Duality of Equations and Coequations via Contravariant Adjunctions

Julian Salamanca (CWI Amsterdam)
joint work with Marcello Bonsangue, and Jurriaan Rot.

April 2, 2016

CMCS 2016

Outline

- Equations and coequations.
  - Free algebras.
  - Cofree coalgebras.
- Contravariant adjuntions.
  - Lifting contravariant adjunctions.
  - Dualities between equations and coequations.
  - Liftings to Eilenberg–Moore categories.
- Applications.
  - Dynamical systems.
  - Deterministic automata.

Free L-algebras

$\mathfrak{F}(S) = (\mathfrak{F}(S), \tau)$: free L–algebra on $S \in \mathcal{D}$ generators.

Free L-algebras

$\mathfrak{F}(S) = (\mathfrak{F}(S), \tau)$: free L–algebra on $S \in \mathcal{D}$ generators.

$$
\begin{tikzcd}
L\mathfrak{F}(S) \arrow[d, "\tau"'] \\
\mathfrak{F}(S) \\
S \arrow[u, "\eta"]
\end{tikzcd}
$$

Free L-algebras

$\mathfrak{F}(S) = (\mathfrak{F}(S), \tau)$: free L–algebra on $S \in \mathcal{D}$ generators.

$$\begin{tikzcd}
L\mathfrak{F}(S) \arrow[d, "\tau"'] & LX \arrow[d, "\alpha"] \\
\mathfrak{F}(S) & X \\
S \arrow[u, "\eta"] \arrow[ur, "\forall f"'] &
\end{tikzcd}$$

Free L-algebras

$\mathfrak{F}(S) = (\mathfrak{F}(S), \tau)$: free L–algebra on $S \in \mathcal{D}$ generators.

$$
\begin{tikzcd}
L\mathfrak{F}(S) \arrow[r, dashed, "Lf^\sharp"] \arrow[d, "\tau"'] & LX \arrow[d, "\alpha"] \\
\mathfrak{F}(S) \arrow[r, dashed, "f^\sharp"] & X \\
S \arrow[u, "\eta"] \arrow[ur, "\forall f"'] &
\end{tikzcd}
$$

Equations for L on S

$\mathfrak{F}(S) = (\mathfrak{F}(S), \tau)$: free L–algebra on $S \in \mathcal{D}$ generators.

Equations: L-algebra epimorphisms e_P with domain $\mathfrak{F}(S)$.

Equations for L on S

$\mathfrak{F}(S) = (\mathfrak{F}(S), \tau)$: free L–algebra on $S \in \mathcal{D}$ generators.

Equations: L-algebra epimorphisms e_P with domain $\mathfrak{F}(S)$.

$$\begin{tikzcd}
L\mathfrak{F}(S) \arrow[r, "Le_P"] \arrow[d, "\tau"'] & LP \arrow[d, "\zeta"] \\
\mathfrak{F}(S) \arrow[r, "e_P"] & P \\
S \arrow[u, "\eta"] &
\end{tikzcd}$$

Equations for L on S

$\mathfrak{F}(S) = (\mathfrak{F}(S), \tau)$: free L–algebra on $S \in \mathcal{D}$ generators.

Equations: L-algebra epimorphisms e_P with domain $\mathfrak{F}(S)$.

$$\begin{tikzcd}
L\mathfrak{F}(S) \arrow[r, "Le_P"] \arrow[d, "\tau"'] & LP \arrow[d, "\zeta"] & LX \arrow[d, "\alpha"] \\
\mathfrak{F}(S) \arrow[r, "e_P"] & P & X \\
S \arrow[u, "\eta"] & &
\end{tikzcd}$$

Equations for L on S

$\mathfrak{F}(S) = (\mathfrak{F}(S), \tau)$: free L–algebra on $S \in \mathcal{D}$ generators.

Equations: L-algebra epimorphisms e_P with domain $\mathfrak{F}(S)$.

$$(X, \alpha) \vDash e_P$$

$$\Updownarrow$$

$$\begin{tikzcd}
L\mathfrak{F}(S) \arrow[r, "Le_P"] \arrow[d, "\tau"'] & LP \arrow[r, dashed, "Lg_f"] \arrow[d, "\zeta"] & LX \arrow[d, "\alpha"] \\
\mathfrak{F}(S) \arrow[r, "e_P"] \arrow[rr, bend right, "f^\sharp"'] & P \arrow[r, dashed, "g_f"] & X \\
S \arrow[u, "\eta"] \arrow[urr, bend right, "\forall f"'] & &
\end{tikzcd}$$

Equations for L on S

$\mathfrak{F}(S) = (\mathfrak{F}(S), \tau)$: free L–algebra on $S \in \mathcal{D}$ generators.

$\mathrm{eq}(L, S)$: category of equations for L on S generators.

Equations for L on S

$\mathfrak{F}(S) = (\mathfrak{F}(S), \tau)$: free L–algebra on $S \in \mathcal{D}$ generators.

$\mathrm{eq}(L, S)$: category of equations for L on S generators.

- Objects: L-algebra epimorphisms with domain $\mathfrak{F}(S)$.

Equations for L on S

$\mathfrak{F}(S) = (\mathfrak{F}(S), \tau)$: free L–algebra on $S \in \mathcal{D}$ generators.

$\mathrm{eq}(L, S)$: category of equations for L on S generators.

- Objects: L-algebra epimorphisms with domain $\mathfrak{F}(S)$.
- Morphisms: L-algebra morphisms between the codomains s.t.:

$$\begin{tikzcd}
\mathfrak{F}(S) \arrow[r, "e_{X_1}"] \arrow[dr, "e_{X_2}"'] & X_1 \arrow[d, "f"] \\
& X_2
\end{tikzcd}$$

Cofree B-coalgebras

$\mathfrak{C}(R) = (\mathfrak{C}(R), \upsilon)$: cofree B–coalgebra on $R \in \mathcal{C}$ colours.

