How to write a coequation.

Fredrik Dahlqvist

Joint work with Todd Schmid

Comonadic meeting - 22 February 2023

Starting point

Starting point

- Coequations have been around for at least 25 years

Starting point

- Coequations have been around for at least 25 years
- Tons of theoretical results

Starting point

- Coequations have been around for at least 25 years
- Tons of theoretical results
- ... but coequations haven't really been adopted as a practical formalism by computer scientists

Starting point

- Coequations have been around for at least 25 years
- Tons of theoretical results
- ... but coequations haven't really been adopted as a practical formalism by computer scientists
- Why?

Some answers

Some answers

- No universally accepted syntax to write a coequation

Some answers

- No universally accepted syntax to write a coequation
- Difficult for the end-user to understand what a coequation is

Some answers

- No universally accepted syntax to write a coequation
- Difficult for the end-user to understand what a coequation is
- Which formalism should be used in practice?

Root cause

Root cause

- This problem is inevitable

Root cause

- This problem is inevitable
- Equations are given by pairs of terms

Root cause

- This problem is inevitable
- Equations are given by pairs of terms
  - Terms are *finite* trees

Root cause

- This problem is inevitable
- Equations are given by pairs of terms
  - Terms are *finite* trees
  - Using brackets there is an unambiguous finite string representation

Root cause

- This problem is inevitable
- Equations are given by pairs of terms
  - Terms are *finite* trees
  - Using brackets there is an unambiguous finite string representation
- Coequations typically deal with generalised trees

Root cause

- This problem is inevitable
- Equations are given by pairs of terms
  - Terms are *finite* trees
  - Using brackets there is an unambiguous finite string representation
- Coequations typically deal with generalised trees
  - Infinitely branching, infinite depth

Root cause

- This problem is inevitable
- Equations are given by pairs of terms
  - Terms are *finite* trees
  - Using brackets there is an unambiguous finite string representation
- Coequations typically deal with generalised trees
  - Infinitely branching, infinite depth
  - No finite string representation

Root cause

- This problem is inevitable
- Equations are given by pairs of terms
  - Terms are *finite* trees
  - Using brackets there is an unambiguous finite string representation
- Coequations typically deal with generalised trees
  - Infinitely branching, infinite depth
  - No finite string representation
  - Often equivalence classes of trees

Root cause

- This problem is inevitable
- Equations are given by pairs of terms
  - Terms are *finite* trees
  - Using brackets there is an unambiguous finite string representation
- Coequations typically deal with generalised trees
  - Infinitely branching, infinite depth
  - No finite string representation
  - Often equivalence classes of trees
- Impossible to get a simple syntax working well in every case

Root cause

- This problem is inevitable
- Equations are given by pairs of terms
  - Terms are *finite* trees
  - Using brackets there is an unambiguous finite string representation
- Coequations typically deal with generalised trees
  - Infinitely branching, infinite depth
  - No finite string representation
  - Often equivalence classes of trees
- Impossible to get a simple syntax working well in every case
- Underlying maths seems harder to grasp (relation versus *corelation*)

Outline

Outline

- History of the notion of coequation

Outline

- History of the notion of coequation
- From this extract 4 kinds of syntax

Outline

- History of the notion of coequation
- From this extract 4 kinds of syntax
  - Coequation-as-corelation
  - Coequation-as-predicate
  - Coequation-as-equation
  - Coequation-as-modal-formula

History of coequations

1970s → 1980s → 1990s → 2000s → Future

1970s

Appearance of the word *'coequational'*, Davis.
Focused on finding examples of comonads/cotriples.

1980s

Concept of *'covariety'* (1985).
Aczel – Non-wellfounded sets terminal coalgebras.

1990s

Early 90s: terminal coalgebra semantics, coalgebraic specification (hd(tl(x))=hd(x))
Jacobs, Reichel, Hensel.
Coequations-as-equations.
Late 90's: birth of coalgebra community.
Rutten & Gumm papers, CMCS 98.
Coequations-as-predicate.
Word '*coequation*' – Cîrstea 1999.

