm{Rel}(G)\big(\bigcap_i R_i, \bigcap_i R_i\big)$

1. Assume relations $(S_i \subseteq U \times V)_{i \in I}$ and $(R_i \subseteq X \times Y)_{i \in I}$:

$$\begin{array}{rcl} \bigcap_i \left(\mathrm{Eq}(A) \Rightarrow_{\mathrm{R}} R_i\right) & = & \mathrm{Eq}(A) \Rightarrow_{\mathrm{R}} \bigcap_i R_i \\ \bigcap_i \left(S_i \Rightarrow_{\mathrm{R}} R_i\right) & \subseteq & \bigcap_i S_i \Rightarrow_{\mathrm{R}} \bigcap_i R_i \end{array}$$

2. Assume $S_1, S_2 \subseteq U \times V$ and $R_1, R_2 \subseteq X \times Y$ with $S_1 \subseteq S_2$ and $R_1 \subseteq R_2$:

$$S_2 \Rightarrow_{\mathrm{R}} R_1 \quad \subseteq \quad S_1 \Rightarrow_{\mathrm{R}} R_2$$

Results for extended polynomial functors

Let $G : \textbf{Sets}^{op} \times \textbf{Sets} \longrightarrow \textbf{Sets}$ be a extended polynomial functor.

Proposition 7 *Relation lifting for G is fibred, i.e. commutes with substitution.*

Proposition 8 *Bisimulations and invariants are closed under arbitrary nonempty intersections.*

Proposition 9 *Bisimulations are closed under composition.*

Proposition 10 *The image $\coprod_f P$ of a morphism with respect to an invariant P is an invariant.*

Proposition 11 *Invariants correspond to subcoalgebras and vice versa.*

Proposition 12 *The predicate $\coprod_{\pi_1} R$ is an invariant for any bisimulation R.*

Proposition 13 *The relation $\pi_1^* P \cap R$ is a bisimulation for any invariant P and any bisimulation R.*

Proposition 14 *The predicate P is an invariant if and only if $\coprod_\delta P$ is a bisimulation.*

Proposition 15 *A function is a G–coalgebra morphism if and only if its graph is a bisimulation.*

Proposition 16 *The kernel of a G–morphism is a bisimulation.*

Proposition 17 *The Aczel/Mendler approach and the Hermida/Jacobs approach yield identical notions of invariant and bisimulation.*

still missing

- union of bisimulations/invariants
- greatest bisimulations/invariants

Invariants according to Hermida & Jacobs

- P is an invariant for c, if

$$\forall x \in X \,.\, P(x) \quad \text{implies} \quad \mathrm{Pred}(H)(P, P)(c(x))$$

Strong Invariants

- P is a strong invariant for c, if

$$\forall x \in X \,.\, P(x) \quad \text{implies} \quad \mathrm{Pred}(H)(\top_X, P)(c(x))$$

Example method merge : **Self** $\times$ **Self**$\longrightarrow$**Self**, predicate $P \subseteq$ **Self**

invariant	$\forall x, y : \mathbf{Self} \,.\, P(x) \wedge P(y) \implies P(\mathsf{merge}(\mathsf{x}, \mathsf{y}))$
strong invariant	$\forall x, y : \mathbf{Self} \,.\, P(x) \implies P(\mathsf{merge}(\mathsf{x}, \mathsf{y}))$

Proposition 18 *For a family $(P_i \subseteq X)_{i \in I}$ of strong invariants both $\bigcap_i P_i$ and $\bigcup_i P_i$ are strong invariants.*

Example: Union of Bisimulations

Functor $K(Y, X) \stackrel{\text{def}}{=} Y \Rightarrow \mathsf{bool}$

***K*–bisimulation** $R \subseteq X \times X \quad c : X \longrightarrow X \Rightarrow \mathsf{bool}$

$$\forall x, x', y, y' : X \,.\, x\, R\, x' \;\wedge\; y\, R\, y' \;\Longrightarrow\; c\, x\, y \;=\; d\, x'\, y'$$

