Characterizing Tree Valuation Weighted Languages by Multioperator Weighted Tree Languages

Markus Teichmann[1] Johannes Osterholzer[2]

September 26, 2013

[1]Supported by DFG Graduiertenkolleg 1763 (QuantLA)
[2]Partially supported by DFG Graduiertenkolleg 1763 (QuantLA)

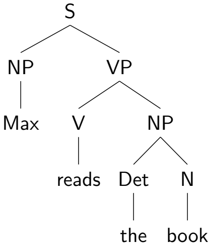
Outline

Motivation

Tree Languages

- automated theorem proving, databases, natural language processing, XML, etc.
- recognized/defined by [Büchi/Elgot]
  - finite state tree automata
  - monadic second order logic (mso)

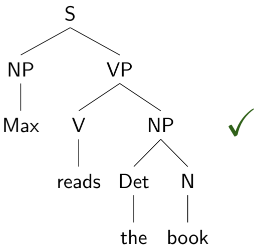
Motivation



Tree Languages

- automated theorem proving, databases, natural language processing, XML, etc.
- recognized/defined by [Büchi/Elgot]
  - finite state tree automata
  - monadic second order logic (mso)

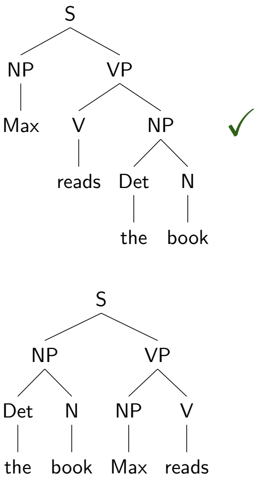
Motivation



Tree Languages

- automated theorem proving, databases, natural language processing, XML, etc.
- recognized/defined by [Büchi/Elgot]
  - finite state tree automata
  - monadic second order logic (mso)

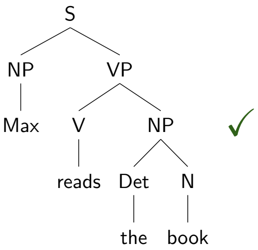
Motivation

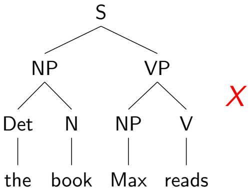


Tree Languages

- automated theorem proving, databases, natural language processing, XML, etc.
- recognized/defined by [Büchi/Elgot]
  - finite state tree automata
  - monadic second order logic (mso)

Motivation

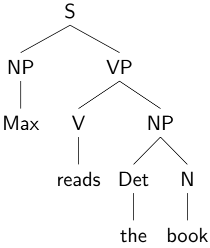


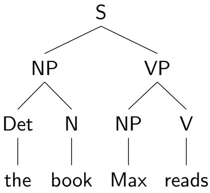


Tree Languages

- automated theorem proving, databases, natural language processing, XML, etc.
- recognized/defined by [Büchi/Elgot]
  - finite state tree automata
  - monadic second order logic (mso)

Motivation





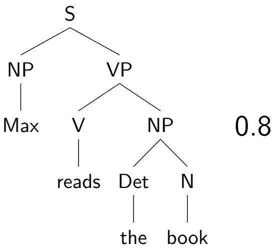
Tree Languages

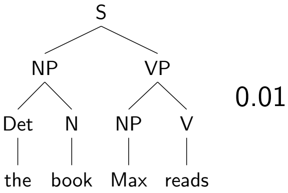
- automated theorem proving, databases, natural language processing, XML, etc.
- recognized/defined by [Büchi/Elgot]
  - finite state tree automata
  - monadic second order logic (mso)

Weighted Tree Languages

- quantitative aspects
  - degree of belief
  - counts of features
- recognized/defined by
  - weighted finite state automata
  - weighted mso

Motivation





Tree Languages

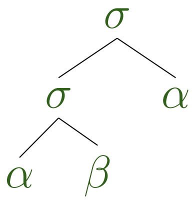
- automated theorem proving, databases, natural language processing, XML, etc.
- recognized/defined by [Büchi/Elgot]
  - finite state tree automata
  - monadic second order logic (mso)

Weighted Tree Languages

- quantitative aspects
  - degree of belief
  - counts of features
- recognized/defined by
  - weighted finite state automata
  - weighted mso

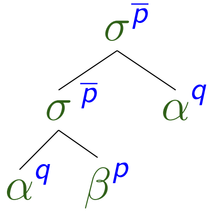
Local State Behavior

- finite state set $Q = \{\overline{p}, p, q\}$
- runs $R_Q(\xi) = \{r \mid r\colon \mathrm{pos}(\xi) \to Q\}$

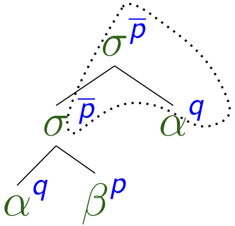


Local State Behavior

- finite state set $Q = \{\overline{p}, p, q\}$
- runs $R_Q(\xi) = \{r \mid r : \mathrm{pos}(\xi) \to Q\}$

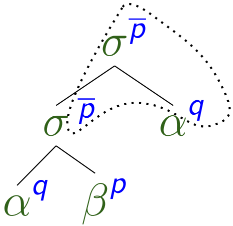


Local State Behavior



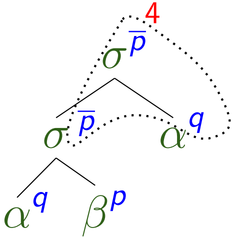
- finite state set $Q = \{\overline{p}, p, q\}$
- runs $R_Q(\xi) = \{r \mid r \colon \mathrm{pos}(\xi) \to Q\}$
- local state behavior:
  - symbol: σ
  - state: $\overline{p}$
  - successor states: $\overline{p}, q$

Local State Behavior



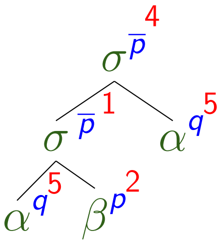
- finite state set $Q = \{\overline{p}, p, q\}$
- runs $R_Q(\xi) = \{r \mid r \colon \mathrm{pos}(\xi) \to Q\}$
- local state behavior:
  - symbol: σ
  - state: $\overline{p}$
  - successor states: $\overline{p}, q$
- local weight $\delta_\sigma(\overline{p}, \overline{p}q)$

Local State Behavior



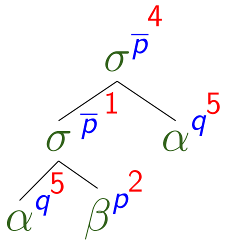
- finite state set $Q = \{\overline{p}, p, q\}$
- runs $R_Q(\xi) = \{r \mid r \colon \mathrm{pos}(\xi) \to Q\}$
- local state behavior:
  - symbol: σ
  - state: $\overline{p}$
  - successor states: $\overline{p}, q$
- local weight $\delta_\sigma(\overline{p}, \overline{p}q)$

Local State Behavior



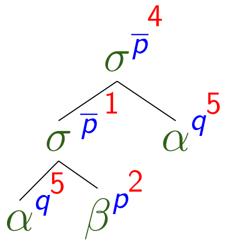
- finite state set $Q = \{\overline{p}, p, q\}$
- runs $R_Q(\xi) = \{r \mid r \colon \mathrm{pos}(\xi) \to Q\}$
- local state behavior:
  - symbol: σ
  - state: $\overline{p}$
  - successor states: $\overline{p}, q$
- local weight $\delta_\sigma(\overline{p}, \overline{p}q)$

Local State Behavior



- finite state set $Q = \{\overline{p}, p, q\}$
- runs $R_Q(\xi) = \{r \mid r \colon \mathrm{pos}(\xi) \to Q\}$
- local state behavior:
  - symbol: σ
  - state: $\overline{p}$
  - successor states: $\overline{p}, q$
- local weight $\delta_\sigma(\overline{p}, \overline{p}q)$
- accumulate weights:

