On Boolean closed full trios and rational Kripke frames

Markus Lohrey[1] Dietrich Kuske[2], Georg Zetzsche[3]

[1]Department für Elektrotechnik und Informatik, Universität Siegen

[2]Institut für Theoretische Informatik, Technische Universität Ilmenau

[3]LSV, CNRS & ENS Cachan, Université Paris-Saclay

INFINI Seminar, 31.05.2016

Closure properties

Common closure properties

- Homomorphism: $h : \Sigma^* \to \Gamma^*$, replaces letters by words
- Inverse homomorphism: $\{w \in \Sigma^* \mid h(w) \in L\}$
- Intersection with regular sets.
- Boolean operations: union, intersection, complementation.

Closure properties

Common closure properties

- Homomorphism: $h : \Sigma^* \to \Gamma^*$, replaces letters by words
- Inverse homomorphism: $\{w \in \Sigma^* \mid h(w) \in L\}$
- Intersection with regular sets.
- Boolean operations: union, intersection, complementation.

Definition

Language class $\mathcal{C}$ is a *full trio*, if it is closed under the first three above.

Closure properties

Common closure properties

- Homomorphism: $h : \Sigma^* \to \Gamma^*$, replaces letters by words
- Inverse homomorphism: $\{w \in \Sigma^* \mid h(w) \in L\}$
- Intersection with regular sets.
- Boolean operations: union, intersection, complementation.

Definition

Language class $\mathcal{C}$ is a *full trio*, if it is closed under the first three above.

Examples

REG, CF, LIN, Petri net languages, blind multicounter languages, classes of various grammar types, etc.

Observation

The regular languages constitute a Boolean closed full trio.

Observation

The regular languages constitute a Boolean closed full trio.

Boolean closed full trios

Are there Boolean closed full trios beyond REG?

Observation

The regular languages constitute a Boolean closed full trio.

Boolean closed full trios

Are there Boolean closed full trios beyond REG?

- Automatic structures beyond regular languages

Observation

The regular languages constitute a Boolean closed full trio.

Boolean closed full trios

Are there Boolean closed full trios beyond REG?

- Automatic structures beyond regular languages
- Complementation closure for union closed full trios

$\mathrm{RE}(\mathcal{C})$: Accepted by Turing machine with oracle $L \in \mathcal{C}$.

Definition

Arithmetical hierarchy:

$$\Sigma_0 = \mathrm{REC}, \qquad \Sigma_{n+1} = \mathrm{RE}(\Sigma_n) \text{ for } n \geqslant 0, \qquad \mathrm{AH} = \bigcup_{n \geqslant 0} \Sigma_n.$$

$\mathrm{RE}(\mathcal{C})$: Accepted by Turing machine with oracle $L \in \mathcal{C}$.

Definition

Arithmetical hierarchy:

$$\Sigma_0 = \mathrm{REC}, \qquad \Sigma_{n+1} = \mathrm{RE}(\Sigma_n) \text{ for } n \geqslant 0, \qquad \mathrm{AH} = \bigcup_{n \geqslant 0} \Sigma_n.$$

Relative arithmetical hierarchy:

$$\Sigma_0(L) = \mathrm{REC}(L), \quad \Sigma_{n+1}(L) = \mathrm{RE}(\Sigma_n(L)) \text{ for } n \geqslant 0, \quad \mathrm{AH}(L) = \bigcup_{n \geqslant 0} \Sigma_n(L).$$

$\mathsf{RE}(\mathcal{C})$: Accepted by Turing machine with oracle $L \in \mathcal{C}$.

Definition

Arithmetical hierarchy:

$$\Sigma_0 = \mathsf{REC}, \qquad \Sigma_{n+1} = \mathsf{RE}(\Sigma_n) \text{ for } n \geqslant 0, \qquad \mathsf{AH} = \bigcup_{n \geqslant 0} \Sigma_n.$$

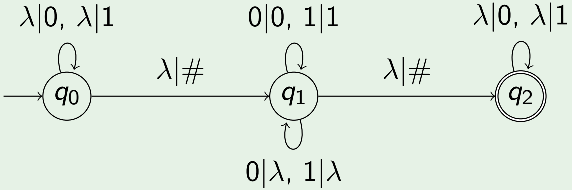
Relative arithmetical hierarchy:

$$\Sigma_0(L) = \mathsf{REC}(L), \quad \Sigma_{n+1}(L) = \mathsf{RE}(\Sigma_n(L)) \text{ for } n \geqslant 0, \quad \mathsf{AH}(L) = \bigcup_{n \geqslant 0} \Sigma_n(L).$$

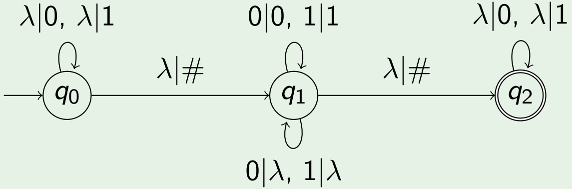
Theorem

Let $\mathcal{T}$ be a Boolean closed full trio. If $\mathcal{T}$ contains any non-regular language L, then $\mathcal{T}$ includes $\mathsf{AH}(L)$.