2000s

Early 00s: Wolter, Kurtz – coequations as formal dual to equations.
Coequations-as-corelations.
Gumm, Adámek – coequations as pattern avoidance. Coequational logic.
Mid 00s: Birth of coalgebraic modal logic.
Coequations-as-modal-formulas.

Future

???

Coequation-as-corelation

Equations

Equations

Equations are *relations* under which one can take a *quotient*

$$\begin{tikzcd} Rel \arrow[r, shift left, "e_1"] \arrow[r, shift right, "e_2"'] & F_T Var \arrow[r, two heads] \arrow[d, "v"] & Q \arrow[dl, dashed] \\ & A & \end{tikzcd}$$

Equations

Equations are *relations* under which one can take a *quotient*

$$\begin{tikzcd} Rel \arrow[r, shift left, "e_1"] \arrow[r, shift right, "e_2"'] & F_T\,Var \arrow[r, two heads] \arrow[d, "v"] & Q \arrow[dl, dashed] \\ & A & \end{tikzcd}$$

Example: semigroups

$TX = X \times X$, $Var = \{x, y, z\}$, $Rel = 1$, $e_1(*) = (xy)z$, $e_2(*) = x(yz)$

$$\begin{tikzcd} 1 \arrow[r, shift left, "e_1"] \arrow[r, shift right, "e_2"'] & F_T\{x, y, z\} \arrow[r, two heads] \arrow[d, "v"] & Q \arrow[dl, dashed] \\ & A & \end{tikzcd}$$

Coequations

Coequations

Dually, coequations are *corelations* defining a subobject

$$\begin{tikzcd} S \arrow[r, tail] & C_T Col \arrow[r, shift left, "c_1"] \arrow[r, shift right, "c_2"'] & CoRel \\ & W \arrow[u, "c"'] \arrow[ul, dashed] & \end{tikzcd}$$

Coequations

Dually, coequations are *corelations* defining a subobject

$$\begin{tikzcd} S \arrow[r, tail] & C_T Col \arrow[r, shift left, "c_1"] \arrow[r, shift right, "c_2"'] & CoRel \\ & W \arrow[u, "c"'] \arrow[ul, dashed] & \end{tikzcd}$$

Example: deterministic binary trees $TX = X \times X$, $Col = \{b, w\}$, $CoRel = 2$, $c_1(t) = 1$ if $\mathrm{Left}(t) = b$, $c_2(t) = 1$ if $\mathrm{Right}(t) = b$

$$\begin{tikzcd} S \arrow[r, tail] & C_T\{b, w\} \arrow[r, shift left, "c_1"] \arrow[r, shift right, "c_2"'] & 2 \\ & W \arrow[u, "c"'] \arrow[ul, dashed] & \end{tikzcd}$$

Birkhoff's HSC theorem

Theorem

Let T : $\mathbf{Set} \to \mathbf{Set}$ be a covarietor. A class of T-coalgebras is a covariety iff it is closed under Homomorphic images (H), Subcoalgebras (S) and Coproducts (C).

Birkhoff's HSC theorem

Theorem

Let $T : \mathbf{Set} \to \mathbf{Set}$ *be a covarietor. A class of* T*-coalgebras is a covariety iff it is closed under Homomorphic images (H), Subcoalgebras (S) and Coproducts (C).*

Note: the HSC conditions make sense even if T is not a covarietor – for example $T = \mathcal{P}$. We will call a class of T-coalgebras a *structural covariety* if it is closed under HSC.

Coequation-as-predicate

Coequations-as-predicate: behavioural coequations

Coequations-as-predicate: behavioural coequations

- Let $T : \mathbf{Set} \to \mathbf{Set}$ have a terminal coalgebra νT

Coequations-as-predicate: behavioural coequations

- Let $T : \mathbf{Set} \to \mathbf{Set}$ have a terminal coalgebra νT
- A *behavioural coequation* is a subset $W \subseteq \nu T$

Coequations-as-predicate: behavioural coequations

- Let $T : \mathbf{Set} \to \mathbf{Set}$ have a terminal coalgebra νT
- A *behavioural coequation* is a subset $W \subseteq \nu T$
- A T-coalgebra (X, γ) satisfies W if

$$\mathrm{im}!_{(X,\gamma)} \subseteq W$$

In other words: every behaviour in (X, γ) belongs to W.