Cofree B-coalgebras

$\mathfrak{C}(R) = (\mathfrak{C}(R), \upsilon)$: cofree B–coalgebra on $R \in \mathcal{C}$ colours.

$$\begin{tikzcd} B\mathfrak{C}(R) \\ \mathfrak{C}(R) \arrow[u, "\upsilon"'] \arrow[d, "\epsilon"'] \\ R \end{tikzcd}$$

Cofree B-coalgebras

$\mathfrak{C}(R) = (\mathfrak{C}(R), \upsilon)$: cofree B–coalgebra on $R \in \mathcal{C}$ colours.

$$\begin{tikzcd}
BY & & B\mathfrak{C}(R) \\
Y \arrow[u, "\beta"] \arrow[drr, "\forall f"'] & & \mathfrak{C}(R) \arrow[u, "\upsilon"'] \arrow[d, "\epsilon"] \\
& & R
\end{tikzcd}$$

Cofree B-coalgebras

$\mathfrak{C}(R) = (\mathfrak{C}(R), \upsilon)$: cofree B–coalgebra on $R \in \mathcal{C}$ colours.

$$\begin{tikzcd}
BY \arrow[r, dashed, "Bf^\flat"] & B\mathfrak{C}(R) \\
Y \arrow[u, "\beta"] \arrow[r, dashed, "f^\flat"] \arrow[dr, "\forall f"'] & \mathfrak{C}(R) \arrow[u, "\upsilon"'] \arrow[d, "\epsilon"] \\
& R
\end{tikzcd}$$

Coequations for B on R

$\mathfrak{C}(R) = (\mathfrak{C}(R), \upsilon)$: cofree B–coalgebra on $R \in \mathcal{C}$ colours.

Coequations: B-coalgebra monomorphisms m_Q
with codomain $\mathfrak{C}(R)$.

Coequations for B on R

$\mathfrak{C}(R) = (\mathfrak{C}(R), \upsilon)$: cofree B–coalgebra on $R \in \mathcal{C}$ colours.

Coequations: B-coalgebra monomorphisms m_Q with codomain $\mathfrak{C}(R)$.

$$\begin{tikzcd}
BQ \arrow[r, "Bm_Q"] & L\mathfrak{C}(R) \\
Q \arrow[u, "\delta"'] \arrow[r, "m_Q"] & \mathfrak{C}(R) \arrow[u, "\upsilon"'] \arrow[d, "\epsilon"] \\
& R
\end{tikzcd}$$

Coequations for B on R

$\mathfrak{C}(R) = (\mathfrak{C}(R), \upsilon)$: cofree B–coalgebra on $R \in \mathcal{C}$ colours.

Coequations: B-coalgebra monomorphisms m_Q
with codomain $\mathfrak{C}(R)$.

$$\begin{tikzcd} BY \\ Y \arrow[u, "\beta"] \end{tikzcd} \qquad \begin{tikzcd} BQ \arrow[r, "Bm_Q"] & L\mathfrak{C}(R) \\ Q \arrow[u, "\delta"'] \arrow[r, "m_Q"] & \mathfrak{C}(R) \arrow[u, "\upsilon"'] \arrow[d, "\epsilon"] \\ & R \end{tikzcd}$$

Coequations for B on R

$\mathfrak{C}(R) = (\mathfrak{C}(R), \upsilon)$: cofree B–coalgebra on $R \in \mathcal{C}$ colours.

Coequations: B-coalgebra monomorphisms m_Q with codomain $\mathfrak{C}(R)$.

$$(Y, \beta) \models m_Q$$

$$\Updownarrow$$

$$\begin{tikzcd}
BY \arrow[r, dashed, "Bg_f"] & BQ \arrow[r, "Bm_Q"] & L\mathfrak{C}(R) \\
Y \arrow[u, "\beta"] \arrow[r, dashed, "g_f"] \arrow[rr, bend right, "f^\flat"'] \arrow[rrd, bend right, "\forall f"'] & Q \arrow[u, "\delta"'] \arrow[r, "m_Q"] & \mathfrak{C}(R) \arrow[u, "\upsilon"'] \arrow[d, "\epsilon"] \\
 & & R
\end{tikzcd}$$

Coequations for B on R

$\mathfrak{C}(R) = (\mathfrak{C}(R), \upsilon)$: cofree B–coalgebra on $R \in \mathcal{C}$ colours.

$\mathrm{coeq}(B, R)$: category of coequations for B on R colours.

Coequations for B on R

$\mathfrak{C}(R) = (\mathfrak{C}(R), \upsilon)$: cofree B–coalgebra on $R \in \mathcal{C}$ colours.

$\mathrm{coeq}(B, R)$: category of coequations for B on R colours.

- Objects: B-coalgebra monomorphisms with codomain $\mathfrak{C}(R)$.

Coequations for B on R

$\mathfrak{C}(R) = (\mathfrak{C}(R), \upsilon)$: cofree B–coalgebra on $R \in \mathcal{C}$ colours.

$\mathrm{coeq}(B, R)$: category of coequations for B on R colours.

- Objects: B-coalgebra monomorphisms with codomain $\mathfrak{C}(R)$.
- Morphisms: B-coalgebra morphisms between the domains s.t.:

$$\begin{tikzcd} Y_1 \arrow[rr, "m_{Y_1}"] \arrow[d, "g"'] & & \mathfrak{C}(R) \\ Y_2 \arrow[urr, "m_{Y_2}"'] & & \end{tikzcd}$$

Contravariant adjunctions

A *contravariant adjunction* between two contravariant functors $F : \mathcal{C} \to \mathcal{D}$ and $G : \mathcal{D} \to \mathcal{C}$, denoted as $F \dashv\vdash G$, is a bijection

$$\mathcal{D}(X, FY) \cong \mathcal{C}(Y, GX)$$

which is natural in both $X \in \mathcal{D}$ and $Y \in \mathcal{C}$.

