THE DUAL EQUIVALENCE OF EQUATIONS AND COEQUATIONS FOR AUTOMATA

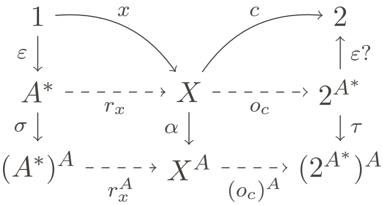
COIN: COalgebra In the Netherlands

Nijmegen, 2014

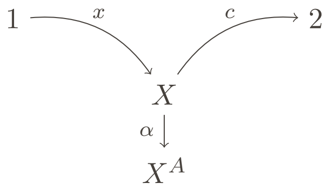
Enric Cosme Llópez [1]
Jan Rutten[2]

[1] **Universitat de València**
[2] **CWI and Radboud Universiteit Nijmegen**

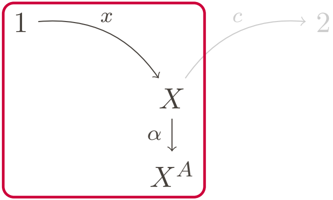
THE SCENE



THE SCENE

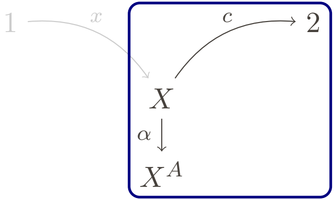


THE SCENE



Because of the isomorphism $X \to X^A \cong X \times A \to X$, the above frame can be seen as an algebra.

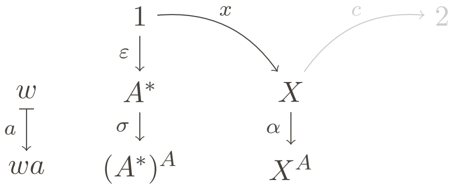
THE SCENE



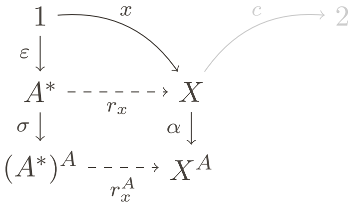
THE SCENE

$$
\begin{array}{ccccc}
1 & \xrightarrow{\quad x \quad} & & & \\
\varepsilon \big\downarrow & & \searrow & & \\
A^* & & X & \xrightarrow{\quad c \quad} & 2 \\
\sigma \big\downarrow & & \alpha \big\downarrow & & \\
(A^*)^A & & X^A & &
\end{array}
$$

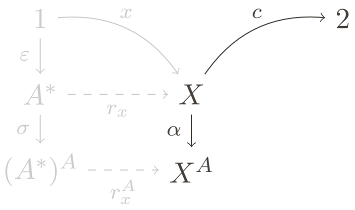
THE SCENE



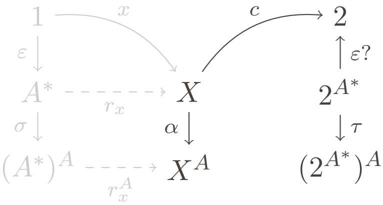
THE SCENE



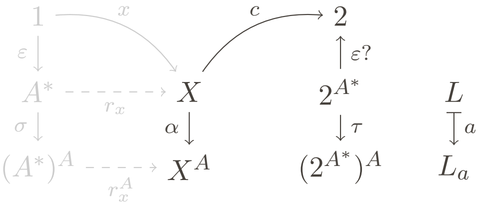
THE SCENE



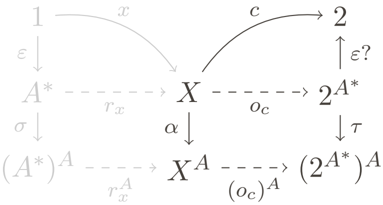
THE SCENE



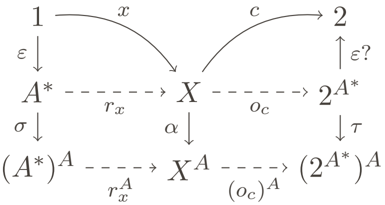
THE SCENE



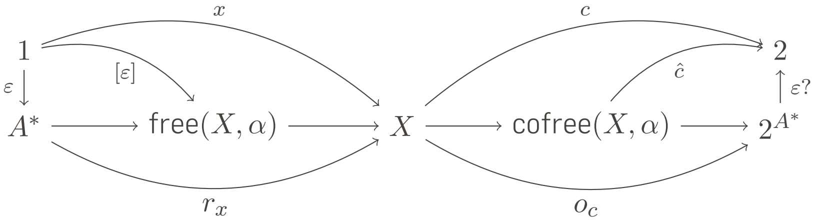
THE SCENE



THE SCENE



THE EXTENDED SCENE



EQUATIONS

Definition

A set of equations is a bisimulation equivalence $E \subseteq A^* \times A^*$ on the initial automaton (A^*, σ).