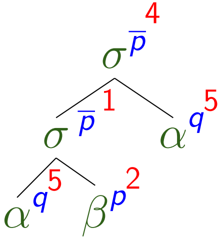
Local State Behavior



- finite state set $Q = \{\overline{p}, p, q\}$
- runs $R_Q(\xi) = \{r \mid r \colon \mathrm{pos}(\xi) \to Q\}$
- local state behavior:
  - symbol: σ
  - state: $\overline{p}$
  - successor states: $\overline{p}, q$
- local weight $\delta_\sigma(\overline{p}, \overline{p}q)$
- accumulate weights:

Semiring: $S_{\mathbb{N}} = (\mathbb{N}, +, \cdot, 0, 1)$

Local State Behavior

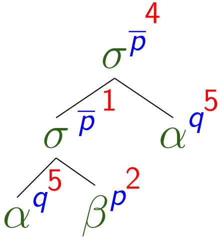


- finite state set $Q = \{\overline{p}, p, q\}$
- runs $R_Q(\xi) = \{r \mid r\colon \mathrm{pos}(\xi) \to Q\}$
- local state behavior:
  - symbol: σ
  - state: $\overline{p}$
  - successor states: $\overline{p}, q$
- local weight $\delta_\sigma(\overline{p}, \overline{p}q)$
- accumulate weights:

Semiring: $S_{\mathbb{N}} = (\mathbb{N}, +, \cdot, 0, 1)$

$$4 \cdot 1 \cdot 5 \cdot 2 \cdot 5 = 200$$

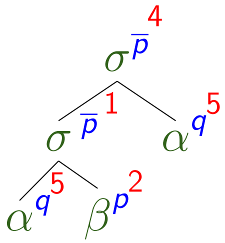
Local State Behavior



- finite state set $Q = \{\overline{p}, p, q\}$
- runs $R_Q(\xi) = \{r \mid r\colon \mathrm{pos}(\xi) \to Q\}$
- local state behavior:
  - symbol: σ
  - state: $\overline{p}$
  - successor states: $\overline{p}, q$
- local weight $\delta_\sigma(\overline{p}, \overline{p}q)$
- accumulate weights:

Semiring: $S_{\mathrm{cap}} = (\mathbb{N}_{+\infty}, \max, \min, 0, +\infty)$

Local State Behavior

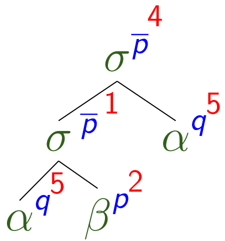


- finite state set $Q = \{\overline{p}, p, q\}$
- runs $R_Q(\xi) = \{r \mid r \colon \mathrm{pos}(\xi) \to Q\}$
- local state behavior:
  - symbol: σ
  - state: $\overline{p}$
  - successor states: $\overline{p}, q$
- local weight $\delta_\sigma(\overline{p}, \overline{p}q)$
- accumulate weights:

Semiring: $S_{\mathrm{cap}} = (\mathbb{N}_{+\infty}, \max, \min, 0, +\infty)$

$$\min\{4, 1, 5, 2, 5\} = 1$$

Local State Behavior



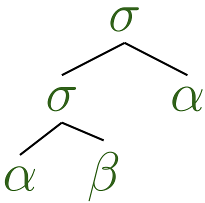
- finite state set $Q = \{\overline{p}, p, q\}$
- runs $R_Q(\xi) = \{r \mid r\colon \mathrm{pos}(\xi) \to Q\}$
- local state behavior:
  - symbol: σ
  - state: $\overline{p}$
  - successor states: $\overline{p}, q$
- local weight $\delta_\sigma(\overline{p}, \overline{p}q)$
- accumulate weights:

Semiring: $S_{\mathrm{cap}} = (\mathbb{N}_{+\infty}, \max, \min, 0, +\infty)$

$$\min\{4, 1, 5, 2, 5\} = 1$$

- need for more flexible formalisms

Multioperator Weighted Tree Automaton [FSV12]



multioperator monoid:

$\mathcal{A} = (A, +, 0, \Omega)$

- $(A, +, 0)$ commutative monoid
- $\Omega \subseteq \mathrm{Ops}(A)$ - absorptive

m-wta over Σ and $\mathcal{A}$

$\mathcal{M} = (Q, \delta, F)$

- $\delta = (\delta_\sigma \mid \sigma \in \Sigma)$

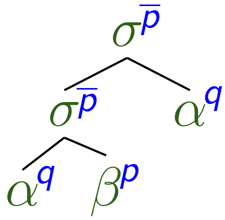
  $\delta_\sigma \colon Q^k \times Q \to \Omega^{(k)}$

language recognized by $\mathcal{M}$

$$[\![\mathcal{M}]\!] : T_\Sigma \to \mathcal{A}$$

$$[\![\mathcal{M}]\!](\xi) = \sum_{\substack{r \in R_Q(\xi) \\ r(\varepsilon) \in F}} \delta(r, \xi, \varepsilon)$$

Multioperator Weighted Tree Automaton [FSV12]



multioperator monoid:
$\mathcal{A} = (A, +, 0, \Omega)$

- $(A, +, 0)$ commutative monoid
- $\Omega \subseteq \mathrm{Ops}(A)$ - absorptive

m-wta over Σ and $\mathcal{A}$
$\mathcal{M} = (Q, \delta, F)$

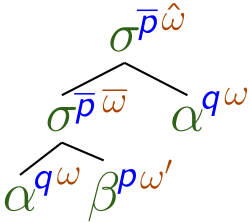
- $\delta = (\delta_\sigma \mid \sigma \in \Sigma)$
  $\delta_\sigma \colon Q^k \times Q \to \Omega^{(k)}$

language recognized by $\mathcal{M}$

$[\![\mathcal{M}]\!] : T_\Sigma \to \mathcal{A}$

$[\![\mathcal{M}]\!](\xi) = \sum_{\substack{r \in R_Q(\xi) \\ r(\varepsilon) \in F}} \delta(r, \xi, \varepsilon)$

Multioperator Weighted Tree Automaton [FSV12]



multioperator monoid:
$\mathcal{A} = (A, +, 0, \Omega)$

- $(A, +, 0)$ commutative monoid
- $\Omega \subseteq \mathrm{Ops}(A)$ - absorptive

m-wta over Σ and $\mathcal{A}$
$\mathcal{M} = (Q, \delta, F)$

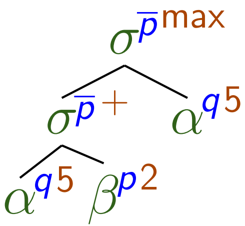
- $\delta = (\delta_\sigma \mid \sigma \in \Sigma)$
  $\delta_\sigma \colon Q^k \times Q \to \Omega^{(k)}$

language recognized by $\mathcal{M}$

$[\![\mathcal{M}]\!] : T_\Sigma \to \mathcal{A}$

$$[\![\mathcal{M}]\!](\xi) = \sum_{\substack{r \in R_Q(\xi) \\ r(\varepsilon) \in F}} \delta(r, \xi, \varepsilon)$$

Multioperator Weighted Tree Automaton [FSV12]



multioperator monoid:

$\mathcal{A} = (A, +, 0, \Omega)$

- $(A, +, 0)$ commutative monoid
- $\Omega \subseteq \mathrm{Ops}(A)$ - absorptive

m-wta over Σ and $\mathcal{A}$

$\mathcal{M} = (Q, \delta, F)$

- $\delta = (\delta_\sigma \mid \sigma \in \Sigma)$

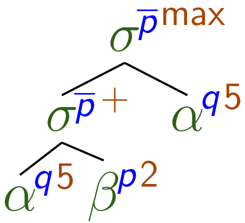
  $\delta_\sigma \colon Q^k \times Q \to \Omega^{(k)}$

language recognized by $\mathcal{M}$

$[\![\mathcal{M}]\!] : T_\Sigma \to \mathcal{A}$

$[\![\mathcal{M}]\!](\xi) = \sum_{\substack{r \in R_Q(\xi) \\ r(\varepsilon) \in F}} \delta(r, \xi, \varepsilon)$

Multioperator Weighted Tree Automaton [FSV12]