Contravariant adjunctions

A *contravariant adjunction* between two contravariant functors $F : \mathcal{C} \to \mathcal{D}$ and $G : \mathcal{D} \to \mathcal{C}$, denoted as $F \dashv\vdash G$, is a bijection

$$\mathcal{D}(X, FY) \cong \mathcal{C}(Y, GX)$$

which is natural in both $X \in \mathcal{D}$ and $Y \in \mathcal{C}$.

Main setting:

$$\begin{tikzcd}
B \circlearrowright \mathcal{C} \arrow[rr, bend left, "F"] \arrow[rr, phantom, "\dashv\vdash"] & & \mathcal{D} \circlearrowleft L \arrow[ll, bend left, "G"]
\end{tikzcd}$$

Contravariant adjunctions

Proposition. (Hermida and Jacobs) If there is a natural isomorphism $\gamma : GL \Rightarrow BG$ then we can lift the contravariant adjunction as in the picture:

$$
\begin{tikzcd}
\mathrm{coalg}(B) \arrow[d] \arrow[r, bend left, "\widehat{F}"] \arrow[r, phantom, "\perp"] & \mathrm{alg}(L) \arrow[d] \arrow[l, bend left, "\widehat{G}"] \\
\mathcal{C} \arrow[loop left, "B"] \arrow[r, bend left, "F"] \arrow[r, phantom, "\perp"] & \mathcal{D} \arrow[loop right, "L"] \arrow[l, bend left, "G"]
\end{tikzcd}
$$

Contravariant adjunctions

Proposition. (Hermida and Jacobs) If there is a natural isomorphism $\gamma : GL \Rightarrow BG$ then we can lift the duality as in the picture:

$$
\begin{tikzcd}
\mathrm{coalg}(B) \arrow[rr, bend left, "\widehat{F}"] \arrow[rr, phantom, "\cong"] \arrow[d] & & \mathrm{alg}(L) \arrow[ll, bend left, "\widehat{G}"] \arrow[d] \\
\mathcal{C} \arrow[loop left, "B"] \arrow[rr, bend left, "F"] \arrow[rr, phantom, "\cong"] & & \mathcal{D} \arrow[ll, bend left, "G"] \arrow[loop right, "L"]
\end{tikzcd}
$$

Duality between equations and coequations

$$\begin{tikzcd} B \arrow[r, hook] & \mathcal{C} \arrow[rr, bend left, "F"] \arrow[rr, phantom, "\cong"] & & \mathcal{D} \arrow[ll, bend left, "G"] & L \arrow[l, hook'] \end{tikzcd}$$

Duality between equations and coequations

$$
\begin{tikzcd}
\mathrm{coalg}(B) \arrow[rr, bend left, "\widehat{F}", "\shortmid" marking] \arrow[d] \arrow[rr, phantom, "\cong"] & & \mathrm{alg}(L) \arrow[ll, bend left, "\widehat{G}", "\shortmid" marking] \arrow[d] \\
B \circlearrowright \mathcal{C} \arrow[rr, bend left, "F", "\shortmid" marking] \arrow[rr, phantom, "\cong"] & & \mathcal{D} \circlearrowleft L \arrow[ll, bend left, "G", "\shortmid" marking]
\end{tikzcd}
$$

Duality between equations and coequations

$$\begin{tikzcd}
\mathrm{coeq}(B, G(S)) \arrow[r, bend left, "\widehat{F}"] \arrow[r, phantom, "\cong"] \arrow[d] & \mathrm{eq}(L, S) \arrow[l, bend left, "\widehat{G}"] \arrow[d] \\
\mathrm{coalg}(B) \arrow[r, bend left, "\widehat{F}"] \arrow[r, phantom, "\cong"] \arrow[d] & \mathrm{alg}(L) \arrow[l, bend left, "\widehat{G}"] \arrow[d] \\
B \circlearrowright \mathcal{C} \arrow[r, bend left, "F"] \arrow[r, phantom, "\cong"] & \mathcal{D} \circlearrowleft L \arrow[l, bend left, "G"]
\end{tikzcd}$$

Duality between equations and coequations

$$
\begin{tikzcd}
\mathrm{coeq}(B, G(S)) \arrow[rr, bend left, "\widehat{F}"] \arrow[d] & \cong & \mathrm{eq}(L, S) \arrow[ll, bend left, "\widehat{G}"] \arrow[d] & \ni e_P \\
\mathrm{coalg}(B) \arrow[rr, bend left, "\widehat{F}"] \arrow[d] & \cong & \mathrm{alg}(L) \arrow[ll, bend left, "\widehat{G}"] \arrow[d] & \ni (X, \alpha) \\
B \circlearrowright \mathcal{C} \arrow[rr, bend left, "F"] & \cong & \mathcal{D} \circlearrowleft L \arrow[ll, bend left, "G"] &
\end{tikzcd}
$$

Duality between equations and coequations

$$
\begin{tikzcd}[column sep=huge]
\mathrm{coeq}(B,G(S)) \arrow[r, bend left, "\widehat{F}"] \arrow[r, phantom, "\cong"] \arrow[d] & \mathrm{eq}(L,S) \ni e_P \arrow[l, bend left, "\widehat{G}"] \arrow[d] \\
\mathrm{coalg}(B) \arrow[r, bend left, "\widehat{F}"] \arrow[r, phantom, "\cong"] \arrow[d] & \mathrm{alg}(L) \ni (X,\alpha) \arrow[l, bend left, "\widehat{G}"] \arrow[d] \\
B \circlearrowright \mathcal{C} \arrow[r, bend left, "F"] \arrow[r, phantom, "\cong"] & \mathcal{D} \circlearrowleft L \arrow[l, bend left, "G"]
\end{tikzcd}
$$

$$(X,\alpha) \models e_P \quad \Leftrightarrow \quad \widehat{G}(X,\alpha) \mathrel{|}\!\models \widehat{G}(e_P)$$