EQUATIONS

Definition

A set of equations is a bisimulation equivalence $E \subseteq A^* \times A^*$ on the initial automaton (A^*, σ).

Definition

We say that the pointed automaton (X, α, x) satisfies E

$$(X, \alpha, x) \models E \quad \Leftrightarrow \quad \forall (v, w) \in E, \;\; x_v = x_w$$

EQUATIONS

Definition

A set of equations is a bisimulation equivalence $E \subseteq A^* \times A^*$ on the initial automaton (A^*, σ).

Definition

We say that the pointed automaton (X, α, x) satisfies E

$$(X, \alpha, x) \models E \quad \Leftrightarrow \quad \forall (v, w) \in E,\ \ x_v = x_w$$

We define:

$$(X, \alpha) \models E \quad \Leftrightarrow \quad \forall x : 1 \to X,\ \ (X, \alpha, x) \models E$$

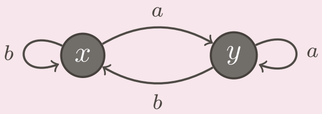
EQUATIONS

Let $v, w \in A^*$, we consider the shorthand $v = w$ to denote the smallest bisimulation equivalence on A^* containing (v, w).

EQUATIONS

Let $v, w \in A^*$, we consider the shorthand $v = w$ to denote the smallest bisimulation equivalence on A^* containing (v, w).

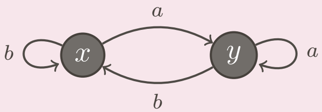
Example



EQUATIONS

Let $v, w \in A^*$, we consider the shorthand $v = w$ to denote the smallest bisimulation equivalence on A^* containing (v, w).

Example

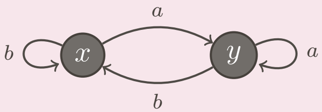


$$(X, \alpha, x) \models \{b = \varepsilon,\ ab = \varepsilon,\ aa = a\}$$

EQUATIONS

Let $v, w \in A^*$, we consider the shorthand $v = w$ to denote the smallest bisimulation equivalence on A^* containing (v, w).

Example



$$(X, \alpha, x) \models \{b = \varepsilon,\ ab = \varepsilon,\ aa = a\}$$

$$(X, \alpha, y) \models \{a = \varepsilon,\ ba = \varepsilon,\ bb = b\}$$

COEQUATIONS

Definition

A set of coequations is a subautomaton $D \leq 2^{A^*}$ of the final automaton $(2^{A^*}, \tau)$.

COEQUATIONS

Definition

A set of coequations is a subautomaton $D \leq 2^{A^*}$ of the final automaton $(2^{A^*}, \tau)$.

Definition

We say that the coloured automaton (X, α, c) satisfies D

$$(X, \alpha, c) \models D \quad \Leftrightarrow \quad \forall x \in X, \;\; o_c(x) \in D$$

COEQUATIONS

Definition

A set of coequations is a subautomaton $D \leq 2^{A^*}$ of the final automaton $(2^{A^*}, \tau)$.

Definition

We say that the coloured automaton (X, α, c) satisfies D

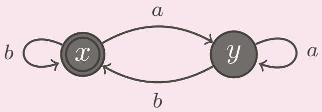
$$(X, \alpha, c) \models D \quad \Leftrightarrow \quad \forall x \in X,\; o_c(x) \in D$$

We define:

$$(X, \alpha) \models D \quad \Leftrightarrow \quad \forall c : X \to 2,\; (X, \alpha, c) \models D$$

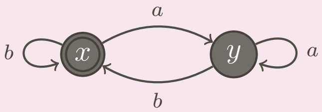
COEQUATIONS

Example



COEQUATIONS

Example

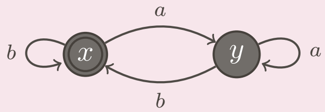


Under the observability map we obtain:

$$o_c(x) = (a^*b)^* \qquad o_c(y) = (a^*b)^+$$

COEQUATIONS

Example



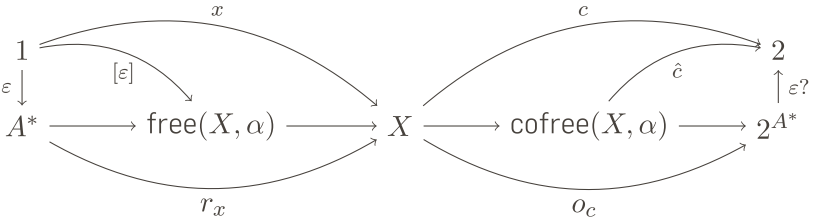
Under the observability map we obtain:

$$o_c(x) = (a^*b)^* \qquad o_c(y) = (a^*b)^+$$

therefore,

$$(X, \alpha, c) \models \{(a^*b)^*, (a^*b)^+\}$$

THE EXTENDED SCENE



FREE

Let (X, α) be an arbitrary automaton. We show how to construct an automaton that corresponds to the largest set of equations satisfied by (X, α). And, dually, we construct an automaton that corresponds to the smallest set of coequations satisfied by (X, α).