$\max(+(5,2),5) = 7$

multioperator monoid:
$\mathcal{A} = (A, +, 0, \Omega)$

- $(A, +, 0)$ commutative monoid
- $\Omega \subseteq \mathrm{Ops}(A)$ - absorptive

m-wta over Σ and $\mathcal{A}$
$\mathcal{M} = (Q, \delta, F)$

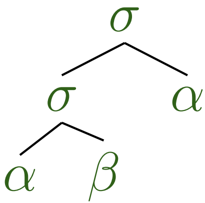
- $\delta = (\delta_\sigma \mid \sigma \in \Sigma)$
  $\delta_\sigma \colon Q^k \times Q \to \Omega^{(k)}$

language recognized by $\mathcal{M}$

$[\![\mathcal{M}]\!] : T_\Sigma \to \mathcal{A}$

$[\![\mathcal{M}]\!](\xi) = \sum_{\substack{r \in R_Q(\xi) \\ r(\varepsilon) \in F}} \delta(r, \xi, \varepsilon)$

Tree Valuation Weighted Tree Automaton [DGMM11]



product tree valuation monoid:

$\mathbb{D} = (D, +, \diamond, 0, 1, \mathrm{Val})$

- $(D, +, 0)$ commutative monoid
- $\mathrm{Val}\colon T^u_D \to D$ - only for 0-free
- $\diamond$ “multiplication” (only for logics)

tv-wta over Σ and $\mathbb{D}$

$\mathcal{N} = (Q, \mu, F)$

- $\mu = (\mu_\sigma \mid \sigma \in \Sigma)$

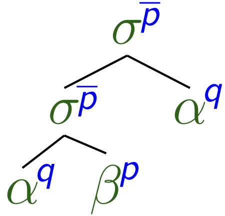
  $\mu_\sigma\colon Q^k \times Q \to D$

language recognized by $\mathcal{N}$

$[\![\mathcal{N}]\!]\colon T_\Sigma \to D$

$$[\![\mathcal{N}]\!](\xi) = \sum_{\substack{r \in R(\xi) \\ r(\varepsilon) \in F}} \mathrm{Val}(\mu(r, \xi))$$

Tree Valuation Weighted Tree Automaton [DGMM11]



product tree valuation monoid:

$\mathbb{D} = (D, +, \diamond, 0, 1, \mathrm{Val})$

- $(D, +, 0)$ commutative monoid
- $\mathrm{Val}\colon T_D^u \to D$ - only for 0-free
- $\diamond$ "multiplication" (only for logics)

tv-wta over Σ and $\mathbb{D}$

$\mathcal{N} = (Q, \mu, F)$

- $\mu = (\mu_\sigma \mid \sigma \in \Sigma)$

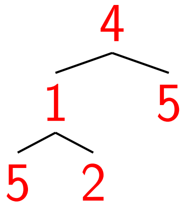
  $\mu_\sigma\colon Q^k \times Q \to D$

language recognized by $\mathcal{N}$

$$[\![\mathcal{N}]\!]\colon T_\Sigma \to D$$

$$[\![\mathcal{N}]\!](\xi) = \sum_{\substack{r \in R(\xi) \\ r(\varepsilon) \in F}} \mathrm{Val}(\mu(r, \xi))$$

Tree Valuation Weighted Tree Automaton [DGMM11]



product tree valuation monoid:
$\mathbb{D} = (D, +, \diamond, 0, 1, \mathrm{Val})$

- $(D, +, 0)$ commutative monoid
- $\mathrm{Val}\colon T_D^u \to D$ - only for 0-free
- $\diamond$ "multiplication" (only for logics)

tv-wta over Σ and $\mathbb{D}$
$\mathcal{N} = (Q, \mu, F)$

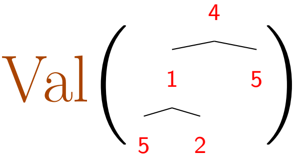
- $\mu = (\mu_\sigma \mid \sigma \in \Sigma)$
  $\mu_\sigma\colon Q^k \times Q \to D$

language recognized by $\mathcal{N}$

$[\![\mathcal{N}]\!]\colon T_\Sigma \to D$

$[\![\mathcal{N}]\!](\xi) = \sum_{\substack{r \in R(\xi) \\ r(\varepsilon) \in F}} \mathrm{Val}(\mu(r, \xi))$

Tree Valuation Weighted Tree Automaton [DGMM11]



product tree valuation monoid:
$\mathbb{D} = (D, +, \diamond, 0, 1, \mathrm{Val})$

- $(D, +, 0)$ commutative monoid
- $\mathrm{Val}\colon T_D^u \to D$ - only for 0-free
- $\diamond$ “multiplication” (only for logics)

tv-wta over Σ and $\mathbb{D}$
$\mathcal{N} = (Q, \mu, F)$

- $\mu = (\mu_\sigma \mid \sigma \in \Sigma)$
  $\mu_\sigma\colon Q^k \times Q \to D$

language recognized by $\mathcal{N}$

$[\![\mathcal{N}]\!]\colon T_\Sigma \to D$

$[\![\mathcal{N}]\!](\xi) = \sum_{\substack{r \in R(\xi) \\ r(\varepsilon) \in F}} \mathrm{Val}(\mu(r, \xi))$

Theorem

For every regular product tv-monoid $\mathbb{D} = (D, +, \diamond, 0, 1, \mathrm{Val})$
m-monoid $\mathcal{A} = (A, \oplus, \mathbf{0}, \Omega)$

tv-wta

m-wta

$\mathrm{Rec}(\Sigma, X)$: class of recognizable weighted tree languages over X
$\mathrm{Def}(\Sigma, X)$: class of definable weighted tree languages over X

Theorem

For every regular product tv-monoid $\mathbb{D} = (D, +, \diamond, 0, 1, \mathrm{Val})$
m-monoid $\mathcal{A} = (A, \oplus, \mathbf{0}, \Omega)$

$$\text{tv-wta} \xrightarrow{[\![\cdot]\!]} \mathrm{Rec}(\Sigma, \mathbb{D})$$

$$\text{m-wta} \xrightarrow[{[\![\cdot]\!]}]{} \mathrm{Rec}(\Sigma, \mathcal{A} \ \ )$$

$\mathrm{Rec}(\Sigma, X)$: class of recognizable weighted tree languages over X
$\mathrm{Def}(\Sigma, X)$: class of definable weighted tree languages over X

Theorem

For every regular product tv-monoid $\mathbb{D} = (D, +, \diamond, 0, 1, \mathrm{Val})$

$$\text{tv-wta} \xrightarrow{[\![\cdot]\!]} \mathrm{Rec}(\Sigma, \mathbb{D})$$

$$\text{m-wta} \xrightarrow[{[\![\cdot]\!]}]{} \mathrm{Rec}(\Sigma, \mathcal{A}_{\mathbb{D}})$$

$\mathrm{Rec}(\Sigma, X)$: class of recognizable weighted tree languages over X
$\mathrm{Def}(\Sigma, X)$: class of definable weighted tree languages over X

Construction of the m-monoid $\mathcal{A}_{\mathbb{D}}$

$$\mathbb{D} = (D, +, \diamond, 0, 1, \mathrm{Val}) \rightsquigarrow \mathcal{A}_{\mathbb{D}} = (D \times T_D^u, \oplus, (0,0), \Omega)$$

Construction of the m-monoid $\mathcal{A}_{\mathbb{D}}$

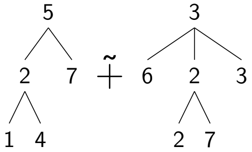
$$\mathbb{D} = (D, +, \diamond, 0, 1, \mathrm{Val}) \rightsquigarrow \mathcal{A}_{\mathbb{D}} = (D \times T_D^u, \oplus, (0,0), \Omega)$$

$$\oplus \colon (D \times T_D^u) \times (D \times T_D^u) \to (D \times T_D^u)$$

$$(d_1, \xi_1) \oplus (d_2, \xi_2) = (d_1 + d_2, \xi_1 \tilde{+} \xi_2)$$

Construction of the m-monoid $\mathcal{A}_{\mathbb{D}}$

$$\mathbb{D} = (D, +, \diamond, 0, 1, \mathrm{Val}) \rightsquigarrow \mathcal{A}_{\mathbb{D}} = (D \times T_D^u, \oplus, (0,0), \Omega)$$