Duality between equations and coequations

$$\begin{tikzcd}
m_Q \in \mathrm{coeq}(B, G(S)) \arrow[r, bend left, "\widehat{F}"] \arrow[r, phantom, "\cong"] \arrow[d] & \mathrm{eq}(L, S) \ni e_P \arrow[l, bend left, "\widehat{G}"] \arrow[d] \\
(Y, \beta) \in \mathrm{coalg}(B) \arrow[r, bend left, "\widehat{F}"] \arrow[r, phantom, "\cong"] \arrow[d] & \mathrm{alg}(L) \ni (X, \alpha) \arrow[l, bend left, "\widehat{G}"] \arrow[d] \\
\mathcal{C} \arrow[loop left, "B"] \arrow[r, bend left, "F"] \arrow[r, phantom, "\cong"] & \mathcal{D} \arrow[loop right, "L"] \arrow[l, bend left, "G"]
\end{tikzcd}$$

$$(X, \alpha) \vDash e_P \quad \Leftrightarrow \quad \widehat{G}(X, \alpha) \Vdash \widehat{G}(e_P)$$

Duality between equations and coequations

$$
\begin{tikzcd}
m_Q \in \mathrm{coeq}(B, G(S)) \arrow[r, bend left, "\widehat{F}"] \arrow[r, phantom, "\cong"] \arrow[d] & \mathrm{eq}(L, S) \ni e_P \arrow[l, bend left, "\widehat{G}"] \arrow[d] \\
(Y, \beta) \in \mathrm{coalg}(B) \arrow[r, bend left, "\widehat{F}"] \arrow[r, phantom, "\cong"] \arrow[d] & \mathrm{alg}(L) \ni (X, \alpha) \arrow[l, bend left, "\widehat{G}"] \arrow[d] \\
B \circlearrowright \mathcal{C} \arrow[r, bend left, "F"] \arrow[r, phantom, "\cong"] & \mathcal{D} \circlearrowleft L \arrow[l, bend left, "G"]
\end{tikzcd}
$$

$$(X, \alpha) \vDash e_P \quad \Leftrightarrow \quad \widehat{G}(X, \alpha) \models \widehat{G}(e_P)$$

$$\widehat{F}(Y, \beta) \vDash \widehat{F}(m_Q) \quad \Leftrightarrow \quad (Y, \beta) \models m_Q$$

Consider the setting:

$$
\begin{tikzcd}
\mathsf{B} = (B, \epsilon, \delta) \circlearrowright \mathcal{C} \arrow[rr, bend left, "F"] \arrow[rr, phantom, "\perp"] & & \mathcal{D} \circlearrowleft \mathsf{L} = (L, \eta, \mu) \arrow[ll, bend left, "G"]
\end{tikzcd}
$$

Liftings to Eilenberg-Moore categories

Consider the setting:

$$\mathsf{B} = (B, \epsilon, \delta) \circlearrowright \mathcal{C} \underset{G}{\overset{F}{\rightleftarrows}} \mathcal{D} \circlearrowleft \mathsf{L} = (L, \eta, \mu) \qquad F \dashv G$$

Assume there is a natural isomorphism $\gamma : GL \Rightarrow BG$ s.t.

$$\begin{tikzcd}
G & BG \arrow[l, "\epsilon_G"'] \\
 & GL \arrow[ul, "G\eta"] \arrow[u, "\gamma"']
\end{tikzcd}
\qquad
\begin{tikzcd}
BBG & & BG \arrow[ll, "\delta_G"'] \\
BGL \arrow[u, "B\gamma"] & GLL \arrow[l, "\gamma_L"] & GL \arrow[l, "G\mu"] \arrow[u, "\gamma"']
\end{tikzcd}$$

Liftings to Eilenberg-Moore categories

Proposition. Consider the setting:

$$\begin{tikzcd} \mathsf{B} = (B, \epsilon, \delta) \arrow[loop left] & \mathcal{C} \arrow[rr, bend left, "F"] \arrow[rr, phantom, "\perp"] & & \mathcal{D} \arrow[ll, bend left, "G"] & \mathsf{L} = (L, \eta, \mu) \arrow[loop right] \end{tikzcd}$$

Assume there is a natural isomorphism $\gamma : GL \Rightarrow BG$ s.t.

$$\begin{tikzcd} G & BG \arrow[l, "\epsilon_G"'] \\ & GL \arrow[ul, "G\eta"] \arrow[u, "\gamma"'] \end{tikzcd} \qquad \begin{tikzcd} BBG & & BG \arrow[ll, "\delta_G"'] \\ BGL \arrow[u, "B\gamma"] & GLL \arrow[l, "\gamma_L"] & GL \arrow[l, "G\mu"] \arrow[u, "\gamma"'] \end{tikzcd}$$

Then the contravariant adjunction lifts to the corresponding Eilenberg–Moore categories.