FREE

Let (X, α) be an arbitrary automaton. We show how to construct an automaton that corresponds to the largest set of equations satisfied by (X, α). And, dually, we construct an automaton that corresponds to the smallest set of coequations satisfied by (X, α).

For notational convenience we assume X to be finite but nothing will depend on that assumption.

FREE

Let $X = \{x_1, \ldots, x_n\}$ be the set of states of a finite automaton (X, α). We define a pointed automaton free(X, α) in two steps, as follows:

FREE

Let $X = \{x_1, \ldots, x_n\}$ be the set of states of a finite automaton (X, α). We define a pointed automaton free(X, α) in two steps, as follows:

(i) First, we take the product of the n pointed automata (X, x_i, α) that we obtain by letting the initial element x_i range over X. This yields a pointed automaton $(\Pi X, \bar{x}, \bar{\alpha})$ with

$$\Pi X = \prod_{x:1 \to X} X_x \quad \cong \quad X^n$$

(where $X_x = X$), with $\bar{x} = (x_1, \ldots, x_n)$, and with $\bar{\alpha} : \Pi X \to (\Pi X)^A$ defined by

$$\bar{\alpha}(y_1, \ldots, y_n)(a) = ((y_1)_a, \ldots, (y_n)_a)$$

FREE

(ii) Next we consider the reachability map $r_{\bar{x}} : A^* \to \Pi X$ and define:

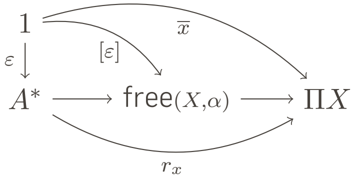
$$\mathsf{Eq}(X, \alpha) = \ker(r_{\bar{x}}) \qquad \mathsf{free}(X, \alpha) = A^*/\mathsf{Eq}(X, \alpha)$$

FREE

(ii) Next we consider the reachability map $r_{\bar{x}} : A^* \to \Pi X$ and define:

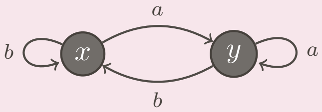
$$\mathsf{Eq}(X, \alpha) = \ker(r_{\bar{x}}) \qquad \mathsf{free}(X, \alpha) = A^*/\mathsf{Eq}(X, \alpha)$$

This yields the pointed automaton $(\mathsf{free}(X, \alpha), [\varepsilon], [\sigma])$:



ALL TOGETHER NOW

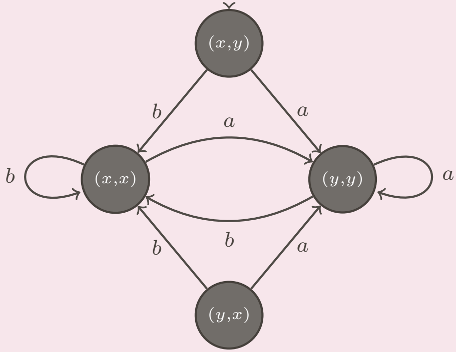
Example



ALL TOGETHER NOW

1st Step. Construct the product automaton.

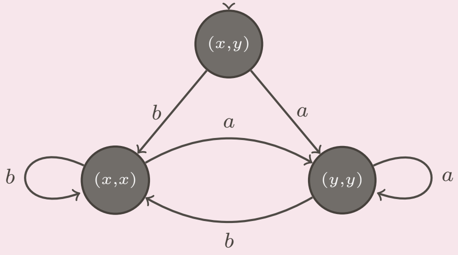
Example



ALL TOGETHER NOW

2nd Step. Take the image under the reachability map $r_{\overline{x}}$

Example



ALL TOGETHER NOW

We define $\mathsf{Eq}(X, \alpha)$ as $\ker(r_{\overline{x}})$.

Example

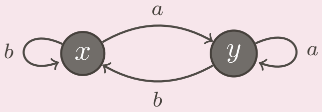
$$\mathsf{Eq}(X, \alpha) = \{aa = a, bb = b, ab = b, ba = a\}$$

ALL TOGETHER NOW

We define $\mathsf{Eq}(X, \alpha)$ as $\ker(r_{\overline{x}})$.

Example

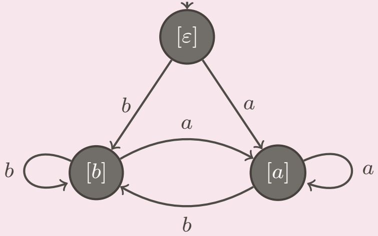
$$\mathsf{Eq}(X, \alpha) = \{aa = a, bb = b, ab = b, ba = a\}$$



ALL TOGETHER NOW

free(X, α) is the quotient automaton of A^* over Eq(X, α).