Example (Transducer)

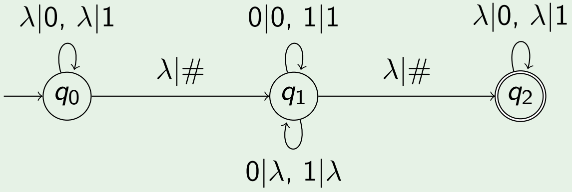


Example (Transducer)



$$T(A) = \{(x, u\#v\#w) \mid u, v, w, x \in \{0,1\}^*,\ v \preceq x\}$$

Example (Transducer)



$$T(A) = \{(x, u\#v\#w) \mid u, v, w, x \in \{0,1\}^*,\ v \preceq x\}$$

Definition

- *Rational transduction*: set of pairs given by a finite state transducer.
- For rational transduction $T \subseteq X^* \times Y^*$ and language $L \subseteq Y^*$, let

$$TL = \{y \in X^* \mid \exists x \in L : (x, y) \in T\}$$

Theorem (Nivat 1968)

A language class is a full trio iff it is closed under rational transductions.

Proof I

Let $\Delta = \{+, -, z\}$: increment, decrement, and zero test.

Proof I

Let $\Delta = \{+, -, z\}$: increment, decrement, and zero test.

Definition

Let $C \subseteq \Delta^*$ be the set of words $\delta_1 \cdots \delta_m$, $\delta_1, \ldots, \delta_m \in \Delta$, for which there are numbers $x_0, \ldots, x_m \in \mathbb{N}$ such that for $1 \leqslant i \leqslant m$:

Proof I

Let $\Delta = \{+, -, z\}$: increment, decrement, and zero test.

Definition

Let $C \subseteq \Delta^*$ be the set of words $\delta_1 \cdots \delta_m$, $\delta_1, \ldots, \delta_m \in \Delta$, for which there are numbers $x_0, \ldots, x_m \in \mathbb{N}$ such that for $1 \leqslant i \leqslant m$:

1. if $\delta_i = +$, then $x_i = x_{i-1} + 1$,

Proof I

Let $\Delta = \{+, -, z\}$: increment, decrement, and zero test.

Definition

Let $C \subseteq \Delta^*$ be the set of words $\delta_1 \cdots \delta_m$, $\delta_1, \ldots, \delta_m \in \Delta$, for which there are numbers $x_0, \ldots, x_m \in \mathbb{N}$ such that for $1 \leqslant i \leqslant m$:

1. if $\delta_i = +$, then $x_i = x_{i-1} + 1$,
2. if $\delta_i = -$, then $x_i = x_{i-1} - 1$, and

Proof I

Let $\Delta = \{+, -, z\}$: increment, decrement, and zero test.

Definition

Let $C \subseteq \Delta^*$ be the set of words $\delta_1 \cdots \delta_m$, $\delta_1, \ldots, \delta_m \in \Delta$, for which there are numbers $x_0, \ldots, x_m \in \mathbb{N}$ such that for $1 \leqslant i \leqslant m$:

1. if $\delta_i = +$, then $x_i = x_{i-1} + 1$,
2. if $\delta_i = -$, then $x_i = x_{i-1} - 1$, and
3. if $\delta_i = z$, then $x_i = x_{i-1} = 0$.

Proof I

Let $\Delta = \{+, -, z\}$: increment, decrement, and zero test.

Definition

Let $C \subseteq \Delta^*$ be the set of words $\delta_1 \cdots \delta_m$, $\delta_1, \ldots, \delta_m \in \Delta$, for which there are numbers $x_0, \ldots, x_m \in \mathbb{N}$ such that for $1 \leqslant i \leqslant m$:

1. if $\delta_i = +$, then $x_i = x_{i-1} + 1$,
2. if $\delta_i = -$, then $x_i = x_{i-1} - 1$, and
3. if $\delta_i = z$, then $x_i = x_{i-1} = 0$.

Crucial step: If $\mathcal{T}$ contains non-regular L, then $\mathcal{T}$ contains C.

Proof I

Let $\Delta = \{+, -, z\}$: increment, decrement, and zero test.

Definition

Let $C \subseteq \Delta^*$ be the set of words $\delta_1 \cdots \delta_m$, $\delta_1, \ldots, \delta_m \in \Delta$, for which there are numbers $x_0, \ldots, x_m \in \mathbb{N}$ such that for $1 \leqslant i \leqslant m$:

1. if $\delta_i = +$, then $x_i = x_{i-1} + 1$,
2. if $\delta_i = -$, then $x_i = x_{i-1} - 1$, and
3. if $\delta_i = z$, then $x_i = x_{i-1} = 0$.

Crucial step: If $\mathcal{T}$ contains non-regular L, then $\mathcal{T}$ contains C.

Definition

$u \equiv_L v$: for each $w \in X^*$, $uw \in L$ iff $vw \in L$.

Proof I

Let $\Delta = \{+, -, z\}$: increment, decrement, and zero test.

Definition

Let $C \subseteq \Delta^*$ be the set of words $\delta_1 \cdots \delta_m$, $\delta_1, \ldots, \delta_m \in \Delta$, for which there are numbers $x_0, \ldots, x_m \in \mathbb{N}$ such that for $1 \leqslant i \leqslant m$:

1. if $\delta_i = +$, then $x_i = x_{i-1} + 1$,
2. if $\delta_i = -$, then $x_i = x_{i-1} - 1$, and
3. if $\delta_i = z$, then $x_i = x_{i-1} = 0$.

Crucial step: If $\mathcal{T}$ contains non-regular L, then $\mathcal{T}$ contains C.

Definition

$u \equiv_L v$: for each $w \in X^*$, $uw \in L$ iff $vw \in L$.