Coequations-as-predicate: behavioural coequations

- Let $T : \mathbf{Set} \to \mathbf{Set}$ have a terminal coalgebra νT
- A *behavioural coequation* is a subset $W \subseteq \nu T$
- A T-coalgebra (X, γ) satisfies W if

$$\mathrm{im}!_{(X,\gamma)} \subseteq W$$

  In other words: every behaviour in (X, γ) belongs to W.
- No particular syntax, any way of describing W will do

Coequations-as-predicate: behavioural coequations

Theorem (Rutten, 1996)

The class of T-coalgebras satisfying a behavioural coequation forms a structural covariety.

Coequations-as-predicate: behavioural coequations

Theorem (Rutten, 1996)

The class of T-coalgebras satisfying a behavioural coequation forms a structural covariety.

But not all structural covarieties are behavioural, e.g. locally finite automata.

Coequations-as-predicate: behavioural coequations

Theorem (Rutten, 1996)

The class of T-coalgebras satisfying a behavioural coequation forms a structural covariety.

But not all structural covarieties are behavioural, e.g. locally finite automata.

Theorem (Gumm-Schröder, 1998)

Let T preserve weak pullbacks. Then a structural T-covariety is behavioural if and only if it is closed under total bisimulations.

Coequation-as-predicate: beyond behaviour

- To capture more classes of coalgebras we need to consider 'labelled/coloured' coalgebras. Let $T : \mathbf{Set} \to \mathbf{Set}$ be a covarietor and let Col be a set of colors.

Coequation-as-predicate: beyond behaviour

- To capture more classes of coalgebras we need to consider 'labelled/coloured' coalgebras. Let $T : \mathbf{Set} \to \mathbf{Set}$ be a covarietor and let Col be a set of colors.
- A coequation-as-predicate can be either
  - a subcoalgebra $Coeq \rightarrowtail C_T Col$, or
  - a subset $Coeq \rightarrowtail U_T C_T Col$

Coequation-as-predicate: beyond behaviour

- To capture more classes of coalgebras we need to consider 'labelled/coloured' coalgebras. Let $T : \mathbf{Set} \to \mathbf{Set}$ be a covarietor and let Col be a set of colors.
- A coequation-as-predicate can be either
  - a subcoalgebra $Coeq \rightarrowtail C_T Col$, or
  - a subset $Coeq \rightarrowtail U_T C_T Col$
- A coalgebra (X, γ) satisfies $Coeq$ if for every colouring map $c : X \to Col$

$$\mathrm{im}\hat{c} \subseteq Coeq$$

Coequation-as-predicate: beyond behaviour

- To capture more classes of coalgebras we need to consider 'labelled/coloured' coalgebras. Let $T : \mathbf{Set} \to \mathbf{Set}$ be a covarietor and let *Col* be a set of colors.
- A coequation-as-predicate can be either
  - a subcoalgebra $\mathit{Coeq} \rightarrowtail C_T \mathit{Col}$, or
  - a subset $\mathit{Coeq} \rightarrowtail U_T C_T \mathit{Col}$
- A coalgebra (X, γ) satisfies *Coeq* if for every colouring map $c : X \to \mathit{Col}$

$$\mathrm{im}\hat{c} \subseteq \mathit{Coeq}$$

- No syntax, any way of describing a subcoalgebra/subset will do.

Coequation-as-predicate: beyond behaviour

- To capture more classes of coalgebras we need to consider 'labelled/coloured' coalgebras. Let $T : \mathbf{Set} \to \mathbf{Set}$ be a covarietor and let *Col* be a set of colors.
- A coequation-as-predicate can be either
  - a subcoalgebra $Coeq \rightarrowtail C_T Col$, or
  - a subset $Coeq \rightarrowtail U_T C_T Col$
- A coalgebra (X, γ) satisfies *Coeq* if for every colouring map $c : X \to Col$

$$\mathrm{im}\hat{c} \subseteq Coeq$$

- No syntax, any way of describing a subcoalgebra/subset will do.
- Special syntax for pattern avoidance in coalgebras for polynomial functors (Gumm, Adámek *et al*): $\boxtimes t$, $t \in C_T Col$