Proposition. Consider the setting:

$$
\begin{tikzcd}
\mathsf{B} = (B, \epsilon, \delta) \arrow[loop left] & \mathcal{C} \arrow[rr, bend left, "F"] \arrow[rr, phantom, "\color{red}\cong"] & & \mathcal{D} \arrow[ll, bend left, "G"] \arrow[loop right] & \mathsf{L} = (L, \eta, \mu)
\end{tikzcd}
$$

Assume there is a natural isomorphism $\gamma : GL \Rightarrow BG$ s.t.

$$
\begin{tikzcd}
G & BG \arrow[l, "\epsilon_G"'] \\
 & GL \arrow[u, "\gamma"'] \arrow[ul, "G\eta"]
\end{tikzcd}
\qquad
\begin{tikzcd}
BBG & & BG \arrow[ll, "\delta_G"'] \\
BGL \arrow[u, "B\gamma"] & GLL \arrow[l, "\gamma_L"] & GL \arrow[l, "G\mu"] \arrow[u, "\gamma"']
\end{tikzcd}
$$

Then the duality lifts to the corresponding Eilenberg–Moore categories.

Defining a monad from a comonad

Consider the setting:

$$\mathsf{B} = (B, \epsilon, \delta) \circlearrowright \mathcal{C} \; \begin{tikzcd} \mathcal{C} \arrow[r, bend left, "F"] \arrow[r, phantom, "\perp"] & \mathcal{D} \arrow[l, bend left, "G"] \end{tikzcd}$$

Defining a monad from a comonad

Consider the setting:

$$\begin{tikzcd} \mathsf{B} = (B,\epsilon,\delta) \circlearrowright \mathcal{C} \arrow[rr, bend left, "F"] \arrow[rr, phantom, "\perp"] && \mathcal{D} \arrow[ll, bend left, "G"] \end{tikzcd}$$

Define $\mathsf{L} = (L, \eta, \mu)$ as:

- $L = FBG$
- $\eta = (Id_{\mathcal{D}} \xRightarrow{\eta^{FG}} FG \xRightarrow{F\epsilon_G} FBG)$
- $\mu = (FBGFBG \xRightarrow{FB\eta^{GF}_{BG}} FBBG \xRightarrow{F\delta_G} FBG)$

Defining a monad from a comonad

Consider the setting:

$$\begin{tikzcd} \mathsf{B} = (B, \epsilon, \delta) \circlearrowright \mathcal{C} \arrow[rr, bend left, "F"] \arrow[rr, phantom, "\perp"] & & \mathcal{D} \arrow[ll, bend left, "G"] \end{tikzcd}$$

Define $\mathsf{L} = (L, \eta, \mu)$ as:

- $L = FBG$
- $\eta = (Id_{\mathcal{D}} \xRightarrow{\eta^{FG}} FG \xRightarrow{F\epsilon_G} FBG)$
- $\mu = (FBGFBG \xRightarrow{FB\eta^{GF}_{BG}} FBBG \xRightarrow{F\delta_G} FBG)$

Then $\mathsf{L} = (L, \eta, \mu)$ is a monad on $\mathcal{D}$.

Defining a monad from a comonad

Consider the setting:

$$\begin{tikzcd} \mathsf{B} = (B, \epsilon, \delta) \arrow[loop left] & \mathcal{C} \arrow[rr, bend left, "F"] \arrow[rr, phantom, "\dashv"] & & \mathcal{D} \arrow[ll, bend left, "G"] \end{tikzcd}$$

Define $\mathsf{L} = (L, \eta, \mu)$ as:

- $L = FBG$
- $\eta = (Id_{\mathcal{D}} \xRightarrow{\eta^{FG}} FG \xRightarrow{F\epsilon_G} FBG)$
- $\mu = (FBGFBG \xRightarrow{FB\eta^{GF}_{BG}} FBBG \xRightarrow{F\delta_G} FBG)$

Then $\mathsf{L} = (L, \eta, \mu)$ is a monad on $\mathcal{D}$. If η^{GF} is a natural isomorphism then the contravariant adjunction lifts to the corresponding Eilenberg–Moore categories.

Defining a comonad from a monad

Consider the setting:

$$
\begin{tikzcd}
\mathcal{C} \arrow[rr, bend left, "F"] \arrow[rr, phantom, "\cong"] & & \mathcal{D} \arrow[ll, bend left, "G"] \arrow[loop right]
\end{tikzcd}
\mathsf{L} = (L, \eta, \mu)
$$

Defining a comonad from a monad

Consider the setting:

$$\begin{tikzcd} \mathcal{C} \arrow[rr, bend left, "F"] \arrow[rr, phantom, "\cong"] & & \mathcal{D} \arrow[ll, bend left, "G"] \arrow[loop right] \end{tikzcd} \mathsf{L} = (L, \eta, \mu)$$

Define $\mathsf{B} = (B, \epsilon, \delta)$ as:

- $B = GLF$
- $\epsilon = (GLF \xRightarrow{G\eta_F} GF \xRightarrow{(\eta^{GF})^{-1}} Id_{\mathcal{C}})$
- $\delta = (GLF \xRightarrow{G\mu_F} GLLF \xRightarrow{GL(\eta^{FG})^{-1}_{LF}} GLFGLF)$

Defining a comonad from a monad

Consider the setting:

$$\begin{tikzcd} \mathcal{C} \arrow[rr, bend left, "F"] \arrow[rr, phantom, "\cong"] & & \mathcal{D} \arrow[ll, bend left, "G"] \arrow[loop right] \end{tikzcd} \mathsf{L} = (L, \eta, \mu)$$

Define $\mathsf{B} = (B, \epsilon, \delta)$ as:

- $B = GLF$
- $\epsilon = (GLF \xRightarrow{G\eta_F} GF \xRightarrow{(\eta^{GF})^{-1}} Id_{\mathcal{C}})$
- $\delta = (GLF \xRightarrow{G\mu_F} GLLF \xRightarrow{GL(\eta^{FG})^{-1}_{LF}} GLFGLF)$

Then $\mathsf{B} = (B, \epsilon, \delta)$ is a comonad on $\mathcal{C}$ and the duality lifts to the corresponding Eilenberg–Moore categories.

Application: dynamical systems

Let $M = (M, \cdot, e)$ be a monoid.