Theorem (Myhill-Nerode)

L is regular if and only if $\equiv_L$ has finite index.

Proof II

Idea: In order to obtain C, construct $\hat{C}_L$:

Definition

Let $\hat{C}_L$ be the set of all words

$$v_0\delta_1 v_1 \cdots \delta_m v_m \# u_0 \# \cdots u_n \#$$

with $\delta_i \in \Delta$, $v_i \in X^*$, $u_j \in X^*$

Proof II

Idea: In order to obtain C, construct $\hat{C}_L$:

Definition

Let $\hat{C}_L$ be the set of all words

$$v_0\delta_1 v_1 \cdots \delta_m v_m \# u_0 \# \cdots u_n \#$$

with $\delta_i \in \Delta$, $v_i \in X^*$, $u_j \in X^*$, such that $u_k \not\equiv_L u_\ell$ $(k \neq \ell)$

Proof II

Idea: In order to obtain C, construct $\hat{C}_L$:

Definition

Let $\hat{C}_L$ be the set of all words

$$v_0\delta_1 v_1 \cdots \delta_m v_m \# u_0 \# \cdots u_n \#$$

with $\delta_i \in \Delta$, $v_i \in X^*$, $u_j \in X^*$, such that $u_k \not\equiv_L u_\ell$ $(k \neq \ell)$ and for each $1 \leqslant i \leqslant m$ there is a $1 \leqslant j \leqslant n$ with

Proof II

Idea: In order to obtain C, construct $\hat{C}_L$:

Definition

Let $\hat{C}_L$ be the set of all words

$$v_0 \delta_1 v_1 \cdots \delta_m v_m \# u_0 \# \cdots u_n \#$$

with $\delta_i \in \Delta$, $v_i \in X^*$, $u_j \in X^*$, such that $u_k \not\equiv_L u_\ell$ ($k \neq \ell$) and for each $1 \leqslant i \leqslant m$ there is a $1 \leqslant j \leqslant n$ with

- if $\delta_i = +$, then $v_{i-1} \equiv_L u_{j-1}$, $v_i \equiv_L u_j$

Proof II

Idea: In order to obtain C, construct $\hat{C}_L$:

Definition

Let $\hat{C}_L$ be the set of all words

$$v_0 \delta_1 v_1 \cdots \delta_m v_m \# u_0 \# \cdots u_n \#$$

with $\delta_i \in \Delta$, $v_i \in X^*$, $u_j \in X^*$, such that $u_k \not\equiv_L u_\ell$ $(k \neq \ell)$ and for each $1 \leqslant i \leqslant m$ there is a $1 \leqslant j \leqslant n$ with

- if $\delta_i = +$, then $v_{i-1} \equiv_L u_{j-1}$, $v_i \equiv_L u_j$
- if $\delta_i = -$, then $v_{i-1} \equiv_L u_j$, $v_i \equiv_L u_{j-1}$

Proof II

Idea: In order to obtain C, construct $\hat{C}_L$:

Definition

Let $\hat{C}_L$ be the set of all words

$$v_0\delta_1 v_1 \cdots \delta_m v_m \# u_0 \# \cdots u_n \#$$

with $\delta_i \in \Delta$, $v_i \in X^*$, $u_j \in X^*$, such that $u_k \not\equiv_L u_\ell$ $(k \neq \ell)$ and for each $1 \leqslant i \leqslant m$ there is a $1 \leqslant j \leqslant n$ with

- if $\delta_i = +$, then $v_{i-1} \equiv_L u_{j-1}$, $v_i \equiv_L u_j$
- if $\delta_i = -$, then $v_{i-1} \equiv_L u_j$, $v_i \equiv_L u_{j-1}$
- if $\delta_i = z$, then $v_{i-1} \equiv_L v_i \equiv_L u_j \equiv_L u_0$.

Proof II

Idea: In order to obtain C, construct $\hat{C}_L$:

Definition

Let $\hat{C}_L$ be the set of all words

$$v_0\delta_1 v_1 \cdots \delta_m v_m \# u_0 \# \cdots u_n \#$$

with $\delta_i \in \Delta$, $v_i \in X^*$, $u_j \in X^*$, such that $u_k \not\equiv_L u_\ell$ $(k \neq \ell)$ and for each $1 \leqslant i \leqslant m$ there is a $1 \leqslant j \leqslant n$ with

- if $\delta_i = +$, then $v_{i-1} \equiv_L u_{j-1}$, $v_i \equiv_L u_j$
- if $\delta_i = -$, then $v_{i-1} \equiv_L u_j$, $v_i \equiv_L u_{j-1}$
- if $\delta_i = z$, then $v_{i-1} \equiv_L v_i \equiv_L u_j \equiv_L u_0$.

Observation

If L is non-regular, C can be obtained from $\hat{C}_L$.