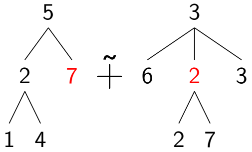


$$\oplus \colon (D \times T_D^u) \times (D \times T_D^u) \to (D \times T_D^u)$$

$$(d_1, \xi_1) \oplus (d_2, \xi_2) = (d_1 + d_2, \xi_1 \tilde{+} \xi_2)$$

Construction of the m-monoid $\mathcal{A}_{\mathbb{D}}$

$$\mathbb{D} = (D, +, \diamond, 0, 1, \mathrm{Val}) \rightsquigarrow \mathcal{A}_{\mathbb{D}} = (D \times T_D^u, \oplus, (0,0), \Omega)$$



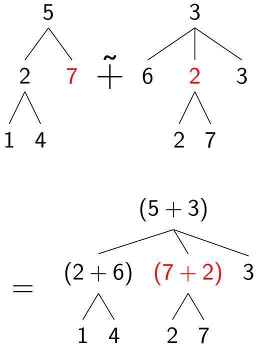
$$\oplus \colon (D \times T_D^u) \times (D \times T_D^u) \to (D \times T_D^u)$$

$$(d_1, \xi_1) \oplus (d_2, \xi_2) = (d_1 + d_2, \xi_1 \tilde{+} \xi_2)$$

$=$

Construction of the m-monoid $\mathcal{A}_{\mathbb{D}}$

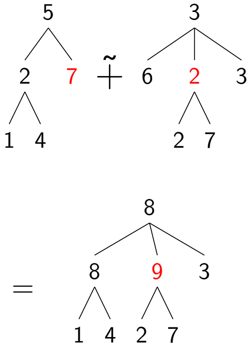
$$\mathbb{D} = (D, +, \diamond, 0, 1, \mathrm{Val}) \rightsquigarrow \mathcal{A}_{\mathbb{D}} = (D \times T_D^u, \oplus, (0,0), \Omega)$$



$$\oplus \colon (D \times T_D^u) \times (D \times T_D^u) \to (D \times T_D^u)$$
$$(d_1, \xi_1) \oplus (d_2, \xi_2) = (d_1 + d_2, \xi_1 \tilde{+} \xi_2)$$

Construction of the m-monoid $\mathcal{A}_{\mathbb{D}}$

$$\mathbb{D} = (D, +, \diamond, 0, 1, \mathrm{Val}) \rightsquigarrow \mathcal{A}_{\mathbb{D}} = (D \times T_D^u, \oplus, (0,0), \Omega)$$

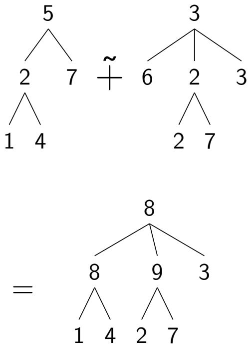


$$\oplus \colon (D \times T_D^u) \times (D \times T_D^u) \to (D \times T_D^u)$$

$$(d_1, \xi_1) \oplus (d_2, \xi_2) = (d_1 + d_2, \xi_1 \tilde{+} \xi_2)$$

Construction of the m-monoid $\mathcal{A}_{\mathbb{D}}$

$$\mathbb{D} = (D, +, \diamond, 0, 1, \mathrm{Val}) \rightsquigarrow \mathcal{A}_{\mathbb{D}} = (D \times T^u_D, \oplus, (0,0), \Omega)$$



$$\oplus \colon (D \times T^u_D) \times (D \times T^u_D) \to (D \times T^u_D)$$

$$(d_1, \xi_1) \oplus (d_2, \xi_2) = (d_1 + d_2, \xi_1 \tilde{+} \xi_2)$$

$$\Omega = \{\mathrm{valtop}_d{}^{(k)} \mid d \in D, k \in \mathbb{N}\}$$

Construction of the m-monoid $\mathcal{A}_{\mathbb{D}}$

$$\mathbb{D} = (D, +, \diamond, 0, 1, \mathrm{Val}) \rightsquigarrow \mathcal{A}_{\mathbb{D}} = (D \times T_D^u, \oplus, (0,0), \Omega)$$

$$\oplus\colon (D \times T_D^u) \times (D \times T_D^u) \to (D \times T_D^u)$$

$$(d_1, \xi_1) \oplus (d_2, \xi_2) = (d_1 + d_2, \xi_1 \tilde{+} \xi_2)$$

$$\Omega = \{\mathrm{valtop}_d{}^{(k)} \mid d \in D, k \in \mathbb{N}\}$$

Construction of the m-monoid $\mathcal{A}_{\mathbb{D}}$

$$\mathbb{D} = (D, +, \diamond, 0, 1, \mathrm{Val}) \rightsquigarrow \mathcal{A}_{\mathbb{D}} = (D \times T^u_D, \oplus, (0,0), \Omega)$$

$$\oplus\colon (D \times T^u_D) \times (D \times T^u_D) \to (D \times T^u_D)$$
$$(d_1, \xi_1) \oplus (d_2, \xi_2) = (d_1 + d_2, \xi_1 \tilde{+} \xi_2)$$

$$\Omega = \{\mathrm{valtop}_d{}^{(k)} \mid d \in D, k \in \mathbb{N}\}$$

$$\mathrm{valtop}_d{}^{(k)}\colon (D \times T^u_D)^k \to D \times T^u_D$$
$$\mathrm{valtop}_d{}^{(k)}((d_1, \xi_1), \ldots, (d_k, \xi_k)) = (\mathrm{Val}(\xi), \xi)$$
$$\xi = d(\xi_1, \ldots, \xi_k)$$

Construction of the m-monoid $\mathcal{A}_{\mathbb{D}}$

$$\mathbb{D} = (D, +, \diamond, 0, 1, \mathrm{Val}) \rightsquigarrow \mathcal{A}_{\mathbb{D}} = (D \times T_D^u, \oplus, (0,0), \Omega)$$

$$\oplus \colon (D \times T_D^u) \times (D \times T_D^u) \to (D \times T_D^u)$$

$$(d_1, \xi_1) \oplus (d_2, \xi_2) = (d_1 + d_2, \xi_1 \tilde{+} \xi_2)$$

$\mathrm{valtop}_5{}^{(2)}$:

$$\Omega = \{\mathrm{valtop}_d{}^{(k)} \mid d \in D, k \in \mathbb{N}\}$$

$$\mathrm{valtop}_d{}^{(k)} \colon (D \times T_D^u)^k \to D \times T_D^u$$

$$\mathrm{valtop}_d{}^{(k)}((d_1, \xi_1), \ldots, (d_k, \xi_k)) = (\mathrm{Val}(\xi), \xi)$$

$$\xi = d(\xi_1, \ldots, \xi_k)$$

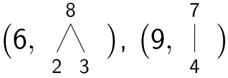
Construction of the m-monoid $\mathcal{A}_{\mathbb{D}}$

$$\mathbb{D} = (D, +, \diamond, 0, 1, \mathrm{Val}) \rightsquigarrow \mathcal{A}_{\mathbb{D}} = (D \times T_D^u, \oplus, (0,0), \Omega)$$

$$\oplus\colon (D \times T_D^u) \times (D \times T_D^u) \to (D \times T_D^u)$$

$$(d_1, \xi_1) \oplus (d_2, \xi_2) = (d_1 + d_2, \xi_1 \tilde{+} \xi_2)$$

$\mathrm{valtop}_5{}^{(2)}$:



$$\Omega = \{\mathrm{valtop}_d{}^{(k)} \mid d \in D, k \in \mathbb{N}\}$$

$$\mathrm{valtop}_d{}^{(k)}\colon (D \times T_D^u)^k \to D \times T_D^u$$

$$\mathrm{valtop}_d{}^{(k)}((d_1, \xi_1), \ldots, (d_k, \xi_k)) = (\mathrm{Val}(\xi), \xi)$$