Application: dynamical systems

Let $M = (M, \cdot, e)$ be a monoid.

$$\begin{tikzcd}
\mathrm{CABA} \arrow[rr, bend left, "F = \mathrm{At}(_)"] \arrow[rr, phantom, "\cong"] & & \mathrm{Set} \arrow[ll, bend left, "G = 2^{-}"]
\end{tikzcd}$$

Application: dynamical systems

Let $M = (M, \cdot, e)$ be a monoid.

$$\begin{tikzcd} \mathrm{CABA} \arrow[rr, bend left, "F = \mathrm{At}(\_)"] \arrow[rr, phantom, "\cong"] & & \mathrm{Set} \arrow[ll, bend left, "G = 2^{-}"] \arrow[loop right, "\mathsf{L} = (L, \eta, \mu)"] \end{tikzcd}$$

$$LX = X \times M$$

$$\mu_X(x, m, n) = (x, m \cdot n)$$

Application: dynamical systems

Let $M = (M, \cdot, e)$ be a monoid.

$$\begin{tikzcd} \mathrm{CABA} \arrow[rr, bend left, "F = \mathrm{At}(\_)"] \arrow[rr, phantom, "\cong"] & & \mathrm{Set} \arrow[ll, bend left, "G = 2^{-}"] \arrow[loop right, "\mathsf{L} = (L, \eta, \mu)"] \end{tikzcd}$$

$$LX = X \times M$$

$$\mu_X(x, m, n) = (x, m \cdot n)$$

$\mathrm{Alg}(\mathsf{L})$: dynamical systems on M (monoid actions).

Application: dynamical systems

Let $M = (M, \cdot, e)$ be a monoid.

$$
\begin{tikzcd}
\mathsf{B} = (B, \epsilon, \delta) \arrow[r, bend left] & \mathrm{CABA} \arrow[rr, bend left, "F = \mathrm{At}(\_)"] & \cong & \mathrm{Set} \arrow[ll, bend left, "G = 2^{-}"] & \mathsf{L} = (L, \eta, \mu) \arrow[l, bend left]
\end{tikzcd}
$$

$BY = Y^M$

$\delta_Y(f) = \lambda m.\lambda n f(n \cdot m)$

$LX = X \times M$

$\mu_X(x, m, n) = (x, m \cdot n)$

$\mathrm{Alg}(\mathsf{L})$: dynamical systems on M (monoid actions).

Let $M = (M, \cdot, e)$ be a monoid.

$$\begin{tikzcd}
\mathsf{B} = (B, \epsilon, \delta) & \mathrm{CABA} \arrow[loop left] \arrow[rr, bend left, "F = \mathrm{At}(_)"] \arrow[rr, phantom, "\cong"] & & \mathrm{Set} \arrow[ll, bend left, "G = 2^{-}"] \arrow[loop right] & \mathsf{L} = (L, \eta, \mu)
\end{tikzcd}$$

$$BY = Y^M \qquad\qquad LX = X \times M$$

$$\delta_Y(f) = \lambda m.\lambda n f(n \cdot m) \qquad\qquad \mu_X(x, m, n) = (x, m \cdot n)$$

$\mathrm{Alg}(\mathsf{L})$: dynamical systems on M (monoid actions).

$\mathrm{Coalg}(\mathsf{B})$: (Y, β) s.t. $\beta(x)(e) = x$ and $\beta(\beta(x)(m))(n) = \beta(x)(n \cdot m)$.

Application: dynamical systems

Let $M = (M, \cdot, e)$ be a monoid.

$$
\begin{tikzcd}
\mathsf{B} = (B, \epsilon, \delta) & \mathrm{CABA} \arrow[loop left] \arrow[r, bend left, "F = \mathrm{At}(\_)"] \arrow[r, phantom, "\cong"] & \mathrm{Set} \arrow[l, bend left, "G = 2^{-}"] \arrow[loop right] & \mathsf{L} = (L, \eta, \mu)
\end{tikzcd}
$$

$$BY = Y^M \qquad LX = X \times M$$

$$\delta_Y(f) = \lambda m.\lambda n f(n \cdot m) \qquad \mu_X(x, m, n) = (x, m \cdot n)$$

$$GLX = 2^{X \times M} \overset{\gamma_X}{\cong} (2^X)^M = BGX$$

$\mathrm{Alg}(\mathsf{L})$: dynamical systems on M (monoid actions).

$\mathrm{Coalg}(\mathsf{B})$: (Y, β) s.t. $\beta(x)(e) = x$ and $\beta(\beta(x)(m))(n) = \beta(x)(n \cdot m)$.

Application: dynamical systems

Let $M = (M, \cdot, e)$ be a monoid.

$$\begin{tikzcd}
\mathrm{CABA} \arrow[loop left, "\mathsf{B} = (B,\epsilon,\delta)"] \arrow[r, bend left, "F = \mathrm{At}(_)"] \arrow[r, phantom, "\cong"] & \mathrm{Set} \arrow[l, bend left, "G = 2^{-}"] \arrow[loop right, "\mathsf{L} = (L,\eta,\mu)"]
\end{tikzcd}$$

$$BY = Y^M \qquad G = 2^{-} \qquad LX = X \times M$$

$$\delta_Y(f) = \lambda m.\lambda n f(n \cdot m) \qquad \mu_X(x, m, n) = (x, m \cdot n)$$

$$GLX = 2^{X \times M} \overset{\gamma_X}{\cong} (2^X)^M = BGX$$

$\mathrm{Alg}(\mathsf{L})$: dynamical systems on M (monoid actions).

$\mathrm{Coalg}(\mathsf{B})$: (Y, β) s.t. $\beta(x)(e) = x$ and $\beta(\beta(x)(m))(n) = \beta(x)(n \cdot m)$.