Defining a *K*–coalgebra $X \stackrel{\text{def}}{=} \{x_1, x_2\} \quad R \subseteq X \times X \stackrel{\text{def}}{=} \{(x_1, x_2), (x_2, x_1)\}$

$$c \;:\; X \longrightarrow X \Rightarrow \mathsf{bool} \qquad\qquad c\, x\, y \;=\; \mathsf{if}\ x = y\ \mathsf{then}\ \bot\ \mathsf{else}\ \top\ \mathsf{endif}$$

- $\mathrm{Eq}(X) = \{(x_1, x_1),\ (x_2, x_2)\}$ is a bisimulations for c
- R is a bisimulation for c
- However, $\mathrm{Eq}(X) \cup R = X \times X$ is not a bisimulation

$$\bot \quad = \quad c\, x_1\, x_1 \quad \neq \quad c\, x_1\, x_2 \quad = \quad \top$$

Consider $\mathsf{merge3} : \textbf{Self} \times \textbf{Self} \times \textbf{Self} \longrightarrow \textbf{Self}$

Three bisimulations $R_1, R_2, R_3 \subseteq \textbf{Self} \times \textbf{Self}$

with $x\, R_1\, x'$, $y\, R_2\, y'$ $z\, R_3\, z'$

Intuitive reasoning

$$\begin{array}{cc} & \mathsf{merge3}(x, y, z) \\ x\, R_1\, x' & \wr\| \\ & \mathsf{merge3}(x', y, z) \\ y\, R_2\, y' & \wr\| \\ & \mathsf{merge3}(x', y', z) \\ z\, R_3\, z' & \wr\| \\ & \mathsf{merge3}(x', y', z') \\ \hline \end{array}$$

$$x\, R_1\, x' \;\wedge\; y\, R_2\, y' \;\wedge\; z\, R_3\, z' \quad\Longrightarrow\quad \mathsf{merge3}(x, y, z) \;\cong\; \mathsf{merge3}(x', y', z')$$

Implicit assumptions

- $y\, R_1\, y'$, $z\, R_1\, z$, $x\, R_2\, x$, ...
- $\mathsf{dom}(R_1) \quad = \quad \mathsf{dom}(R_2) \quad = \quad \mathsf{dom}(R_3)$
- bisimulations are partial equivalence relations

Let $R \subseteq X \times X$ be a relation.

Domain of R: $\mathsf{dom}(R) = \{x \mid \exists y \,.\, x\,R\,y \vee y\,R\,x\}$

Partially reflexive $\equiv$ reflexive on domain

R is partially reflexive if $\forall x, y \,.\, x\,R\,y \implies x\,R\,x \wedge y\,R\,y$

Partial equivalence relation $\equiv$ symmetric and transitive

(i.e., an equivalence relation on its domain)

Least partial equivalence relation containing R is denoted with $\overline{R}$

Zigzag in R is a finite sequence

$$\begin{tikzcd}
x_0 \arrow[dr] & & x_2 \arrow[dr] & & \cdots & {} \arrow[dr] & & x_n \\
& x_1 \arrow[ur] & & {} & & & x_{n-1} \arrow[ur] &
\end{tikzcd}$$

$$x\,\overline{R}\,y \iff \exists \text{ a zigzag } x, \ldots, y \text{ in } R$$

Assumption: All relations are partial equivalence relations!