$$\xi = d(\xi_1, \ldots, \xi_k)$$

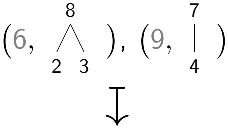
Construction of the m-monoid $\mathcal{A}_{\mathbb{D}}$

$$\mathbb{D} = (D, +, \diamond, 0, 1, \mathrm{Val}) \rightsquigarrow \mathcal{A}_{\mathbb{D}} = (D \times T_D^u, \oplus, (0,0), \Omega)$$

$$\oplus \colon (D \times T_D^u) \times (D \times T_D^u) \to (D \times T_D^u)$$

$$(d_1, \xi_1) \oplus (d_2, \xi_2) = (d_1 + d_2, \xi_1 \tilde{+} \xi_2)$$

$\mathrm{valtop}_5{}^{(2)}$:



$$\Omega = \{\mathrm{valtop}_d{}^{(k)} \mid d \in D, k \in \mathbb{N}\}$$

$$\mathrm{valtop}_d{}^{(k)} \colon (D \times T_D^u)^k \to D \times T_D^u$$

$$\mathrm{valtop}_d{}^{(k)}((d_1, \xi_1), \ldots, (d_k, \xi_k)) = (\mathrm{Val}(\xi), \xi)$$

$$\xi = d(\xi_1, \ldots, \xi_k)$$

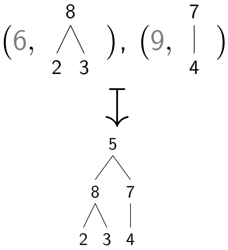
Construction of the m-monoid $\mathcal{A}_{\mathbb{D}}$

$$\mathbb{D} = (D, +, \diamond, 0, 1, \mathrm{Val}) \rightsquigarrow \mathcal{A}_{\mathbb{D}} = (D \times T_D^u, \oplus, (0,0), \Omega)$$

$$\oplus\colon (D \times T_D^u) \times (D \times T_D^u) \to (D \times T_D^u)$$

$$(d_1, \xi_1) \oplus (d_2, \xi_2) = (d_1 + d_2, \xi_1 \tilde{+} \xi_2)$$

$\mathrm{valtop}_5{}^{(2)}$:



$$\Omega = \{\mathrm{valtop}_d{}^{(k)} \mid d \in D, k \in \mathbb{N}\}$$

$$\mathrm{valtop}_d{}^{(k)}\colon (D \times T_D^u)^k \to D \times T_D^u$$

$$\mathrm{valtop}_d{}^{(k)}((d_1, \xi_1), \dots, (d_k, \xi_k)) = (\mathrm{Val}(\xi), \xi)$$

$$\xi = d(\xi_1, \dots, \xi_k)$$

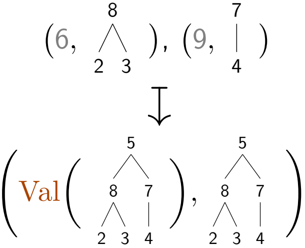
Construction of the m-monoid $\mathcal{A}_{\mathbb{D}}$

$$\mathbb{D} = (D, +, \diamond, 0, 1, \mathrm{Val}) \rightsquigarrow \mathcal{A}_{\mathbb{D}} = (D \times T_D^u, \oplus, (0,0), \Omega)$$

$$\oplus\colon (D \times T_D^u) \times (D \times T_D^u) \to (D \times T_D^u)$$

$$(d_1, \xi_1) \oplus (d_2, \xi_2) = (d_1 + d_2, \xi_1 \tilde{+} \xi_2)$$

$\mathrm{valtop}_5{}^{(2)}$:



$$\Omega = \{\mathrm{valtop}_d{}^{(k)} \mid d \in D, k \in \mathbb{N}\}$$

$$\mathrm{valtop}_d{}^{(k)}\colon (D \times T_D^u)^k \to D \times T_D^u$$

$$\mathrm{valtop}_d{}^{(k)}((d_1, \xi_1), \ldots, (d_k, \xi_k)) = (\mathrm{Val}(\xi), \xi)$$

$$\xi = d(\xi_1, \ldots, \xi_k)$$

Theorem

For every regular product tv-monoid $\mathbb{D} = (D, +, \diamond, 0, 1, \mathrm{Val})$

$$\text{tv-wta} \xrightarrow{[\![\cdot]\!]} \mathrm{Rec}(\Sigma, \mathbb{D})$$

$$\text{m-wta} \xrightarrow[{[\![\cdot]\!]}]{} \quad \mathrm{Rec}(\Sigma, \mathcal{A}_{\mathbb{D}})$$

$\mathrm{Rec}(\Sigma, X)$: class of recognizable weighted tree languages over X
$\mathrm{Def}(\Sigma, X)$: class of definable weighted tree languages over X

Theorem

For every regular product tv-monoid $\mathbb{D} = (D, +, \diamond, 0, 1, \mathrm{Val})$

$$\text{tv-wta} \xrightarrow{[\![\cdot]\!]} \mathrm{Rec}(\Sigma, \mathbb{D})$$

$$\text{m-wta} \xrightarrow[{[\![\cdot]\!]}]{} \pi_1^2(\mathrm{Rec}(\Sigma, \mathcal{A}_{\mathbb{D}}))$$

$\mathrm{Rec}(\Sigma, X)$: class of recognizable weighted tree languages over X
$\mathrm{Def}(\Sigma, X)$: class of definable weighted tree languages over X

Theorem

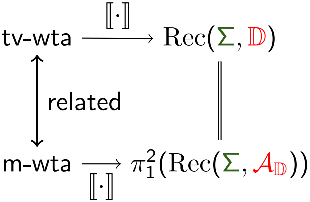
For every regular product tv-monoid $\mathbb{D} = (D, +, \diamond, 0, 1, \mathrm{Val})$

$$\begin{array}{ccc}
\text{tv-wta} & \xrightarrow{[\![\cdot]\!]} & \mathrm{Rec}(\Sigma, \mathbb{D}) \\
\updownarrow \text{ related} & & \\
\text{m-wta} & \xrightarrow[{[\![\cdot]\!]}]{} & \pi_1^2(\mathrm{Rec}(\Sigma, \mathcal{A}_{\mathbb{D}}))
\end{array}$$

$\mathrm{Rec}(\Sigma, X)$: class of recognizable weighted tree languages over X
$\mathrm{Def}(\Sigma, X)$: class of definable weighted tree languages over X

Theorem

For every regular product tv-monoid $\mathbb{D} = (D, +, \diamond, 0, 1, \mathrm{Val})$



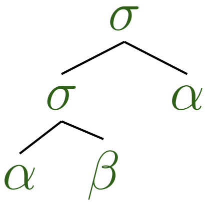
$\mathrm{Rec}(\Sigma, X)$: class of recognizable weighted tree languages over X
$\mathrm{Def}(\Sigma, X)$: class of definable weighted tree languages over X

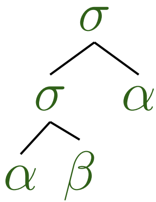
Automata Equivalence

$$\mathcal{N} = (Q, \mu, F) \quad \overset{\text{related}}{\longleftrightarrow} \quad \mathcal{M} = (Q, \delta, F)$$

tv-wta over $\mathbb{D}$ — m-wta over $\mathcal{A}_{\mathbb{D}}$

$$\mu_\sigma(q_1 \ldots q_k, q) = d \quad \text{iff} \quad \delta_\sigma(q_1 \ldots q_k, q) = \mathrm{valtop}_d{}^{(k)}$$



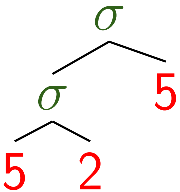


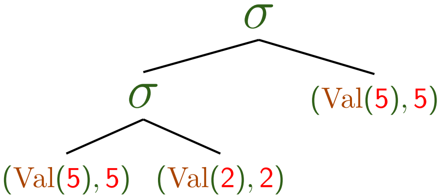
Automata Equivalence

$\mathcal{N} = (Q, \mu, F)$ $\overset{\text{related}}{\longleftrightarrow}$ $\mathcal{M} = (Q, \delta, F)$

tv-wta over $\mathbb{D}$ — m-wta over $\mathcal{A}_{\mathbb{D}}$

$$\mu_\sigma(q_1 \dots q_k, q) = d \quad \text{iff} \quad \delta_\sigma(q_1 \dots q_k, q) = \mathrm{valtop}_d{}^{(k)}$$