$\mathrm{Alg}(\mathsf{L}) \cong \mathrm{Coalg}(\mathsf{B})^{op}$.

Application: dynamical systems

Let $M = (M, \cdot, e)$ be a monoid.

$$
\begin{tikzcd}
\mathsf{B} = (B, \epsilon, \delta) & \mathrm{CABA} \arrow[loop left] \arrow[rr, bend left, "F = \mathrm{At}(\_)"] \arrow[rr, phantom, "\cong"] & & \mathrm{Set} \arrow[ll, bend left, "G = 2^{-}"] \arrow[loop right] & \mathsf{L} = (L, \eta, \mu)
\end{tikzcd}
$$

$$BY = Y^M \qquad\qquad LX = X \times M$$

$$\delta_Y(f) = \lambda m.\lambda n f(n \cdot m) \qquad\qquad \mu_X(x, m, n) = (x, m \cdot n)$$

$$GLX = 2^{X \times M} \overset{\gamma_X}{\cong} (2^X)^M = BGX$$

$\mathrm{Eq}(\mathsf{L}, 1)$: Right congruences on M.

Application: dynamical systems

Let $M = (M, \cdot, e)$ be a monoid.

$$\begin{tikzcd}[column sep=huge] \mathrm{CABA} \arrow[loop left, "\mathsf{B} = (B,\epsilon,\delta)"] \arrow[r, bend left, "F = \mathrm{At}(\_)"] \arrow[r, phantom, "\cong"] & \mathrm{Set} \arrow[l, bend left, "G = 2^{-}"] \arrow[loop right, "\mathsf{L} = (L,\eta,\mu)"] \end{tikzcd}$$

$$BY = Y^M \qquad LX = X \times M$$

$$\delta_Y(f) = \lambda m.\lambda n f(n \cdot m) \qquad \mu_X(x, m, n) = (x, m \cdot n)$$

$$GLX = 2^{X \times M} \overset{\gamma_X}{\cong} (2^X)^M = BGX$$

$\mathrm{Eq}(\mathsf{L}, 1)$: Right congruences on M.

$\mathrm{Coeq}(\mathsf{B}, 2)$: $Q \leq 2^M$ s.t. $\forall\ f \in Q$ and $m \in M$, ${}_n f \in Q$.
${}_n f(m) = f(m \cdot n)$.

Let $M = (M, \cdot, e)$ be a monoid.

$$\begin{tikzcd} \mathrm{CABA} \arrow[loop left, "\mathsf{B} = (B, \epsilon, \delta)"] \arrow[rr, bend left, "F = \mathrm{At}(\_)"] \arrow[rr, phantom, "\cong"] && \mathrm{Set} \arrow[ll, bend left, "G = 2^{-}"] \arrow[loop right, "\mathsf{L} = (L, \eta, \mu)"] \end{tikzcd}$$

$$BY = Y^M \qquad LX = X \times M$$

$$\delta_Y(f) = \lambda m.\lambda n f(n \cdot m) \qquad \mu_X(x, m, n) = (x, m \cdot n)$$

$$GLX = 2^{X \times M} \overset{\gamma_X}{\cong} (2^X)^M = BGX$$

$\mathrm{Eq}(\mathsf{L}, 1)$: Right congruences on M.

$\mathrm{Coeq}(\mathsf{B}, 2)$: $Q \leq 2^M$ s.t. $\forall\ f \in Q$ and $m \in M$, ${}_n f \in Q$.
${}_n f(m) = f(m \cdot n)$.

$\mathrm{Eq}(\mathsf{L}, 1) \cong \mathrm{Coeq}(\mathsf{B}, 2)^{op}$.

Application: deterministic automata

$$\begin{tikzcd} \mathrm{CABA} \arrow[rr, bend left, "F = \mathrm{At}(\_)"] \arrow[rr, phantom, "\cong"] & & \mathrm{Set} \arrow[ll, bend left, "G = 2^{-}"] \end{tikzcd}$$

Application: deterministic automata

$$\begin{tikzcd} \mathrm{CABA} \arrow[rr, bend left, "F = \mathrm{At}(\_)"] \arrow[rr, phantom, "\cong"] & & \mathrm{Set} \arrow[ll, bend left, "G = 2^{-}"] \arrow[loop right] \end{tikzcd} \quad \mathsf{L} = (L, \eta, \mu)$$

$$LX = X^*$$

Application: deterministic automata

$$
\begin{tikzcd}
\mathrm{CABA} \arrow[rr, bend left, "F = \mathrm{At}(\_)"] \arrow[rr, phantom, "\cong"] & & \mathrm{Set} \arrow[ll, bend left, "G = 2^{-}"] \arrow[loop right, "\mathsf{L} = (L, \eta, \mu)"]
\end{tikzcd}
$$

$$LX = X^*$$

- $\mathrm{Alg}(\mathsf{L})$: monoids.

Application: deterministic automata

$$
\begin{tikzcd}
\mathrm{CABA} \arrow[rr, bend left, "F = \mathrm{At}(\_)"] \arrow[rr, phantom, "\cong"] & & \mathrm{Set} \arrow[ll, bend left, "G = 2^{-}"] \arrow[loop right, "\mathsf{L} = (L, \eta, \mu)"]
\end{tikzcd}
$$

$$LX = X^*$$

- $\mathrm{Alg}(\mathsf{L})$: monoids.
- There exists a comonad $\mathsf{B} = (B, \epsilon, \delta)$ on CABA s.t. $\mathrm{Coalg}(\mathsf{B})^{op} \cong \mathrm{Alg}(\mathsf{L})$.