Lemma 19

$$\begin{array}{rcl} \overline{S +_{\mathrm{R}} R} & = & \overline{S} +_{\mathrm{R}} \overline{R} \\ \overline{S \times_{\mathrm{R}} R} & = & \overline{S} \times_{\mathrm{R}} \overline{R} \end{array}$$

Lemma 20

$$\begin{array}{rcl} \overline{\bigcup_i (S_i +_{\mathrm{R}} R_i)} & = & \overline{(\bigcup_i S_i) +_{\mathrm{R}} (\bigcup_i R_i)} \\ \overline{\bigcup_i (S_i \times_{\mathrm{R}} R_i)} & = & \overline{(\bigcup_i S_i) \times_{\mathrm{R}} (\bigcup_i R_i)} \end{array}$$

Lemma 21

$$\mathrm{Rel}(K)(\overline{\bigcup_i R_i}) \quad = \quad \overline{\bigcup_i \mathrm{Rel}(K)(R_i)}$$

(Another technical lemma about $\mathrm{Rel}(G)(\overline{\bigcup_i S_i}, \overline{\bigcup_i R_i})$ proved by induction on G.)

Theorem 22 *For a coalgebra of an extended cartesian functor the partial bisimulation equivalences on a fixed domain form a complete lattice. (The join of a family (R_i) of bisimulations is $\overline{\bigcup_i R_i}$.)*

Functor $G(Y, X) \stackrel{\mathsf{def}}{=} (\mathbb{N} \Rightarrow Y) \Rightarrow \mathsf{bool}$

Defining a G–coalgebra on $\mathbb{N}$

$$c \;:\; \mathbb{N} \longrightarrow (\mathbb{N} \Rightarrow \mathbb{N}) \Rightarrow \mathsf{bool} \qquad\qquad c\,x\,f \;=\; \begin{cases} \top & \exists n\,.\,\forall i\,.\,f\,i < n \\ \bot & \text{otherwise} \end{cases}$$

G–bisimulation for c $\quad R \subseteq \mathbb{N} \times \mathbb{N}$

$$\forall x, y : \mathbb{N},\; f, g : \mathbb{N} \longrightarrow \mathbb{N}\,.\, x\,R\,y \;\wedge\; \big(\forall n\,.\,(f\,n)\,R\,(g\,n)\big) \;\Longrightarrow\; c\,x\,f \;=\; c\,y\,g$$

Family $(S_n)_{n \in \mathbb{N}}$

$$i\,S_n\,j \quad\Longleftrightarrow\quad i = j \;\vee\; (i < n \wedge j < n)$$

- all S_n are bisimulations
- every *finite* subset of S_n's has an upper bound (which is a bisimulation)
- $\bigcup_n S_n \;=\; \mathbb{N} \times \mathbb{N}$ is *not* a bisimulation, take $f_0 : \mathbb{N} \longrightarrow \mathbb{N}$ with $f_0\,n = 0$ and $\mathsf{id}_{\mathbb{N}}$: $\forall n\,.\,(n, 0) \in \bigcup_n S_n$ but

$$\top \;=\; c\,0\,\mathsf{id}_{\mathbb{N}} \;\neq\; c\,0\,f_0 \;=\; \bot$$

Problem $\quad (f_0, \mathsf{id}_{\mathbb{N}}) \in \left(\mathrm{Eq}(\mathbb{N}) \Rightarrow_{\mathrm{R}} \overline{\bigcup_n S_n}\right)$ but $(f_0, \mathsf{id}_{\mathbb{N}}) \notin \overline{\bigcup_n (\mathrm{Eq}(\mathbb{N}) \Rightarrow_{\mathrm{R}} S_n)}$

Observation $\quad$ if $\{a \mid f\,a \neq f\,b\}$ is finite $\quad$ + some other assumptions

$$(f, g) \in \left(\mathrm{Eq}(A) \Rightarrow_{\mathrm{R}} \overline{\bigcup_i R_i}\right) \quad \Longrightarrow \quad (f, g) \in \overline{\bigcup_i (\mathrm{Eq}(A) \Rightarrow_{\mathrm{R}} R_i)}$$

$\boldsymbol{d : X \longrightarrow (\mathbb{N} \Rightarrow X) \Rightarrow}$ **bool is finitely based** $\quad$ if $\forall x : X, f : \mathbb{N} \longrightarrow X$

$$\exists A \subseteq \mathbb{N} \,.\, A \text{ is finite } \wedge \forall g : \mathbb{N} \longrightarrow X \,.\, f|_A = g|_A \Longrightarrow d\,x\,f = d\,x\,g$$