Coequation-as-predicate: beyond behaviour

Theorem (Rutten, 1996)

If T is κ-bounded and C is a structural T-covariety, then C is the class of coalgebras satisfying some coequation-as-predicate $Coeq \subseteq C_T\kappa$

Coequation-as-predicate: beyond behaviour

Theorem (Rutten, 1996)

If T is κ-bounded and C is a structural T-covariety, then C is the class of coalgebras satisfying some coequation-as-predicate $Coeq \subseteq C_T\kappa$

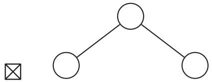
Theorem (Adámek, 2005)

Suppose T is a covarietor that preserves weak pullbacks. A T-covariety is presentable by a predicate coequation in κ colours if and only if it is closed under κ-colour bisimilarity.

$$\begin{tikzcd}
 & (R,\rho) \arrow[dl, dashed] \arrow[dr, dashed] & \\
(X,\gamma) \arrow[d, "c"'] & & (X',\gamma') \arrow[d, dashed, "c'"] \\
\kappa & & \kappa
\end{tikzcd}$$

Examples

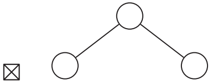
1. For $TX = X \times X + 1$



defines the covariety of binary trees which do *not* have two halting successors.

Examples

1. For $TX = X \times X + 1$

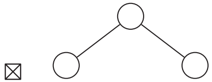


   defines the covariety of binary trees which do *not* have two halting successors.

2. A T-coalgebra (V, γ) is *locally finite* if for every $v \in V$ there exists a finite subcoalgebra S of (V, γ) such that $v \in S$. The class of locally finite T-coalgebra is a covariety. By a theorems from Rutten and Adamek there must exist a coequation in ω-colours describing it.

Examples

1. For $TX = X \times X + 1$



defines the covariety of binary trees which do *not* have two halting successors.

2. A T-coalgebra (V, γ) is *locally finite* if for every $v \in V$ there exists a finite subcoalgebra S of (V, γ) such that $v \in S$. The class of locally finite T-coalgebra is a covariety. By a theorems from Rutten and Adamek there must exist a coequation in ω-colours describing it.

3. The *filter functor* is not a covarietor. A generalized notion of coequation must be used. The class of topological spaces and open maps is a covariety in the class of coalgebras for the filter functor. Kurz and Rosicky present this covariety by a generalized coequation.

Coequation-as-equation

Coequation-as-equation

Coequation-as-equation

- *Specific syntax* to write certain coequations

Coequation-as-equation

- *Specific syntax* to write certain coequations
- Destructor signature: $\sigma : S \times X \to T(X)$
  Example: Bank account

$$\mathrm{bal} : X \to \mathbb{N} \qquad \mathrm{credit} : X \times \mathbb{N} \to X$$

Coequation-as-equation

- *Specific syntax* to write certain coequations
- Destructor signature: $\sigma : S \times X \to T(X)$
  Example: Bank account

$$\mathrm{bal} : X \to \mathbb{N} \qquad \mathrm{credit} : X \times \mathbb{N} \to X$$

- Build a grammar of terms from variables, signature and anything useful

$$x : X, n : \mathbb{N} \qquad (-) + (-) : \mathbb{N} \times \mathbb{N} \to \mathbb{N}$$

Coequation-as-equation

- *Specific syntax* to write certain coequations
- Destructor signature: $\sigma : S \times X \to T(X)$
  Example: Bank account

$$\mathrm{bal} : X \to \mathbb{N} \qquad \mathrm{credit} : X \times \mathbb{N} \to X$$

- Build a grammar of terms from variables, signature and anything useful

$$x : X, n : \mathbb{N} \qquad (-) + (-) : \mathbb{N} \times \mathbb{N} \to \mathbb{N}$$