Proof III

$$W_1 = \{u\#v\#w \mid u, v, w \in X^*,\ uw \in L\},$$
$$W_2 = \{u\#v\#w \mid u, v, w \in X^*,\ vw \in L\}.$$

Proof III

$$W_1 = \{u\#v\#w \mid u, v, w \in X^*,\ uw \in L\},$$
$$W_2 = \{u\#v\#w \mid u, v, w \in X^*,\ vw \in L\}.$$

$$W' = \{u\#v\#w \mid u, v, w \in X^*, (uw \in L, vw \notin L) \text{ or } (uw \notin L, vw \in L)\}$$

Proof III

$$W_1 = \{u\#v\#w \mid u, v, w \in X^*,\ uw \in L\},$$
$$W_2 = \{u\#v\#w \mid u, v, w \in X^*,\ vw \in L\}.$$

$$W' = \{u\#v\#w \mid u, v, w \in X^*, (uw \in L, vw \notin L) \text{ or } (uw \notin L, vw \in L)\}$$
$$= (W_1 \cap \overline{W_2}) \cup (\overline{W_1} \cap W_2)$$

Proof III

$$W_1 = \{u\#v\#w \mid u, v, w \in X^*,\ uw \in L\},$$
$$W_2 = \{u\#v\#w \mid u, v, w \in X^*,\ vw \in L\}.$$

$$W' = \{u\#v\#w \mid u, v, w \in X^*, (uw \in L, vw \notin L) \text{ or } (uw \notin L, vw \in L)\}$$
$$= (W_1 \cap \overline{W_2}) \cup (\overline{W_1} \cap W_2)$$

$$W = \{u\#v \mid u, v \in X^*, u \not\equiv_L v\}$$

Proof III

$$W_1 = \{u\#v\#w \mid u, v, w \in X^*,\ uw \in L\},$$
$$W_2 = \{u\#v\#w \mid u, v, w \in X^*,\ vw \in L\}.$$

$$W' = \{u\#v\#w \mid u, v, w \in X^*, (uw \in L, vw \notin L) \text{ or } (uw \notin L, vw \in L)\}$$
$$= (W_1 \cap \overline{W_2}) \cup (\overline{W_1} \cap W_2)$$

$$W = \{u\#v \mid u, v \in X^*, u \not\equiv_L v\} = \{u\#v \mid u\#v\#w \in W' \text{ for some } w \in X^*\}$$

Proof III

$$W_1 = \{u\#v\#w \mid u, v, w \in X^*,\ uw \in L\},$$
$$W_2 = \{u\#v\#w \mid u, v, w \in X^*,\ vw \in L\}.$$

$$W' = \{u\#v\#w \mid u, v, w \in X^*, (uw \in L, vw \notin L) \text{ or } (uw \notin L, vw \in L)\}$$
$$= (W_1 \cap \overline{W_2}) \cup (\overline{W_1} \cap W_2)$$

$$W = \{u\#v \mid u, v \in X^*, u \not\equiv_L v\} = \{u\#v \mid u\#v\#w \in W' \text{ for some } w \in X^*\}$$

$$P = \{u\#v \mid u \equiv_L v\}$$

Proof III

$$W_1 = \{u\#v\#w \mid u, v, w \in X^*,\ uw \in L\},$$
$$W_2 = \{u\#v\#w \mid u, v, w \in X^*,\ vw \in L\}.$$

$$W' = \{u\#v\#w \mid u, v, w \in X^*, (uw \in L, vw \notin L) \text{ or } (uw \notin L, vw \in L)\}$$
$$= (W_1 \cap \overline{W_2}) \cup (\overline{W_1} \cap W_2)$$

$$W = \{u\#v \mid u, v \in X^*, u \not\equiv_L v\} = \{u\#v \mid u\#v\#w \in W' \text{ for some } w \in X^*\}$$

$$P = \{u\#v \mid u \equiv_L v\} = X^*\#X^* \setminus W$$

Proof III

$$W_1 = \{u\#v\#w \mid u, v, w \in X^*,\ uw \in L\},$$
$$W_2 = \{u\#v\#w \mid u, v, w \in X^*,\ vw \in L\}.$$

$$\begin{aligned} W' &= \{u\#v\#w \mid u, v, w \in X^*, (uw \in L, vw \notin L) \text{ or } (uw \notin L, vw \in L)\} \\ &= (W_1 \cap \overline{W_2}) \cup (\overline{W_1} \cap W_2) \end{aligned}$$

$$W = \{u\#v \mid u, v \in X^*, u \not\equiv_L v\} = \{u\#v \mid u\#v\#w \in W' \text{ for some } w \in X^*\}$$

$$P = \{u\#v \mid u \equiv_L v\} = X^*\#X^* \setminus W$$

$$S = \{u_0\#u_1\# \cdots u_n\# \mid u_i \not\equiv_L u_j \text{ for all } i \neq j\}$$

Proof III

$$W_1 = \{u\#v\#w \mid u, v, w \in X^*,\ uw \in L\},$$
$$W_2 = \{u\#v\#w \mid u, v, w \in X^*,\ vw \in L\}.$$

$$W' = \{u\#v\#w \mid u, v, w \in X^*, (uw \in L, vw \notin L) \text{ or } (uw \notin L, vw \in L)\}$$
$$= (W_1 \cap \overline{W_2}) \cup (\overline{W_1} \cap W_2)$$

$$W = \{u\#v \mid u, v \in X^*, u \not\equiv_L v\} = \{u\#v \mid u\#v\#w \in W' \text{ for some } w \in X^*\}$$

$$P = \{u\#v \mid u \equiv_L v\} = X^*\#X^* \setminus W$$

$$S = \{u_0\#u_1\# \cdots u_n\# \mid u_i \not\equiv_L u_j \text{ for all } i \neq j\}$$
$$= (X^*\#)^* \setminus \{ru\#sv\#t \mid r, s, t \in (X^*\#)^*, u\#v \in P\}.$$

Proof IV

Let M (*matching*) be the set of all words $v_1 \delta v_2 \# u_1 \# u_2$, $v_1, v_2, u_1, u_2 \in X^*$, with

