How to write a coequation

Fredrik Dahlqvist and Todd Schmidt

University College London

Salzburg via London, 3 September 2021

How to write a coequation

Fredrik Dahlqvist and Todd Schmidt

University College London

Salzburg via London, 3 September 2021

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?

Central premise

Central premise

- No universally accepted syntax to write a coequation

Central premise

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

Central premise

- 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 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 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 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 with generalised trees
  - Infinitely branching, infinite depth
  - No finite string representation
- Impossible to get a simple syntax working well in every case

Outline of the paper

Outline of the paper

- History of the notion of coequation

Outline of the paper

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

Outline of the paper

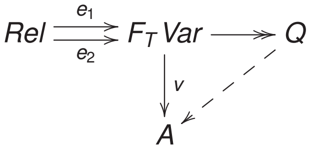
- History of the notion of coequation
- From this extract 4 kinds of syntax
  - Coequation-as-corelation
  - Coequation-as-predicate
  - Coequation-as-equation
  - Coequation-as-modal-formula

Coequation-as-corelation

Equations

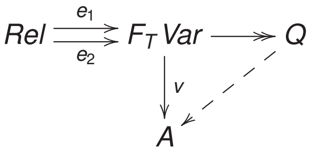
Equations

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



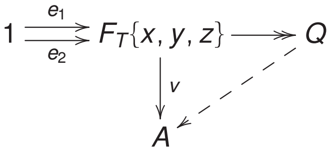
Equations

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



Example: semigroups

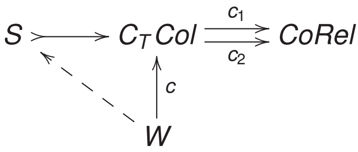
$TX = X \times X$, $Var = \{x, y, z\}$, $Rel = 1$, $e_1(*) = (xy)z$, $e_2(*) = x(yz)$



Coequations

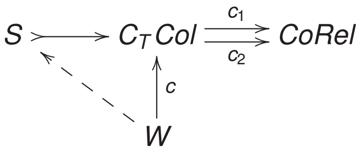
Coequations

Dually, coequations are *corelations* defining a subobject

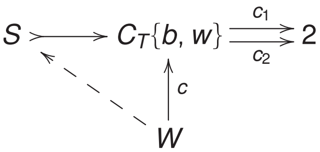


Coequations

Dually, coequations are *corelations* defining a subobject



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



Coequation-as-predicate

Coequation-as-predicate

Coequation-as-predicate

Two flavours: for a covarietor T, a coequation-as-predicate can be

- A subcoalgebra $Coeq \rightarrowtail C_T Col$
- A subset $Coeq \rightarrowtail U_T C_T Col$

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

Coequation-as-predicate

Two flavours: for a covarietor T, a coequation-as-predicate can be

- A subcoalgebra $Coeq \rightarrowtail C_T Col$
- A subset $Coeq \rightarrowtail U_T C_T Col$

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

Coequation-as-predicate

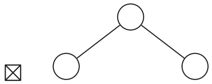
Two flavours: for a covarietor T, a coequation-as-predicate can be

- A subcoalgebra $Coeq \rightarrowtail C_T Col$
- A subset $Coeq \rightarrowtail U_T C_T Col$

No particular syntax, any way of describing a subcoalgebra/subset will do.
Special syntax for pattern avoidance (Gumm, Adamek and friends): $\boxtimes t$

Examples

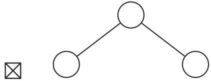
1. For $TX = X \times X + 1$



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

Examples

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

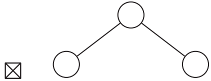


defines the covariety of binary trees which do *not* have two halting successors.

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 equations also guarantee that currying is possible

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

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 equations also guarantee that currying is possible

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

- We get a coequation-as-corelation

  $$C_T 1 \rightrightarrows \mathbb{N}^{\mathbb{N}}$$

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 equations also guarantee that currying is possible

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

- We get a coequation-as-corelation

  $$C_T 1 \rightrightarrows \mathbb{N}^{\mathbb{N}}$$

- 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-formula are coequations-as-predicate

Coequations-as-modal-formula are coequations-as-predicate

- *Specific syntax* for coequation-as-predicate

Coequations-as-modal-formula are coequations-as-predicate

- *Specific syntax* for coequation-as-predicate
- Idea: for a covarietor T and a set of atomic proposition $\mathtt{At}$ consider

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

Coequations-as-modal-formula are coequations-as-predicate

- *Specific syntax* for coequation-as-predicate
- Idea: for a covarietor T and a set of atomic proposition $\mathtt{At}$ consider

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

- A coalgebraic modal formulas for T-systems can be canonically interpreted in this coalgebra

Coequations-as-modal-formula are coequations-as-predicate

- *Specific syntax* for coequation-as-predicate
- Idea: for a covarietor T and a set of atomic proposition $\mathtt{At}$ consider

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

- A coalgebraic modal formulas for T-systems can be canonically interpreted in this coalgebra
- Picks the coequation-as-predicate

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

Coequations-as-modal-formula are coequations-as-predicate

- *Specific syntax* for coequation-as-predicate
- Idea: for a covarietor T and a set of atomic proposition $\mathtt{At}$ consider

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

- A coalgebraic modal formulas for T-systems can be canonically interpreted in this coalgebra
- Picks the coequation-as-predicate

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

- Looks at local behaviour (typically ∼1,2 steps ahead), uncountably many colours

Coequations-as-modal-formula are coequations-as-predicate

- *Specific syntax* for coequation-as-predicate
- Idea: for a covarietor T and a set of atomic proposition $\mathtt{At}$ consider

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

- A coalgebraic modal formulas for T-systems can be canonically interpreted in this coalgebra
- Picks the coequation-as-predicate

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

- Looks at local behaviour (typically ∼1,2 steps ahead), uncountably many colours
- 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.