Example



$$\text{free}(X, \alpha) = A^* / \text{Eq}(X, \alpha)$$

COFREE

Dually, let $X = \{x_1, \ldots, x_n\}$ be the set of states of a finite automaton (X, α). We define a coloured automaton cofree(X, α) in two steps, as follows:

COFREE

Dually, let $X = \{x_1, \ldots, x_n\}$ be the set of states of a finite automaton (X, α). We define a coloured automaton cofree(X, α) in two steps, as follows:

(i) First, we take the coproduct of the 2^n coloured automata (X, c, α) that we obtain by letting c range over the set $X \to 2$ of all colouring functions. This yields a coloured automaton $(\Sigma X, \hat{c}, \hat{\alpha})$ with

$$\Sigma X = \sum_{c:X\to 2} X_c$$

(where $X_c = X$), and with $\hat{c}$ and $\hat{\alpha}$ defined component-wise.

COFREE

(ii) Next we consider the observability map $o_{\hat{c}} : \Sigma X \to 2^{A^*}$ and define:

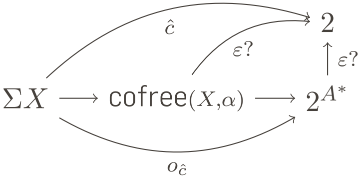
$$\mathsf{coEq}(X, \alpha) = \mathsf{im}(o_{\hat{c}}) \qquad \mathsf{cofree}(X, \alpha) = \mathsf{coEq}(X, \alpha)$$

COFREE

(ii) Next we consider the observability map $o_{\hat{c}} : \Sigma X \to 2^{A^*}$ and define:

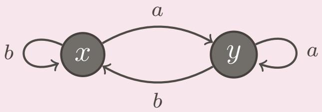
$$\text{coEq}(X, \alpha) = \text{im}(o_{\hat{c}}) \qquad \text{cofree}(X, \alpha) = \text{coEq}(X, \alpha)$$

This yields the coloured automaton (cofree$(X, \alpha), \varepsilon?, \tau$):



ALL TOGETHER NOW

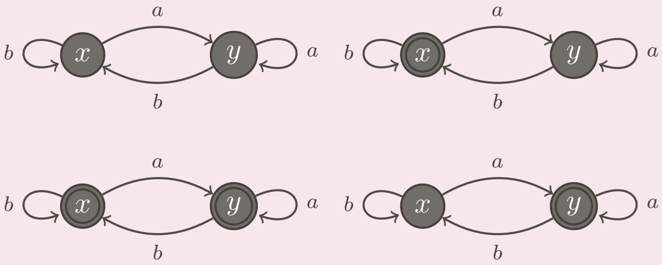
Example



ALL TOGETHER NOW

1st Step. Construct the coproduct automaton.

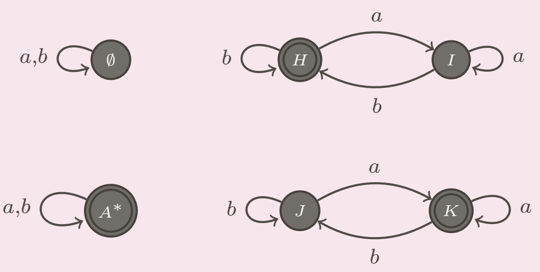
Example



ALL TOGETHER NOW

2nd Step. Take the image under the observability map $o_{\hat{c}}$.

Example



$H = (a^* b)^*$

$I = (a^* b)^+$

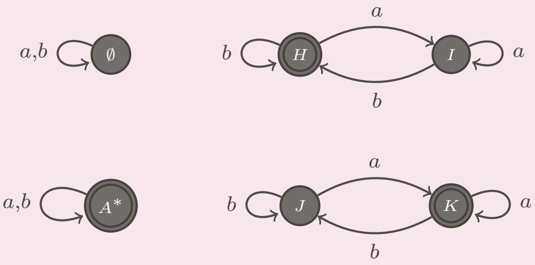
$J = (b^* a)^+$

$K = (b^* a)^*$

ALL TOGETHER NOW

We define $\text{coEq}(X, \alpha)$ as $\text{im}(o_{\hat{c}})$ and $\text{cofree}(X, \alpha) = \text{coEq}(X, \alpha)$.

Example



$H = (a^* b)^*$
$I = (a^* b)^+$

$J = (b^* a)^+$
$K = (b^* a)^*$

$$\text{cofree}(X, \alpha) = \text{coEq}(X, \alpha)$$

A DUAL EQUIVALENCE

In this section, we shall first show that -- when suitably restricted -- the constructions of free and cofree are in fact functorial, that is, they act not only on automata but also on homomorphisms.

A DUAL EQUIVALENCE

In this section, we shall first show that -- when suitably restricted -- the constructions of free and cofree are in fact functorial, that is, they act not only on automata but also on homomorphisms.