- Write specifications in the usual equational format

$$\mathrm{bal}(x) + n = \mathrm{bal}(\mathrm{credit}(n, x))$$

Coequations-as-equation are coequations-as-corelation

Coequations-as-equation are coequations-as-corelation

- Format of destructor signatures guarantee that currying is possible
  Taking products, bank account signature becomes

$$X \to \mathbb{N} \times X^{\mathbb{N}}$$

  i.e. a particular bank account instance is a coalgebra for $TX = \mathbb{N} \times X^{\mathbb{N}}$

Coequations-as-equation are coequations-as-corelation

- Format of destructor signatures guarantee that currying is possible
  Taking products, bank account signature becomes

  $$X \to \mathbb{N} \times X^{\mathbb{N}}$$

  i.e. a particular bank account instance is a coalgebra for $TX = \mathbb{N} \times X^{\mathbb{N}}$
- Format of terms guarantee that they can be interpreted as maps on X

  $$X \to \mathbb{N}^{\mathbb{N}}, x \mapsto \lambda n.[\![\mathrm{bal}(x) + n]\!]$$

Coequations-as-equation are coequations-as-corelation

- Format of destructor signatures guarantee that currying is possible
  Taking products, bank account signature becomes

  $$X \to \mathbb{N} \times X^{\mathbb{N}}$$

  i.e. a particular bank account instance is a coalgebra for $TX = \mathbb{N} \times X^{\mathbb{N}}$
- Format of terms guarantee that they can be interpreted as maps on X

  $$X \to \mathbb{N}^{\mathbb{N}}, x \mapsto \lambda n.[\![\mathrm{bal}(x) + n]\!]$$

- We get a coequation-as-corelation

  $$\nu T \rightrightarrows \mathbb{N}^{\mathbb{N}}$$

Coequations-as-equation are coequations-as-corelation

- Format of destructor signatures guarantee that currying is possible
  Taking products, bank account signature becomes

  $$X \to \mathbb{N} \times X^{\mathbb{N}}$$

  i.e. a particular bank account instance is a coalgebra for $TX = \mathbb{N} \times X^{\mathbb{N}}$
- Format of terms guarantee that they can be interpreted as maps on X

  $$X \to \mathbb{N}^{\mathbb{N}}, x \mapsto \lambda n. [\![\mathrm{bal}(x) + n]\!]$$

- We get a coequation-as-corelation

  $$\nu T \rightrightarrows \mathbb{N}^{\mathbb{N}}$$

- Classify behaviours according to what the functions $\lambda n. [\![\mathrm{bal}(x) + n]\!]$ and $\lambda n. [\![\mathrm{bal}(\mathrm{credit}(n, x))]\!]$ do, then *select* those for which the classifications match up

Coequation-as-modal-formula

Coequations-as-modal-formulas

Coequations-as-modal-formulas

- Syntax for coequation-as-predicate, given by *coalgebraic modal logic*

Coequations-as-modal-formulas

- Syntax for coequation-as-predicate, given by *coalgebraic modal logic*
- Idea: for a covarietor T and a set of atomic proposition $\mathtt{At}$ consider

$$C_T \mathcal{P} \mathtt{At}$$

Coequations-as-modal-formulas

- Syntax for coequation-as-predicate, given by *coalgebraic modal logic*
- Idea: for a covarietor T and a set of atomic proposition $\mathtt{At}$ consider

$$C_T \mathcal{P}\mathtt{At}$$

- Under very general assumptions, coalgebraic modal formula ϕ for T-systems can be canonically interpreted in this coalgebra

Coequations-as-modal-formulas

- Syntax for coequation-as-predicate, given by *coalgebraic modal logic*
- Idea: for a covarietor T and a set of atomic proposition $\mathtt{At}$ consider

$$C_T\mathcal{P}\mathtt{At}$$

- Under very general assumptions, coalgebraic modal formula ϕ for T-systems can be canonically interpreted in this coalgebra
- Formula ϕ defines the coequation-as-predicate

$$\{x \in C_T\mathcal{P}\mathtt{At} : x \models \phi\}$$

Coequations-as-modal-formulas

- Syntax for coequation-as-predicate, given by *coalgebraic modal logic*
- Idea: for a covarietor T and a set of atomic proposition $\mathtt{At}$ consider

$$C_T \mathcal{P}\mathtt{At}$$

- Under very general assumptions, coalgebraic modal formula ϕ for T-systems can be canonically interpreted in this coalgebra
- Formula ϕ defines the coequation-as-predicate

$$\{x \in C_T \mathcal{P}\mathtt{At} : x \models \phi\}$$

- Looks at local behaviour (typically $\sim$1,2 steps ahead), in practice $|\mathtt{At}| \leqslant 3$ is often enough