Proof IV

Let M (*matching*) be the set of all words $v_1 \delta v_2 \# u_1 \# u_2$, $v_1, v_2, u_1, u_2 \in X^*$, with

- if $\delta = +$, then $v_1 \equiv_L u_1$ and $v_2 \equiv_L u_2$,

Proof IV

Let M (*matching*) be the set of all words $v_1 \delta v_2 \# u_1 \# u_2$, $v_1, v_2, u_1, u_2 \in X^*$, with

- if $\delta = +$, then $v_1 \equiv_L u_1$ and $v_2 \equiv_L u_2$,
- if $\delta = -$, then $v_1 \equiv_L u_2$ and $v_2 \equiv_L u_1$, and

Proof IV

Let M (*matching*) be the set of all words $v_1 \delta v_2 \# u_1 \# u_2$, $v_1, v_2, u_1, u_2 \in X^*$, with

- if $\delta = +$, then $v_1 \equiv_L u_1$ and $v_2 \equiv_L u_2$,
- if $\delta = -$, then $v_1 \equiv_L u_2$ and $v_2 \equiv_L u_1$, and
- if $\delta = z$, then $v_1 \equiv_L v_2 \equiv_L u_1$.

Proof IV

Let M (*matching*) be the set of all words $v_1 \delta v_2 \# u_1 \# u_2$, $v_1, v_2, u_1, u_2 \in X^*$, with

- if $\delta = +$, then $v_1 \equiv_L u_1$ and $v_2 \equiv_L u_2$,
- if $\delta = -$, then $v_1 \equiv_L u_2$ and $v_2 \equiv_L u_1$, and
- if $\delta = z$, then $v_1 \equiv_L v_2 \equiv_L u_1$.

$$\begin{aligned} M = & \{v_1{+}v_2\#u_1\#u_2 \mid v_1\#u_1 \in P,\ v_2\#u_2 \in P\} \\ & \cup \{v_1{-}v_2\#u_1\#u_2 \mid v_1\#u_2 \in P,\ v_2\#u_1 \in P\} \\ & \cup \{v_1 z v_2\#u_1\#u_2 \mid v_1\#v_2 \in P,\ v_1\#u_1 \in P,\ u_2 \in X^*\} \end{aligned}$$

Proof IV

Let M (*matching*) be the set of all words $v_1 \delta v_2 \# u_1 \# u_2$, $v_1, v_2, u_1, u_2 \in X^*$, with

- if $\delta = +$, then $v_1 \equiv_L u_1$ and $v_2 \equiv_L u_2$,
- if $\delta = -$, then $v_1 \equiv_L u_2$ and $v_2 \equiv_L u_1$, and
- if $\delta = z$, then $v_1 \equiv_L v_2 \equiv_L u_1$.

$$\begin{aligned} M = {} & \{v_1{+}v_2\#u_1\#u_2 \mid v_1\#u_1 \in P,\ v_2\#u_2 \in P\} \\ & \cup \{v_1{-}v_2\#u_1\#u_2 \mid v_1\#u_2 \in P,\ v_2\#u_1 \in P\} \\ & \cup \{v_1 z v_2\#u_1\#u_2 \mid v_1\#v_2 \in P,\ v_1\#u_1 \in P,\ u_2 \in X^*\} \end{aligned}$$

Let E (*error*) be the set of words $v_1 \delta v_2 \# u_0 \# \cdots u_n \#$ such that for every $1 \leqslant j \leqslant n$, we have $v_1 \delta v_2 \# u_{j-1} \# u_j \notin M$ or we have $\delta = z$ and $v_1 \not\equiv_L u_0$.

Proof IV

Let M (*matching*) be the set of all words $v_1\delta v_2\#u_1\#u_2$, $v_1, v_2, u_1, u_2 \in X^*$, with

- if $\delta = +$, then $v_1 \equiv_L u_1$ and $v_2 \equiv_L u_2$,
- if $\delta = -$, then $v_1 \equiv_L u_2$ and $v_2 \equiv_L u_1$, and
- if $\delta = z$, then $v_1 \equiv_L v_2 \equiv_L u_1$.

$$\begin{aligned} M = &\{v_1{+}v_2\#u_1\#u_2 \mid v_1\#u_1 \in P,\ v_2\#u_2 \in P\} \\ &\cup \{v_1{-}v_2\#u_1\#u_2 \mid v_1\#u_2 \in P,\ v_2\#u_1 \in P\} \\ &\cup \{v_1 z v_2\#u_1\#u_2 \mid v_1\#v_2 \in P,\ v_1\#u_1 \in P,\ u_2 \in X^*\} \end{aligned}$$

Let E (*error*) be the set of words $v_1\delta v_2\#u_0\#\cdots u_n\#$ such that for every $1 \leqslant j \leqslant n$, we have $v_1\delta v_2\#u_{j-1}\#u_j \notin M$ or we have $\delta = z$ and $v_1 \not\equiv_L u_0$.

$$E' = \{v_1\delta v_2\#ru_1\#u_2\#s \mid v_1\delta v_2\#u_1\#u_2 \in M, r, s \in (X^*\#)^*\}$$

Proof IV

Let M (*matching*) be the set of all words $v_1 \delta v_2 \# u_1 \# u_2$, $v_1, v_2, u_1, u_2 \in X^*$, with