We will be using the following categories:

- $\mathcal{A}$: the category of automata (X, α) and automata homomorphisms
- $\mathcal{A}_m$: the category of automata (X, α) and automata monomorphisms
- $\mathcal{A}_e$: the category of automata (X, α) and automata epimorphisms

FUNCTORIAL FREE

As it turns out, we can extend the definition of free to monomorphisms, such that we obtain functors of the following type:

$$\text{free} : \mathcal{A}_m \to (\mathcal{A}_e)^{\text{op}}$$

Here the superscript op indicates a reversal of arrows.

FUNCTORIAL FREE

As it turns out, we can extend the definition of free to monomorphisms, such that we obtain functors of the following type:

$$\text{free} : \mathcal{A}_m \to (\mathcal{A}_e)^{\text{op}}$$

Here the superscript op indicates a reversal of arrows.

For monomorphisms,

$$\begin{array}{ccc} (X, \alpha) & & \text{free}(X, \alpha) \\ m \big\downarrow & \overset{\text{free}}{\longmapsto} & \big\uparrow \text{free}(m) \\ (Y, \beta) & & \text{free}(Y, \beta) \end{array}$$

where free(m) is defined simply by quotienting. Recall that the existence of the monomorphism m implies Eq$(Y, \beta) \subseteq$ Eq(X, α).

FUNCTORIAL COFREE

Dually, we can extend the definition of cofree to epimorphisms, such that we obtain functors of the following type:

$$\text{cofree} : \mathcal{A}_e \to (\mathcal{A}_m)^{\text{op}}$$

FUNCTORIAL COFREE

Dually, we can extend the definition of cofree to epimorphisms, such that we obtain functors of the following type:

$$\text{cofree} : \mathcal{A}_e \to (\mathcal{A}_m)^{\text{op}}$$

For epimorphisms,

$$\begin{array}{ccc} (X, \alpha) & & \text{cofree}(X, \alpha) \\ e \Big\downarrow & \overset{\text{cofree}}{\longmapsto} & \Big\uparrow \text{cofree}(e) \\ (Y, \beta) & & \text{cofree}(Y, \beta) \end{array}$$

where cofree(e) is just set inclusion. Recall that the existence of the epimorphism e implies coEq$(Y, \beta) \subseteq$ coEq(X, α).

CONGRUENCE QUOTIENTS

Definition

We introduce the category $\mathcal{C}$ of congruence quotients, which is defined as follows:

$$\begin{aligned} \text{objects}(\mathcal{C}) &= \{ (A^*/C, [\sigma]) \mid C \text{ is a congruence relation } \} \\ \text{arrows}(\mathcal{C}) &= \{ e : A^*/C \to A^*/D \mid e \text{ is an epimorphism} \} \end{aligned}$$

CONGRUENCE QUOTIENTS

Definition

We introduce the category $\mathcal{C}$ of congruence quotients, which is defined as follows:

$$\begin{aligned} \text{objects}(\mathcal{C}) &= \{\, (A^*/C, [\sigma]) \mid C \text{ is a congruence relation } \} \\ \text{arrows}(\mathcal{C}) &= \{\, e : A^*/C \to A^*/D \mid e \text{ is an epimorphism} \,\} \end{aligned}$$

Theorem

$\text{free}(\mathcal{A}_m) = \mathcal{C}^{\text{op}}$

VARIETIES OF LANGUAGES

Definition

A variety of languages, is a set $V \subseteq 2^{A^*}$ such that:

(i) V is a complete atomic Boolean subalgebra of 2^{A^*}.

(ii) if $L \in V$ then for all $a \in A$, both L_a and ${}_aL \in V$

VARIETY OF LANGUAGES

Definition

We introduce the category $\mathcal{V}$ of varieties of languages, which is defined as follows:

$$\begin{aligned} \text{objects}(\mathcal{V}) &= \{ (V, \tau) \mid V \text{ is a variety of languages } \} \\ \text{arrows}(\mathcal{V}) &= \{ m : V \to W \mid m \text{ is an monomorphism} \} \end{aligned}$$

VARIETY OF LANGUAGES

Definition

We introduce the category $\mathcal{V}$ of varieties of languages, which is defined as follows:

$$\begin{array}{lcl} \text{objects}(\mathcal{V}) & = & \{\, (V, \tau) \mid V \text{ is a variety of languages } \} \\ \text{arrows}(\mathcal{V}) & = & \{\, m : V \to W \mid m \text{ is an monomorphism} \,\} \end{array}$$

Theorem

$\text{cofree}(\mathcal{C}) = \mathcal{V}^{\text{op}}$

MAIN THEOREM

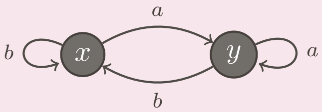
Our main result is a dual equivalence.