Coequations-as-modal-formulas

- Syntax for coequation-as-predicate, given by *coalgebraic modal logic*
- Idea: for a covarietor T and a set of atomic proposition $\mathtt{At}$ consider

$$C_T \mathcal{P}\mathtt{At}$$

- Under very general assumptions, coalgebraic modal formula ϕ for T-systems can be canonically interpreted in this coalgebra
- Formula ϕ defines the coequation-as-predicate

$$\{x \in C_T \mathcal{P}\mathtt{At} : x \models \phi\}$$

- Looks at local behaviour (typically $\sim$1,2 steps ahead), in practice $|\mathtt{At}| \leqslant 3$ is often enough
- Coalgebraic Goldblatt-Thomason theorem

Which coequation is right for you?

Which coequation is right for you?

- Is the behaviour you’re trying to define local?

Which coequation is right for you?

- Is the behaviour you're trying to define local?
  - Yes.
    How are you trying to define it?

Which coequation is right for you?

- Is the behaviour you're trying to define local?
  - Yes.
    How are you trying to define it?
    - Forbidden behaviour: coequation-as-predicate in $\boxtimes t$ format

Which coequation is right for you?

- Is the behaviour you're trying to define local?
  - Yes.
    How are you trying to define it?
    - Forbidden behaviour: coequation-as-predicate in $\boxtimes t$ format
    - Desired behaviour: coequation-as-modal formula

Which coequation is right for you?

- Is the behaviour you're trying to define local?
  - Yes.
    How are you trying to define it?
    - Forbidden behaviour: coequation-as-predicate in $\boxtimes t$ format
    - Desired behaviour: coequation-as-modal formula
    - Identifying behaviours/processes: coequation-as-equation/corelation

Which coequation is right for you?

- Is the behaviour you're trying to define local?
  - Yes.
    How are you trying to define it?
    - Forbidden behaviour: coequation-as-predicate in $\boxtimes t$ format
    - Desired behaviour: coequation-as-modal formula
    - Identifying behaviours/processes: coequation-as-equation/corelation
  - No
    How are you trying to define it?

Which coequation is right for you?

- Is the behaviour you're trying to define local?
  - Yes.
    How are you trying to define it?
    - Forbidden behaviour: coequation-as-predicate in $\boxtimes t$ format
    - Desired behaviour: coequation-as-modal formula
    - Identifying behaviours/processes: coequation-as-equation/corelation
  - No
    How are you trying to define it?
    - Desired behaviour: coequation-as-predicate $\{t : \phi(t)\}$

Which coequation is right for you?

- Is the behaviour you're trying to define local?
  - Yes.
    How are you trying to define it?
    - Forbidden behaviour: coequation-as-predicate in $\boxtimes t$ format
    - Desired behaviour: coequation-as-modal formula
    - Identifying behaviours/processes: coequation-as-equation/corelation
  - No
    How are you trying to define it?
    - Desired behaviour: coequation-as-predicate $\{t : \phi(t)\}$
    - Identifying behaviours: coequation-as-corelation

Which coequation is right for you?

- Is the behaviour you're trying to define local?
  - Yes.
    How are you trying to define it?
    - Forbidden behaviour: coequation-as-predicate in $\boxtimes t$ format
    - Desired behaviour: coequation-as-modal formula
    - Identifying behaviours/processes: coequation-as-equation/corelation
  - No
    How are you trying to define it?
    - Desired behaviour: coequation-as-predicate $\{t : \phi(t)\}$
    - Identifying behaviours: coequation-as-corelation
    - Not sure: Reason directly in terms of covariety?

Thank you.

Reference:

Fredrik Dahlqvist and Todd Schmid. “How to Write a Coequation.” *9th Conference on Algebra and Coalgebra in Computer Science.* 2021.