- if $\delta = +$, then $v_1 \equiv_L u_1$ and $v_2 \equiv_L u_2$,
- if $\delta = -$, then $v_1 \equiv_L u_2$ and $v_2 \equiv_L u_1$, and
- if $\delta = z$, then $v_1 \equiv_L v_2 \equiv_L u_1$.

$$\begin{aligned} M = & \{v_1{+}v_2\#u_1\#u_2 \mid v_1\#u_1 \in P,\ v_2\#u_2 \in P\} \\ & \cup \{v_1{-}v_2\#u_1\#u_2 \mid v_1\#u_2 \in P,\ v_2\#u_1 \in P\} \\ & \cup \{v_1 z v_2\#u_1\#u_2 \mid v_1\#v_2 \in P,\ v_1\#u_1 \in P,\ u_2 \in X^*\} \end{aligned}$$

Let E (*error*) be the set of words $v_1 \delta v_2 \# u_0 \# \cdots u_n \#$ such that for every $1 \leqslant j \leqslant n$, we have $v_1 \delta v_2 \# u_{j-1} \# u_j \notin M$ or we have $\delta = z$ and $v_1 \not\equiv_L u_0$.

$$E' = \{v_1 \delta v_2 \# r u_1 \# u_2 \# s \mid v_1 \delta v_2 \# u_1 \# u_2 \in M, r, s \in (X^* \#)^*\}$$

$$E = \left[(X^* \Delta X^* \# (X^* \#)^* \setminus E'\right]$$

Proof IV

Let M (*matching*) be the set of all words $v_1 \delta v_2 \# u_1 \# u_2$, $v_1, v_2, u_1, u_2 \in X^*$, with

- if $\delta = +$, then $v_1 \equiv_L u_1$ and $v_2 \equiv_L u_2$,
- if $\delta = -$, then $v_1 \equiv_L u_2$ and $v_2 \equiv_L u_1$, and
- if $\delta = z$, then $v_1 \equiv_L v_2 \equiv_L u_1$.

$$\begin{aligned} M = {} & \{v_1{+}v_2\#u_1\#u_2 \mid v_1\#u_1 \in P,\ v_2\#u_2 \in P\} \\ & \cup \{v_1{-}v_2\#u_1\#u_2 \mid v_1\#u_2 \in P,\ v_2\#u_1 \in P\} \\ & \cup \{v_1 z v_2\#u_1\#u_2 \mid v_1\#v_2 \in P,\ v_1\#u_1 \in P,\ u_2 \in X^*\} \end{aligned}$$

Let E (*error*) be the set of words $v_1 \delta v_2 \# u_0 \# \cdots u_n \#$ such that for every $1 \leqslant j \leqslant n$, we have $v_1 \delta v_2 \# u_{j-1} \# u_j \notin M$ or we have $\delta = z$ and $v_1 \not\equiv_L u_0$.

$$E' = \{v_1 \delta v_2 \# r u_1 \# u_2 \# s \mid v_1 \delta v_2 \# u_1 \# u_2 \in M, r, s \in (X^*\#)^*\}$$

$$E = \left[(X^* \Delta X^* \#(X^*\#)^* \setminus E'\right] \cup \{v_1 z v_2 \# u_0 r \mid v_1 \not\equiv_L u_0,\ r \in (X^*\#)^*\}.$$

Proof V

Let N (*no error*) be the set of words $v_0\delta_1 v_1 \cdots \delta_m v_m \# u_0 \# \cdots u_n \#$ such that for every $1 \leqslant i \leqslant m$, there is a $1 \leqslant j \leqslant n$ with $v_{i-1}\delta_i v_i \# u_{j-1} \# u_j \in M$ and if $\delta_i = z$, then $v_{i-1} \equiv_L u_0$.

Proof V

Let N (*no error*) be the set of words $v_0\delta_1 v_1 \cdots \delta_m v_m \# u_0 \# \cdots u_n \#$ such that for every $1 \leqslant i \leqslant m$, there is a $1 \leqslant j \leqslant n$ with $v_{i-1}\delta_i v_i \# u_{j-1} \# u_j \in M$ and if $\delta_i = z$, then $v_{i-1} \equiv_L u_0$.

$$N' = \{w \in (X^*\Delta)^* v_1 \delta v_2 (\Delta X^*)^* \# u_0 \# \cdots u_n \# \mid v_1 \delta v_2 \# u_0 \# \cdots u_n \# \in E\},$$

Proof V

Let N (*no error*) be the set of words $v_0\delta_1 v_1 \cdots \delta_m v_m \# u_0 \# \cdots u_n \#$ such that for every $1 \leqslant i \leqslant m$, there is a $1 \leqslant j \leqslant n$ with $v_{i-1}\delta_i v_i \# u_{j-1} \# u_j \in M$ and if $\delta_i = z$, then $v_{i-1} \equiv_L u_0$.

$$N' = \{w \in (X^*\Delta)^* v_1 \delta v_2 (\Delta X^*)^* \# u_0 \# \cdots u_n \# \mid v_1 \delta v_2 \# u_0 \# \cdots u_n \# \in E\},$$
$$N = (X^*\Delta)^+ X^* \# (X^* \#)^* \setminus N'.$$

Proof V

Let N (*no error*) be the set of words $v_0\delta_1 v_1 \cdots \delta_m v_m \# u_0 \# \cdots u_n \#$ such that for every $1 \leqslant i \leqslant m$, there is a $1 \leqslant j \leqslant n$ with $v_{i-1}\delta_i v_i \# u_{j-1} \# u_j \in M$ and if $\delta_i = z$, then $v_{i-1} \equiv_L u_0$.

$$N' = \{w \in (X^*\Delta)^* v_1 \delta v_2 (\Delta X^*)^* \# u_0 \# \cdots u_n \# \mid v_1 \delta v_2 \# u_0 \# \cdots u_n \# \in E\},$$
$$N = (X^*\Delta)^+ X^* \# (X^* \#)^* \setminus N'.$$

Now we have

$$\hat{C}_L = N \cap (X^*\Delta)^* X^* \# S.$$

Hence, $C \in \mathcal{T}$.