Theorem 23 *For a finitely based coalgebra of an extended polynomial functor the partial bisimulation equivalences on a fixed domain form a complete lattice.*

Proof idea

- consider $(f, g) \in \left(\mathrm{Eq}(A) \Rightarrow_{\mathrm{R}} \overline{\bigcup_i R_i}\right)$
- d is finitely based $\Longrightarrow \exists A \subseteq \mathbb{N} \ldots$
- from f construct f' such that $f|_A = f'|_A$ and $f'|_{\mathbb{N}\setminus A} = g|_{\mathbb{N}\setminus A}$
- $d\,x\,f = d\,x\,f'$ and $(f', g) \in \overline{\bigcup_i (\mathrm{Eq}(A) \Rightarrow_{\mathrm{R}} R_i)}$

Example: Two place Buffer

```
Begin Buf[A : Type] : ClassSpec
  Method
    put : Self × A + 1 -> Self;
    get : Self -> A + 1;

  Assertion SelfVar x : Self
    get_put : Forall(a : A) . get(put(x, up a)) = up a;

  attribute
    next_buffer : Self -> Addr;

  method
    push : Self × [Addr -> Self + 1] -> Self × [Addr -> Self + 1];

  assertion SelfVar x : Self
    push_x : Forall( buf_mem : [Addr -> Self + 1] ) :
      proj_1 (push(x, buf_mem)) = x;

    push_mem : Forall( buf_mem : [Addr -> Self + 1], a : Addr ) :
      proj_2 (push(x, buf_mem)) a =
        if a = next_buffer(x) and up?(buf_mem a)
        then up(put(down(buf_mem a), get x))
        else bot;
end Buf
```

Extension

higher-order	only trivial properties
extended polynomial	structural properties + greatest bisimulations equivalences for finitely based coalgebras
extended cartesian	structural properties + greatest bisimulations equivalences for all coalgebras
polynomial	structural properties + greatest bisimulations + final models

Results/Examples have been checked/developed with PVS.
see `http://wwwtcs.inf.tu-dresden.de/~tews/binary/`

Built into the Coalgebraic Class Specification Language CCSL [RTJ01].
see `http://wwwtcs.inf.tu-dresden.de/~tews/ccsl/`

[HJ98] C. Hermida and B. Jacobs. Structural induction and coinduction in a fibrational setting. *Inf. & Comp.*, pages 107–152, 1998.

[HK99] R. Hennicker and A. Kurz. (Ω, Ξ)–Logic: On the algebraic extension of coalgebraic specifications. In B. Jacobs and J. Rutten, editors, *Coalgebraic Methods in Computer Science '99*, volume 19 of *ENTCS*, pages 195–212. Elsevier, 1999.

[Jac96] B. Jacobs. Objects and classes, co–algebraically. In B. Freitag, C.B. Jones, C. Lengauer, and H.-J. Schek, editors, *Object–Orientation with Parallelism and Peristence*, pages 83–103. Kluwer Acad. Publ., 1996.

[Rei95] H. Reichel. An approach to object semantics based on terminal co–algebras. *Mathematical Structure in Computer Science*, 5:129–152, 1995.

[Roş00] Grigore Roşu. *Hidden Logic*. PhD thesis, University of California at San Diego, 2000.

[RTJ01] J. Rothe, H. Tews, and B. Jacobs. The coalgebraic class specification language CCSL. *Journal of Universal Computer Science*, 7(2):175–193, March 2001.

[Tew01] H. Tews. Coalgebras for binary methods: Properties of bisimulations and invariants. *Theoretical informatics and applications*, 35(1):83–111, February 2001.

[Tew02] H. Tews. Greatest bisimulations for binary methods. Submitted, 2002.

Observation

can allow the functor List : $\mathbf{Set} \longrightarrow \mathbf{Set}$

Further work

- finite powerset functor
- extend to iterated functors