Theorem

cofree: $\mathcal{C} \cong \mathcal{V}^{\mathrm{op}}$: free

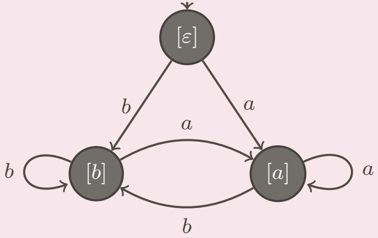
ILLUSTRATING THE DUALITY

Example



ILLUSTRATING THE DUALITY

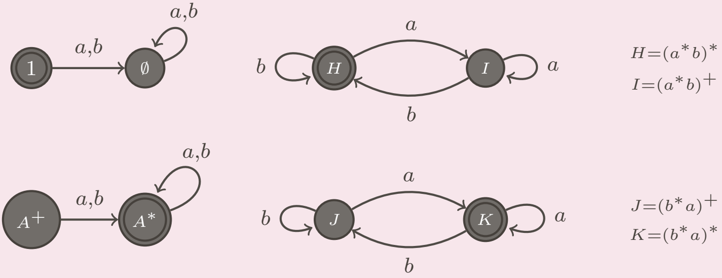
Example



$\mathsf{free}(X, \alpha)$

ILLUSTRATING THE DUALITY

Example



cofree ∘ free(X, α)

ILLUSTRATING THE DUALITY

To better understand the duality, consider a quotient automaton $(A^*/C, [\sigma])$. For a word $w \in A^*$, consider the following colouring:

$$\begin{array}{rccl} \delta_{[w]} : & A^*/C & \longrightarrow & 2 \\ & [v] & \longmapsto & \begin{cases} 1 & \text{if } [v] = [w] \\ 0 & \text{otherwise} \end{cases} \end{array}$$

ILLUSTRATING THE DUALITY

To better understand the duality, consider a quotient automaton $(A^*/C, [\sigma])$. For a word $w \in A^*$, consider the following colouring:

$$\begin{array}{rccl} \delta_{[w]} : & A^*/C & \longrightarrow & 2 \\ & [v] & \longmapsto & \begin{cases} 1 & \text{if } [v] = [w] \\ 0 & \text{otherwise} \end{cases} \end{array}$$

Under this colouring, it holds:

$$o_{\delta_{[w]}}([\varepsilon]) = [w]$$

ILLUSTRATING THE DUALITY

To better understand the duality, consider a quotient automaton $(A^*/C, [\sigma])$. For a word $w \in A^*$, consider the following colouring:

$$\begin{array}{rccl} \delta_{[w]} : & A^*/C & \longrightarrow & 2 \\ & [v] & \longmapsto & \begin{cases} 1 & \text{if } [v] = [w] \\ 0 & \text{otherwise} \end{cases} \end{array}$$

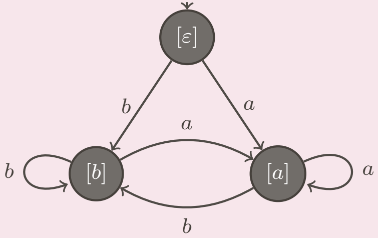
Under this colouring, it holds:

$$o_{\delta_{[w]}}([\varepsilon]) = [w]$$

Hence, every state $[w] \in A^*/C$ also belongs to cofree$(A^*/C, [\sigma])$.