Proof VI

RE $\subseteq \mathcal{T}$ follows by standard techniques:

Proof VI

RE $\subseteq \mathcal{T}$ follows by standard techniques:

- Use intersection to get counter language for two counters.

Proof VI

RE $\subseteq \mathcal{T}$ follows by standard techniques:

- Use intersection to get counter language for two counters.
- Use transducer to obtain language accepted by given two-counter automaton.

Proof VI

RE $\subseteq \mathcal{T}$ follows by standard techniques:

- Use intersection to get counter language for two counters.
- Use transducer to obtain language accepted by given two-counter automaton.

For AH$(L) \subseteq \mathcal{T}$: show that $K \in \mathcal{T}$ implies RE$(K) \subseteq \mathcal{T}$ (as above).

Corollary

Let L be non-regular. The smallest Boolean closed full trio containing L is $\mathrm{AH}(L)$.

Corollary

Let L be non-regular. The smallest Boolean closed full trio containing L is $\mathsf{AH}(L)$.

Corollary

Other than the regular languages, no full trio $\mathcal{T} \subseteq \mathsf{RE}$ is Boolean closed.

Corollary

Let L be non-regular. The smallest Boolean closed full trio containing L is $\mathsf{AH}(L)$*.*

Corollary

Other than the regular languages, no full trio $\mathcal{T} \subseteq \mathsf{RE}$ is Boolean closed.

Corollary

Other than the regular languages, no principal full trio is complementation closed.

Corollary

Let L be non-regular. The smallest Boolean closed full trio containing L is $\mathsf{AH}(L)$.

Corollary

Other than the regular languages, no full trio $\mathcal{T} \subseteq \mathsf{RE}$ is Boolean closed.

Corollary

Other than the regular languages, no principal full trio is complementation closed.

Proof.

Let $\mathcal{T}$ be generated by L.

□

Corollary

Let L be non-regular. The smallest Boolean closed full trio containing L is $\mathsf{AH}(L)$.

Corollary

Other than the regular languages, no full trio $\mathcal{T} \subseteq$ RE is Boolean closed.

Corollary

Other than the regular languages, no principal full trio is complementation closed.

Proof.

Let $\mathcal{T}$ be generated by L. It consists of RL for rational transductions R.

□

Corollary

Let L be non-regular. The smallest Boolean closed full trio containing L is $\mathsf{AH}(L)$.

Corollary

Other than the regular languages, no full trio $\mathcal{T} \subseteq \mathsf{RE}$ *is Boolean closed.*

Corollary

Other than the regular languages, no principal full trio is complementation closed.

Proof.

Let $\mathcal{T}$ be generated by L. It consists of RL for rational transductions R. Hence, $\mathcal{T}$ is union-closed and $\mathcal{T} \subseteq \mathsf{RE}(L) \subsetneq \mathsf{AH}(L)$.

□

Corollary

Let L be non-regular. The smallest Boolean closed full trio containing L is $\mathsf{AH}(L)$.

Corollary

Other than the regular languages, no full trio $\mathcal{T} \subseteq \mathsf{RE}$ is Boolean closed.

Corollary

Other than the regular languages, no principal full trio is complementation closed.

Proof.

Let $\mathcal{T}$ be generated by L. It consists of RL for rational transductions R. Hence, $\mathcal{T}$ is union-closed and $\mathcal{T} \subseteq \mathsf{RE}(L) \subsetneq \mathsf{AH}(L)$. If $\mathcal{T}$ were complementation closed, it would contain $\mathsf{AH}(L)$, contradiction! □

Application: Valence automata

Definition

A monoid is a set M together with an associative operation and a neutral element.

Application: Valence automata

Definition

A monoid is a set M together with an associative operation and a neutral element.

Valence Automata

Valence automaton over M:

- Finite automaton with edges $p \xrightarrow{w|m} q$, $w \in \Sigma^*$, $m \in M$.
- Run $q_0 \xrightarrow{w_1|m_1} q_1 \xrightarrow{w_2|m_2} \cdots \xrightarrow{w_n|m_n} q_n$ is *accepting for* $w_1 \cdots w_n$ if
  - q_0 is the initial state,
  - q_n is a final state, and

Application: Valence automata

Definition

A monoid is a set M together with an associative operation and a neutral element.

Valence Automata

Valence automaton over M:

- Finite automaton with edges $p \xrightarrow{w|m} q$, $w \in \Sigma^*$, $m \in M$.
- Run $q_0 \xrightarrow{w_1|m_1} q_1 \xrightarrow{w_2|m_2} \cdots \xrightarrow{w_n|m_n} q_n$ is *accepting for* $w_1 \cdots w_n$ if
  - q_0 is the initial state,
  - q_n is a final state, and
  - $m_1 \cdots m_n = 1$.

Application: Valence automata

Definition

A monoid is a set M together with an associative operation and a neutral element.