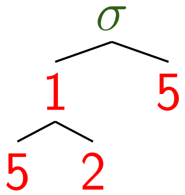


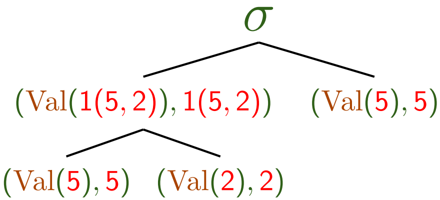
Automata Equivalence

$$\mathcal{N} = (Q, \mu, F) \quad \overset{\text{related}}{\longleftrightarrow} \quad \mathcal{M} = (Q, \delta, F)$$

tv-wta over $\mathbb{D}$ — m-wta over $\mathcal{A}_{\mathbb{D}}$

$$\mu_\sigma(q_1 \dots q_k, q) = d \quad \text{iff} \quad \delta_\sigma(q_1 \dots q_k, q) = \mathrm{valtop}_d{}^{(k)}$$



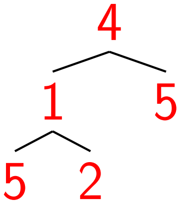


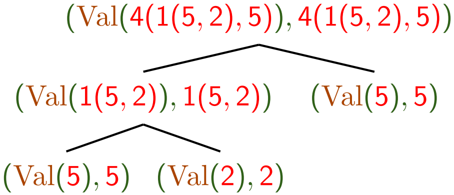
Automata Equivalence

$\mathcal{N} = (Q, \mu, F)$ $\overset{\text{related}}{\longleftrightarrow}$ $\mathcal{M} = (Q, \delta, F)$

tv-wta over $\mathbb{D}$ m-wta over $\mathcal{A}_{\mathbb{D}}$

$$\mu_\sigma(q_1 \ldots q_k, q) = d \quad \text{iff} \quad \delta_\sigma(q_1 \ldots q_k, q) = \mathrm{valtop}_d{}^{(k)}$$



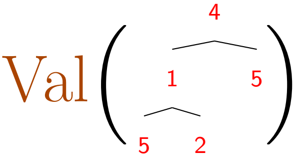


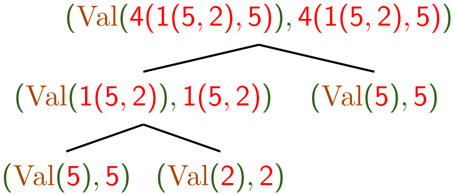
Automata Equivalence

$\mathcal{N} = (Q, \mu, F)$ $\overset{\text{related}}{\longleftrightarrow}$ $\mathcal{M} = (Q, \delta, F)$

tv-wta over $\mathbb{D}$ — m-wta over $\mathcal{A}_{\mathbb{D}}$

$$\mu_\sigma(q_1 \ldots q_k, q) = d \quad \text{iff} \quad \delta_\sigma(q_1 \ldots q_k, q) = \mathrm{valtop}_d{}^{(k)}$$



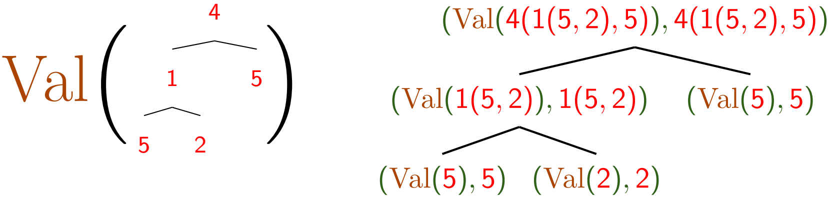


Automata Equivalence

$\mathcal{N} = (Q, \mu, F)$ $\overset{\text{related}}{\longleftrightarrow}$ $\mathcal{M} = (Q, \delta, F)$

tv-wta over $\mathbb{D}$ — m-wta over $\mathcal{A}_{\mathbb{D}}$

$$\mu_\sigma(q_1 \dots q_k, q) = d \quad \text{iff} \quad \delta_\sigma(q_1 \dots q_k, q) = \mathrm{valtop}_d{}^{(k)}$$



$$[\![\mathcal{N}]\!] = \pi_1^2([\![\mathcal{M}]\!])$$

Theorem

For every regular product tv-monoid $\mathbb{D} = (D, +, \diamond, 0, 1, \mathrm{Val})$

$$
\begin{array}{ccc}
\text{tv-wta} & \xrightarrow{[\![\cdot]\!]} & \mathrm{Rec}(\Sigma, \mathbb{D}) \\
\updownarrow \text{related} & & \| \\
\text{m-wta} & \xrightarrow[{[\![\cdot]\!]}]{} & \pi_1^2(\mathrm{Rec}(\Sigma, \mathcal{A}_{\mathbb{D}}))
\end{array}
$$

$\mathrm{Rec}(\Sigma, X)$: class of recognizable weighted tree languages over X
$\mathrm{Def}(\Sigma, X)$: class of definable weighted tree languages over X

Theorem

For every regular product tv-monoid $\mathbb{D} = (D, +, \diamond, 0, 1, \mathrm{Val})$

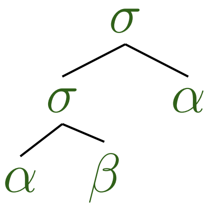
$$\begin{array}{ccc} \text{tv-wta} & \xrightarrow{[\![\cdot]\!]} & \mathrm{Rec}(\Sigma, \mathbb{D}) \\ \updownarrow \text{related} & & \| \\ \text{m-wta} & \xrightarrow[{[\![\cdot]\!]}]{} & \pi_1^2(\mathrm{Rec}(\Sigma, \mathcal{A}_{\mathbb{D}})) \end{array}$$

tv-mso formulas

m-expressions

$\mathrm{Rec}(\Sigma, X)$: class of recognizable weighted tree languages over X
$\mathrm{Def}(\Sigma, X)$: class of definable weighted tree languages over X

Multioperator Expressions [FSV12]



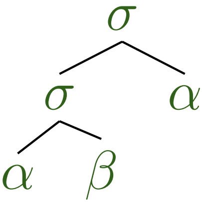
m-expressions over Σ and m-monoid $\mathcal{A}$:

- $H(\omega)$ where $\omega = (\omega_\sigma \in \Omega \mid \sigma \in \Sigma_{\mathcal{U}})$
- $(e_1 + e_2)$
- $\sum_x e$, $\sum_X e$
- $(\varphi \rhd e)$ where $\varphi \in \mathrm{MSO}(\Sigma)$

language defined by e : $[\![e]\!] \colon T_\Sigma \to A$

class of m-expression definable tree languages: $\mathrm{Def}(\Sigma, \mathcal{A})$

Multioperator Expressions [FSV12]



$\sum_x (\mathrm{root}(x) \rhd H(\omega))$

m-expressions over Σ and m-monoid $\mathcal{A}$:

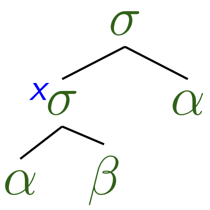
- $H(\omega)$ where $\omega = (\omega_\sigma \in \Omega \mid \sigma \in \Sigma_{\mathcal{U}})$
- $(e_1 + e_2)$
- $\sum_x e$, $\sum_X e$
- $(\varphi \rhd e)$ where $\varphi \in \mathrm{MSO}(\Sigma)$

language defined by e : $[\![e]\!] : T_\Sigma \to A$

class of m-expression definable tree languages: $\mathrm{Def}(\Sigma, \mathcal{A})$

Multioperator Expressions

[FSV12]



$\sum_x (\mathrm{root}(x) \rhd H(\omega))$

m-expressions over Σ and m-monoid $\mathcal{A}$:

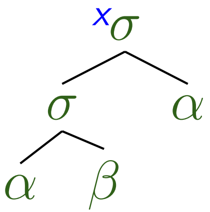
- $H(\omega)$ where $\omega = (\omega_\sigma \in \Omega \mid \sigma \in \Sigma_{\mathcal{U}})$
- $(e_1 + e_2)$
- $\sum_x e$, $\sum_X e$
- $(\varphi \rhd e)$ where $\varphi \in \mathrm{MSO}(\Sigma)$

language defined by e : $[\![e]\!] \colon T_\Sigma \to A$

class of m-expression definable tree languages: $\mathrm{Def}(\Sigma, \mathcal{A})$

Multioperator Expressions

[FSV12]



$\sum_x (\mathrm{root}(\mathrm{x}) \rhd H(\omega))$

m-expressions over Σ and m-monoid $\mathcal{A}$:

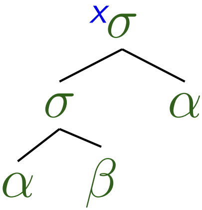
- $H(\omega)$ where $\omega = (\omega_\sigma \in \Omega \mid \sigma \in \Sigma_{\mathcal{U}})$
- $(e_1 + e_2)$
- $\sum_x e$, $\sum_X e$
- $(\varphi \rhd e)$ where $\varphi \in \mathrm{MSO}(\Sigma)$

language defined by e : $[\![e]\!] \colon T_\Sigma \to A$

class of m-expression definable tree languages: $\mathrm{Def}(\Sigma, \mathcal{A})$

Multioperator Expressions

[FSV12]



$\sum_x (\mathrm{root}(x) \rhd H(\omega))$

$$\omega_{(\alpha,\emptyset)}() = 5$$
$$\omega_{(\beta,\emptyset)}() = 2$$
$$\omega_{(\sigma,\emptyset)}(n_1, n_2) = n_1 + n_2$$
$$\omega_{(\sigma,\{x\})}(n_1, n_2) = \max\{n_1, n_2\}$$

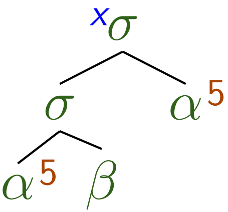
m-expressions over Σ and m-monoid $\mathcal{A}$:

- $H(\omega)$ where $\omega = (\omega_\sigma \in \Omega \mid \sigma \in \Sigma_{\mathcal{U}})$
- $(e_1 + e_2)$
- $\sum_x e$, $\sum_X e$
- $(\varphi \rhd e)$ where $\varphi \in \mathrm{MSO}(\Sigma)$

language defined by e : $[\![e]\!] \colon T_\Sigma \to A$

class of m-expression definable tree languages: $\mathrm{Def}(\Sigma, \mathcal{A})$

Multioperator Expressions [FSV12]



$\sum_x (\mathrm{root}(x) \rhd H(\omega))$

$$\omega_{(\alpha,\emptyset)}() = 5$$
$$\omega_{(\beta,\emptyset)}() = 2$$
$$\omega_{(\sigma,\emptyset)}(n_1, n_2) = n_1 + n_2$$
$$\omega_{(\sigma,\{x\})}(n_1, n_2) = \max\{n_1, n_2\}$$

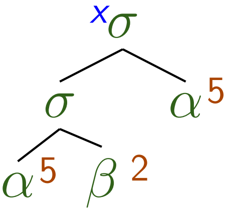
m-expressions over Σ and m-monoid $\mathcal{A}$:

- $H(\omega)$ where $\omega = (\omega_\sigma \in \Omega \mid \sigma \in \Sigma_{\mathcal{U}})$
- $(e_1 + e_2)$
- $\sum_x e$, $\sum_X e$
- $(\varphi \rhd e)$ where $\varphi \in \mathrm{MSO}(\Sigma)$

language defined by e : $[\![e]\!] \colon T_\Sigma \to A$

class of m-expression definable tree languages: $\mathrm{Def}(\Sigma, \mathcal{A})$

Multioperator Expressions [FSV12]



$\sum_x(\mathrm{root}(x) \triangleright H(\omega))$

$$\omega_{(\alpha,\emptyset)}() = 5$$
$$\omega_{(\beta,\emptyset)}() = 2$$
$$\omega_{(\sigma,\emptyset)}(n_1, n_2) = n_1 + n_2$$
$$\omega_{(\sigma,\{x\})}(n_1, n_2) = \max\{n_1, n_2\}$$

m-expressions over Σ and m-monoid $\mathcal{A}$:

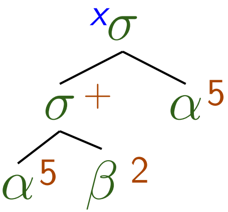
- $H(\omega)$ where $\omega = (\omega_\sigma \in \Omega \mid \sigma \in \Sigma_{\mathcal{U}})$
- $(e_1 + e_2)$
- $\sum_x e$, $\sum_X e$
- $(\varphi \triangleright e)$ where $\varphi \in \mathrm{MSO}(\Sigma)$

language defined by e : $[\![e]\!] \colon T_\Sigma \to A$

class of m-expression definable tree languages: $\mathrm{Def}(\Sigma, \mathcal{A})$

Multioperator Expressions

[FSV12]



$\sum_x(\mathrm{root}(x) \rhd H(\omega))$

$$\omega_{(\alpha,\emptyset)}() = 5$$
$$\omega_{(\beta,\emptyset)}() = 2$$
$$\omega_{(\sigma,\emptyset)}(n_1, n_2) = n_1 + n_2$$
$$\omega_{(\sigma,\{x\})}(n_1, n_2) = \max\{n_1, n_2\}$$

m-expressions over Σ and m-monoid $\mathcal{A}$:

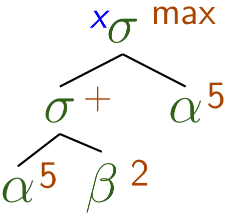
- $H(\omega)$ where $\omega = (\omega_\sigma \in \Omega \mid \sigma \in \Sigma_{\mathcal{U}})$
- $(e_1 + e_2)$
- $\sum_x e$, $\sum_X e$
- $(\varphi \rhd e)$ where $\varphi \in \mathrm{MSO}(\Sigma)$

language defined by e : $[\![e]\!] \colon T_\Sigma \to A$

class of m-expression definable tree languages: $\mathrm{Def}(\Sigma, \mathcal{A})$

Multioperator Expressions

[FSV12]



$\sum_x(\mathrm{root}(\mathrm{x}) \triangleright H(\omega))$

$$\omega_{(\alpha,\emptyset)}() = 5$$
$$\omega_{(\beta,\emptyset)}() = 2$$
$$\omega_{(\sigma,\emptyset)}(n_1, n_2) = n_1 + n_2$$
$$\omega_{(\sigma,\{x\})}(n_1, n_2) = \max\{n_1, n_2\}$$

m-expressions over Σ and m-monoid $\mathcal{A}$:

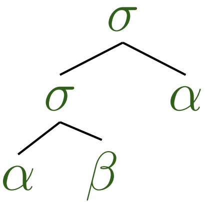
- $H(\omega)$ where $\omega = (\omega_\sigma \in \Omega \mid \sigma \in \Sigma_{\mathcal{U}})$
- $(e_1 + e_2)$
- $\sum_x e$, $\sum_X e$
- $(\varphi \triangleright e)$ where $\varphi \in \mathrm{MSO}(\Sigma)$

language defined by e : $[\![e]\!]: T_\Sigma \to A$

class of m-expression definable tree languages: $\mathrm{Def}(\Sigma, \mathcal{A})$

Tree Valuation MSO

[DGMM11]



$\mathbb{D} = (D, +, \diamond, 0, 1, \mathrm{Val})$
tv-mso over Σ and ptv-monoid $\mathbb{D}$:

$$\varphi ::= d \mid \beta \mid \varphi \vee \varphi \mid \varphi \wedge \varphi \mid \exists x.\varphi \mid \forall x.\varphi \mid \exists X.\varphi$$