Application: deterministic automata

$$\begin{tikzcd} \mathrm{CABA} \arrow[rr, bend left, "F = \mathrm{At}(\_)"] \arrow[rr, phantom, "\cong"] & & \mathrm{Set} \arrow[ll, bend left, "G = 2^{-}"] \arrow[loop right, "\mathsf{L} = (L, \eta, \mu)"] \end{tikzcd}$$

$$LX = X^*$$

- $\mathrm{Alg}(\mathsf{L})$: monoids.
- There exists a comonad $\mathsf{B} = (B, \epsilon, \delta)$ on CABA s.t. $\mathrm{Coalg}(\mathsf{B})^{op} \cong \mathrm{Alg}(\mathsf{L})$.
- $\mathrm{Eq}(\mathsf{L}, A)$: congruences of A^*.

Application: deterministic automata

$$\begin{tikzcd} \mathrm{CABA} \arrow[r, bend left, "F = \mathrm{At}(\_)"] \arrow[r, phantom, "\cong"] & \mathrm{Set} \arrow[l, bend left, "G = 2^{-}"] \arrow[loop right, "\mathsf{L} = (L, \eta, \mu)"] \end{tikzcd}$$

$$LX = X^*$$

- $\mathrm{Alg}(\mathsf{L})$: monoids.
- There exists a comonad $\mathsf{B} = (B, \epsilon, \delta)$ on CABA s.t. $\mathrm{Coalg}(\mathsf{B})^{op} \cong \mathrm{Alg}(\mathsf{L})$.
- $\mathrm{Eq}(\mathsf{L}, A)$: congruences of A^*.
- $\mathrm{Coeq}(\mathsf{B}, 2^A)$: $Q \leq 2^{A^*}$ closed under left and right derivatives.

Application: deterministic automata

$$\begin{tikzcd} \mathrm{CABA} \arrow[rr, bend left, "F = \mathrm{At}(\_)"] \arrow[rr, phantom, "\cong"] & & \mathrm{Set} \arrow[ll, bend left, "G = 2^{-}"] \arrow[loop right, "\mathsf{L} = (L, \eta, \mu)"] \end{tikzcd}$$

$$LX = X^*$$

- $\mathrm{Alg}(\mathsf{L})$: monoids.
- There exists a comonad $\mathsf{B} = (B, \epsilon, \delta)$ on CABA s.t. $\mathrm{Coalg}(\mathsf{B})^{op} \cong \mathrm{Alg}(\mathsf{L})$.
- $\mathrm{Eq}(\mathsf{L}, A)$: congruences of A^*.
- $\mathrm{Coeq}(\mathsf{B}, 2^A)$: $Q \leq 2^{A^*}$ closed under left and right derivatives.
- $\mathrm{Eq}(\mathsf{L}, A) \cong \mathrm{Coeq}(\mathsf{B}, 2^A)^{op}$

Application: deterministic automata

$$
\begin{tikzcd}
\mathrm{CABA} \arrow[rr, bend left, "F = \mathrm{At}(\_)"] \arrow[rr, phantom, "\cong"] & & \mathrm{Set} \arrow[ll, bend left, "G = 2^{-}"] \arrow[loop right, "\mathsf{L} = (L, \eta, \mu)"]
\end{tikzcd}
\qquad LX = X^*
$$

- $\mathrm{Alg}(\mathsf{L})$: monoids.
- There exists a comonad $\mathsf{B} = (B, \epsilon, \delta)$ on CABA s.t. $\mathrm{Coalg}(\mathsf{B})^{op} \cong \mathrm{Alg}(\mathsf{L})$.
- $\mathrm{Eq}(\mathsf{L}, A)$: congruences of A^*.
- $\mathrm{Coeq}(\mathsf{B}, 2^A)$: $Q \leq 2^{A^*}$ closed under left and right derivatives.
- $\mathrm{Eq}(\mathsf{L}, A) \cong \mathrm{Coeq}(\mathsf{B}, 2^A)^{op}$: duality between equations and coequations for deterministic automata (Rutten).

Conclusions

- We proved a theorem for lifting contravariant adjunctions to Eilenberg-Moore categories.

Conclusions

- We proved a theorem for lifting contravariant adjunctions to Eilenberg-Moore categories.
- If the contravariant adjunction is a duality, then the lifting is also a duality and gives a duality between equations and coequations.

Conclusions

- We proved a theorem for lifting contravariant adjunctions to Eilenberg-Moore categories.
- If the contravariant adjunction is a duality, then the lifting is also a duality and gives a duality between equations and coequations.
- Define equations for coalgebras.

Conclusions

- We proved a theorem for lifting contravariant adjunctions to Eilenberg-Moore categories.
- If the contravariant adjunction is a duality, then the lifting is also a duality and gives a duality between equations and coequations.
- Define equations for coalgebras.
- Define coequations for algebras.

Conclusions

- We proved a theorem for lifting contravariant adjunctions to Eilenberg-Moore categories.
- If the contravariant adjunction is a duality, then the lifting is also a duality and gives a duality between equations and coequations.
- Define equations for coalgebras.
- Define coequations for algebras.
- A way of defining a monad from a given comonad and a contravariant adjunction.

Conclusions

- We proved a theorem for lifting contravariant adjunctions to Eilenberg-Moore categories.
- If the contravariant adjunction is a duality, then the lifting is also a duality and gives a duality between equations and coequations.
- Define equations for coalgebras.
- Define coequations for algebras.
- A way of defining a monad from a given comonad and a contravariant adjunction.
- A way of defining a comonad from a given monad and a duality.

Conclusions

- We proved a theorem for lifting contravariant adjunctions to Eilenberg-Moore categories.
- If the contravariant adjunction is a duality, then the lifting is also a duality and gives a duality between equations and coequations.
- Define equations for coalgebras.
- Define coequations for algebras.
- A way of defining a monad from a given comonad and a contravariant adjunction.
- A way of defining a comonad from a given monad and a duality. Using this setting one can prove an Eilenberg-like theorem for algebras on a monad. (CMCS short contributions).