Valence Automata

Valence automaton over M:

- Finite automaton with edges $p \xrightarrow{w|m} q$, $w \in \Sigma^*$, $m \in M$.
- Run $q_0 \xrightarrow{w_1|m_1} q_1 \xrightarrow{w_2|m_2} \cdots \xrightarrow{w_n|m_n} q_n$ is *accepting for* $w_1 \cdots w_n$ if
  - q_0 is the initial state,
  - q_n is a final state, and
  - $m_1 \cdots m_n = 1$.

Language class

VA(M) languages accepted by valence automata over M.

Corollary

Let M be a finitely generated monoid. The following are equivalent:

1. $\mathsf{VA}(M)$ *is complementation closed.*
2. $\mathsf{VA}(M) = \mathsf{REG}$.
3. *M has finitely many right-invertible elements.*

Corollary

Let M be a finitely generated monoid. The following are equivalent:

1. $\mathsf{VA}(M)$ *is complementation closed.*
2. $\mathsf{VA}(M) = \mathsf{REG}$.
3. *M has finitely many right-invertible elements.*

Proof.

If M is finitely generated, $\mathsf{VA}(M)$ is a principal full trio.

□

Corollary

Let M be a finitely generated monoid. The following are equivalent:

1. $\mathrm{VA}(M)$ *is complementation closed.*
2. $\mathrm{VA}(M) = \mathrm{REG}$.
3. M *has finitely many right-invertible elements.*

Proof.

If M is finitely generated, $\mathrm{VA}(M)$ is a principal full trio. Equivalence of 2 and 3 has been shown by Render (2010) and Z. (2011). □

Application: Rational Kripke Frames

Improvement

In order to construct languages over $\{0, 1\}$, three fixed rational transductions suffice.

Application: Rational Kripke Frames

Improvement

In order to construct languages over $\{0, 1\}$, three fixed rational transductions suffice.

Theorem

Let $X = \{0, 1\}$. There is a Kripke frame with

- *X^* as its set of worlds and*
- *rational transductions $R, S, T \subseteq X^* \times X^*$ as modalities*

such that for any non-regular L, in the Kripke structure $\mathcal{K} = (X^*, R, S, T, L)$, for each $K \in \mathrm{AH}(L)$, there is a φ with $[\![\varphi]\!]_{\mathcal{K}} = K$.

What about synchronous rational transductions?

Synchronous rational transductions

- For automatic structures, synchronous rational transductions suffice.

What about synchronous rational transductions?

Synchronous rational transductions

- For automatic structures, synchronous rational transductions suffice.
- Does every language class (beyon REG) closed under synchronous rational transductions and Boolean operations have undecidable emptiness?

What about synchronous rational transductions?

Synchronous rational transductions

- For automatic structures, synchronous rational transductions suffice.
- Does every language class (beyon REG) closed under synchronous rational transductions and Boolean operations have undecidable emptiness?

Result

What about synchronous rational transductions?

Synchronous rational transductions

- For automatic structures, synchronous rational transductions suffice.
- Does every language class (beyon REG) closed under synchronous rational transductions and Boolean operations have undecidable emptiness?

Result

- No: Pick an ω-word w with a decidable MSO theory but with non-regular set of prefixes

What about synchronous rational transductions?

Synchronous rational transductions

- For automatic structures, synchronous rational transductions suffice.
- Does every language class (beyon REG) closed under synchronous rational transductions and Boolean operations have undecidable emptiness?

Result

- No: Pick an ω-word w with a decidable MSO theory but with non-regular set of prefixes
- Example: Characteristic sequence of factorials, k-th powers

What about synchronous rational transductions?

Synchronous rational transductions

- For automatic structures, synchronous rational transductions suffice.
- Does every language class (beyon REG) closed under synchronous rational transductions and Boolean operations have undecidable emptiness?

Result

- No: Pick an ω-word w with a decidable MSO theory but with non-regular set of prefixes
- Example: Characteristic sequence of factorials, k-th powers
- Take the class containing the set of prefixes P of w and build the closure under synchronous rational transductions and Boolean operations

What about synchronous rational transductions?

Synchronous rational transductions

- For automatic structures, synchronous rational transductions suffice.
- Does every language class (beyon REG) closed under synchronous rational transductions and Boolean operations have undecidable emptiness?

Result

- No: Pick an ω-word w with a decidable MSO theory but with non-regular set of prefixes
- Example: Characteristic sequence of factorials, k-th powers
- Take the class containing the set of prefixes P of w and build the closure under synchronous rational transductions and Boolean operations
- Decidable MSO-theory yields decidability

Definition

Let $L \subseteq X^*$. A *neutral word for L* is a word $v \in X^+$ such that $uvw \in L$ if and only if $uw \in L$ for any $u, w \in X^*$.

Definition

Let $L \subseteq X^*$. A *neutral word for L* is a word $v \in X^+$ such that $uvw \in L$ if and only if $uw \in L$ for any $u, w \in X^*$.

Example: Identity language of a finitely generated infinite group.

Definition

Let $L \subseteq X^*$. A *neutral word for* L is a word $v \in X^+$ such that $uvw \in L$ if and only if $uw \in L$ for any $u, w \in X^*$.

Example: Identity language of a finitely generated infinite group.

Theorem

Let $L \subseteq \{0, 1\}^*$ be a non-regular language with a neutral word. Using synchronous rational transductions and Boolean operations, one can construct a non-recursively enumerable language from L. If, in addition, L is recursive, one can construct a Σ_n-hard language from L.