ILLUSTRATING THE DUALITY

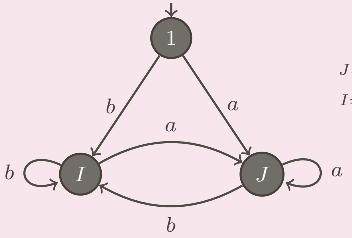
Example



$\text{free}(X, \alpha)$

ILLUSTRATING THE DUALITY

Example



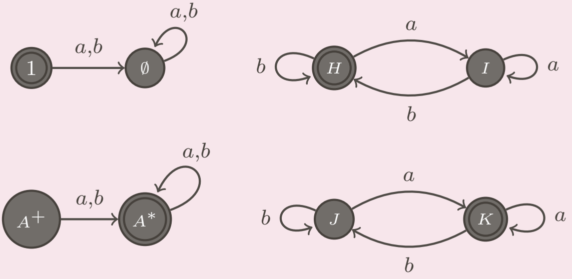
$J = (b^* a)^+$

$I = (a^* b)^+$

$\mathsf{free}(X, \alpha)$

ILLUSTRATING THE DUALITY

Example



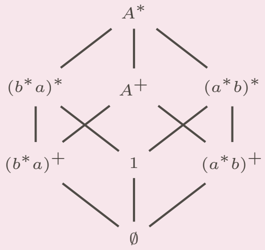
$H = (a^* b)^*$
$I = (a^* b)^+$

$J = (b^* a)^+$
$K = (b^* a)^*$

$$\text{cofree} \circ \text{free}(X, \alpha)$$

ILLUSTRATING THE DUALITY

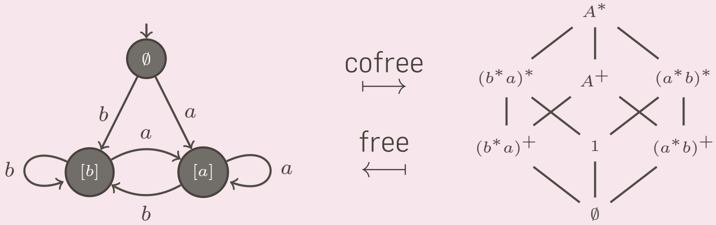
Example



$\text{cofree} \circ \text{free}(X, \alpha)$

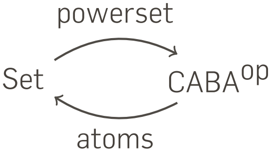
ILLUSTRATING THE DUALITY

Example

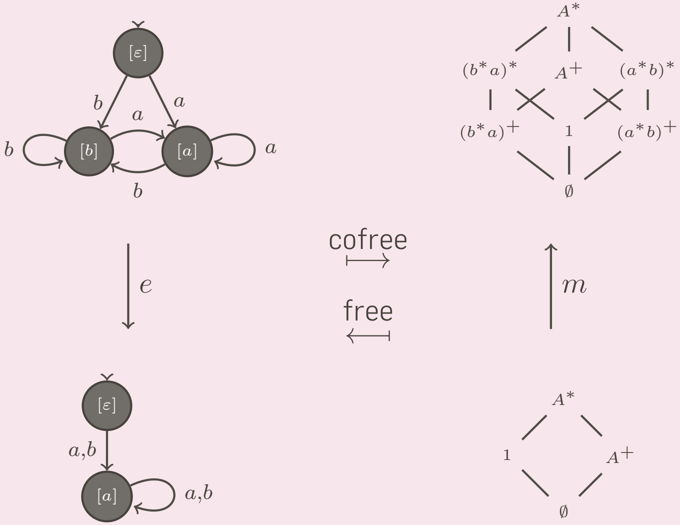


ILLUSTRATING THE DUALITY

That is, forgetting all the automata structure we recover the classical duality:

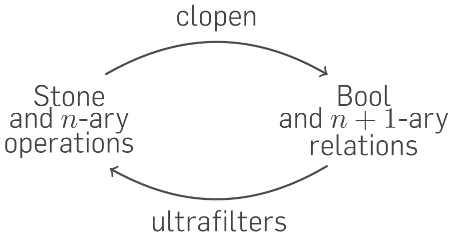


Example



RELATED WORK

In the literature we have found examples of such duality [GGP08 Geh11, Rou11]. They start directly from the extended Stone duality:



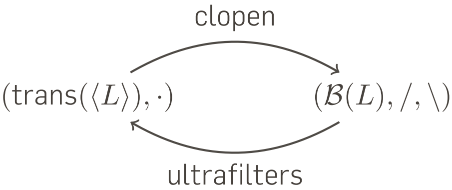
RELATED WORK

For a regular language L. They state the duality:

$$(\text{trans}(\langle L \rangle), \cdot) \underset{\text{ultrafilters}}{\overset{\text{clopen}}{\rightleftarrows}} (\mathcal{B}(L), /, \backslash)$$

RELATED WORK

For a regular language L. They state the duality:



Where $(\text{trans}(\langle L \rangle), \cdot)$ is the transition monoid of $\langle L \rangle$ and $(\mathcal{B}(L), /, \backslash)$ is a boolean algebra with some residuation properties.

RELATED WORK

Theorem

$\text{free}(X, \alpha) \cong \text{trans}(X, \alpha)$

RELATED WORK

Theorem

$\text{free}(X, \alpha) \cong \text{trans}(X, \alpha)$

Proposition

Every variety of languages is a boolean algebra with residuation properties (as in [GGP08]).

RELATED WORK

Theorem

$\text{free}(X, \alpha) \cong \text{trans}(X, \alpha)$

Proposition

Every variety of languages is a boolean algebra with residuation properties (as in [GGP08]). In the finite case, both definitions coincide.

RELATED WORK

Theorem

free$(X, \alpha) \cong$ trans(X, α)

Proposition

Every variety of languages is a boolean algebra with residuation properties (as in [GGP08]). In the finite case, both definitions coincide.

For finite automata, our duality coincides with that of Gehrke et al.

COMMUTATIVE LANGUAGES

Example

Let $A = \{a, b\}$ and let ab=ba denote the smallest congruence on A^* containing the equation (ab, ba).

COMMUTATIVE LANGUAGES

Example

Let $A = \{a, b\}$ and let ab=ba denote the smallest congruence on A^* containing the equation (ab, ba). It is easy to prove that, for all $v, w \in A^*$,

$$(v, w) \in \text{ab=ba} \;\Leftrightarrow\; |v|_a = |w|_a \text{ and } |v|_b = |w|_b$$

Languages $[w]$ in the congruence quotient $A^*/\text{ab=ba}$ satisfy

$$[w] = \{\, v \in A^* \mid v \text{ is a permutation of } w \,\}$$

COMMUTATIVE LANGUAGES

Example

Let $A = \{a, b\}$ and let ab=ba denote the smallest congruence on A^* containing the equation (ab, ba). It is easy to prove that, for all $v, w \in A^*$,

$$(v, w) \in \text{ab=ba} \;\Leftrightarrow\; |v|_a = |w|_a \text{ and } |v|_b = |w|_b$$

Languages $[w]$ in the congruence quotient $A^*/\text{ab=ba}$ satisfy

$$[w] = \{\, v \in A^* \mid v \text{ is a permutation of } w \,\}$$

By the duality Theorem, we have that $V = \text{cofree}(A^*/\text{ab=ba})$ is a variety of languages.

COMMUTATIVE LANGUAGES

Example

We now call a language L commutative whenever $L \in V$.

COMMUTATIVE LANGUAGES

Example

We now call a language L commutative whenever $L \in V$. This terminology is justified by the following equivalences:

$L \in V$ $\Leftrightarrow$ L is the union of permutation equivalence classes $[w]$ $\Leftrightarrow$ $\langle L \rangle \models$ ab=ba

COMMUTATIVE LANGUAGES

Example

We now call a language L commutative whenever $L \in V$. This terminology is justified by the following equivalences:

$L \in V$ ⇔ L is the union of permutation equivalence classes $[w]$ ⇔ $\langle L \rangle \models$ ab=ba

Corollary

$L \in \text{coEq}(A^*/C, [\sigma]) \;\Leftrightarrow\; C \subseteq \text{Eq}(\langle L \rangle, \tau)$

EQUATIONAL BISIMULATIONS

Definition

Let $C \subseteq A^* \times A^*$ be a congruence. We call a relation $R \subseteq 2^{A^*} \times 2^{A^*}$ a C-bisimulation if for all $(K, L) \in R$,

1. $\varepsilon \in K \Leftrightarrow \varepsilon \in L$
2. $\forall (v, w) \in C,\ (K_v, L_w) \in R$

EQUATIONAL BISIMULATIONS

Definition

Let $C \subseteq A^* \times A^*$ be a congruence. We call a relation $R \subseteq 2^{A^*} \times 2^{A^*}$ a C-bisimulation if for all $(K, L) \in R$,

1. $\varepsilon \in K \Leftrightarrow \varepsilon \in L$
2. $\forall (v, w) \in C, \; (K_v, L_w) \in R$

Proposition

If R is a C-bisimulation and $(K, L) \in R$ then,

i. $K = L$
ii. $\langle K \rangle \models C$

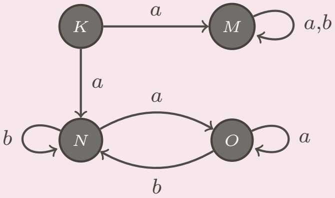
COMMUTATIVE LANGUAGES

Example

Let $K = aA^* + b(a^*b)^* + b(b^*a)^+$. We shall use last proposition to show that K is commutative. Referring to the example of commutative languages, we need to prove that $\langle K \rangle \models$ ab=ba.

COMMUTATIVE LANGUAGES

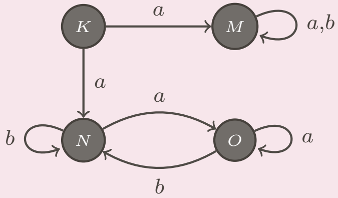
Example



$$K = aA^* + b(a^*b)^* + b(b^*a)^+$$
$$M = A^*$$
$$N = (a^*b)^* + (b^*a)^+$$
$$O = (a^*b)^+ + (b^*a)^*$$

COMMUTATIVE LANGUAGES

Example



$$K = aA^* + b(a^*b)^* + b(b^*a)^+$$
$$M = A^*$$
$$N = (a^*b)^* + (b^*a)^+$$
$$O = (a^*b)^+ + (b^*a)^*$$

Let

$$R = \{(K,K)\} \cup \{M,N,O\}^2$$

Then R is an (ab=ba)-bisimulation. Thus $\langle K \rangle \models$ ab=ba.

BIBLIOGRAPHY

M. Gehrke,
Duality and recognition,
MFCS, volume 6907 of LNCS, pages 3-18, 2011.

M. Gehrke, S. Grigorieff, J.-E. Pin,
Duality and equational theory of regular languages,
In Proceedings of ICALP, pages 246-257, 2008.

J.J.M.M. Rutten, A. Ballester-Bolinches, E. Cosme-Llópez,
Varieties and covarieties of languages (preliminary version),
In Proceedings of MFPS XXIX, pages 7-28, 2013.

F. Roumen,
Canonical automata via duality,
Master's thesis, Radboud Universiteit Nijmegen, 2011.

HSRM Beamer Theme, by Benjamin Weiss.