$\beta \in \mathrm{MSO}(\Sigma)$ - Boolean formula

language defined by φ: $[\![\varphi]\!]: T_\Sigma \to D$

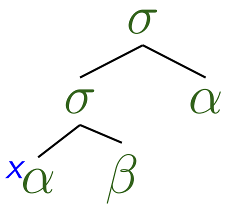
class of syntactically restricted tv-mso
definable tree languages: $\mathrm{Def}(\Sigma, \mathbb{D})$

Example:

$$\forall x.\Big(\mathrm{label}_\alpha(x) \wedge 5 \ \vee \ \mathrm{label}_\beta(x) \wedge 2 \ \vee \ \mathrm{label}_\sigma(x) \wedge 7\Big)$$

Tree Valuation MSO

[DGMM11]



$\mathbb{D} = (D, +, \diamond, 0, 1, \mathrm{Val})$
tv-mso over Σ and ptv-monoid $\mathbb{D}$:

$$\varphi ::= d \mid \beta \mid \varphi \vee \varphi \mid \varphi \wedge \varphi \mid \exists x.\varphi \mid \forall x.\varphi \mid \exists X.\varphi$$

$\beta \in \mathrm{MSO}(\Sigma)$ - Boolean formula

language defined by φ: $[\![\varphi]\!] : T_\Sigma \to D$

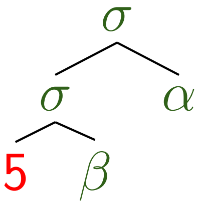
class of syntactically restricted tv-mso definable tree languages: $\mathrm{Def}(\Sigma, \mathbb{D})$

Example:

$$\forall x. \Big(\mathrm{label}_\alpha(x) \wedge 5 \;\vee\; \mathrm{label}_\beta(x) \wedge 2 \;\vee\; \mathrm{label}_\sigma(x) \wedge 7 \Big)$$

Tree Valuation MSO

[DGMM11]



$\mathbb{D} = (D, +, \diamond, 0, 1, \mathrm{Val})$

tv-mso over Σ and ptv-monoid $\mathbb{D}$:

$$\varphi ::= d \mid \beta \mid \varphi \vee \varphi \mid \varphi \wedge \varphi \mid \exists x.\varphi \mid \forall x.\varphi \mid \exists X.\varphi$$

$\beta \in \textsc{mso}(\Sigma)$ - Boolean formula

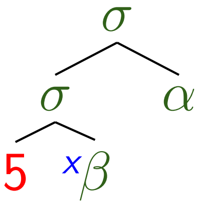
language defined by φ: $[\![\varphi]\!] : T_\Sigma \to D$

class of syntactically restricted tv-mso definable tree languages: $\mathrm{Def}(\Sigma, \mathbb{D})$

Example:

$$\forall x.\Big(\mathrm{label}_\alpha(x) \wedge 5 \;\vee\; \mathrm{label}_\beta(x) \wedge 2 \;\vee\; \mathrm{label}_\sigma(x) \wedge 7\Big)$$

Tree Valuation MSO [DGMM11]



$\mathbb{D} = (D, +, \diamond, 0, 1, \mathrm{Val})$
tv-mso over Σ and ptv-monoid $\mathbb{D}$:

$$\varphi ::= d \mid \beta \mid \varphi \vee \varphi \mid \varphi \wedge \varphi \mid \exists x.\varphi \mid \forall x.\varphi \mid \exists X.\varphi$$

$\beta \in \mathrm{MSO}(\Sigma)$ - Boolean formula

language defined by φ: $[\![\varphi]\!] \colon T_\Sigma \to D$

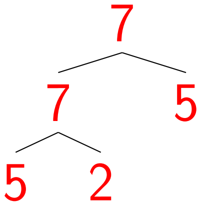
class of syntactically restricted tv-mso
definable tree languages: $\mathrm{Def}(\Sigma, \mathbb{D})$

Example:

$$\forall x.\Big(\mathrm{label}_\alpha(x) \wedge 5 \;\vee\; \mathrm{label}_\beta(x) \wedge 2 \;\vee\; \mathrm{label}_\sigma(x) \wedge 7\Big)$$

Tree Valuation MSO

[DGMM11]



$\mathbb{D} = (D, +, \diamond, 0, 1, \mathrm{Val})$

tv-mso over Σ and ptv-monoid $\mathbb{D}$:

$$\varphi ::= d \mid \beta \mid \varphi \vee \varphi \mid \varphi \wedge \varphi \mid \exists x.\varphi \mid \forall x.\varphi \mid \exists X.\varphi$$

$\beta \in \mathrm{MSO}(\Sigma)$ - Boolean formula

language defined by φ: $[\![\varphi]\!]\colon T_\Sigma \to D$

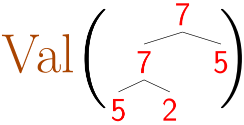
class of syntactically restricted tv-mso definable tree languages: $\mathrm{Def}(\Sigma, \mathbb{D})$

Example:

$$\forall x.\Big(\mathrm{label}_\alpha(x) \wedge 5 \;\vee\; \mathrm{label}_\beta(x) \wedge 2 \;\vee\; \mathrm{label}_\sigma(x) \wedge 7\Big)$$

Tree Valuation MSO

[DGMM11]



$\mathbb{D} = (D, +, \diamond, 0, 1, \mathrm{Val})$

tv-mso over Σ and ptv-monoid $\mathbb{D}$:

$$\varphi ::= d \mid \beta \mid \varphi \vee \varphi \mid \varphi \wedge \varphi \mid \exists x.\varphi \mid \forall x.\varphi \mid \exists X.\varphi$$

$\beta \in \mathrm{MSO}(\Sigma)$ - Boolean formula

language defined by φ: $[\![\varphi]\!] \colon T_\Sigma \to D$

class of syntactically restricted tv-mso definable tree languages: $\mathrm{Def}(\Sigma, \mathbb{D})$

Example:

$$\forall x.\Big(\mathrm{label}_\alpha(x) \wedge 5 \;\vee\; \mathrm{label}_\beta(x) \wedge 2 \;\vee\; \mathrm{label}_\sigma(x) \wedge 7\Big)$$

Theorem

For every regular product tv-monoid $\mathbb{D} = (D, +, \diamond, 0, 1, \mathrm{Val})$

$$\begin{array}{ccc} \text{tv-wta} & \xrightarrow{[\![\cdot]\!]} & \mathrm{Rec}(\Sigma, \mathbb{D}) \\ \updownarrow \text{ related} & & \| \\ \text{m-wta} & \xrightarrow[{[\![\cdot]\!]}]{} & \pi_1^2(\mathrm{Rec}(\Sigma, \mathcal{A}_{\mathbb{D}})) \end{array}$$

tv-mso formulas

m-expressions

$\mathrm{Rec}(\Sigma, X)$: class of recognizable weighted tree languages over X
$\mathrm{Def}(\Sigma, X)$: class of definable weighted tree languages over X

Theorem

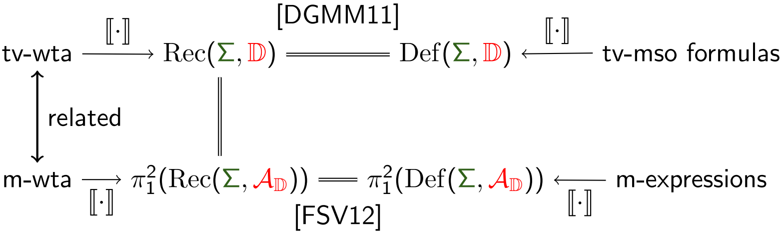
For every regular product tv-monoid $\mathbb{D} = (D, +, \diamond, 0, 1, \mathrm{Val})$

$$
\begin{array}{ccccccc}
\text{tv-wta} & \xrightarrow{[\![\cdot]\!]} & \mathrm{Rec}(\Sigma, \mathbb{D}) & \qquad & \mathrm{Def}(\Sigma, \mathbb{D}) & \xleftarrow{[\![\cdot]\!]} & \text{tv-mso formulas} \\
\big\updownarrow \text{related} & & \| & & & & \\
\text{m-wta} & \xrightarrow[{[\![\cdot]\!]}]{} & \pi_1^2(\mathrm{Rec}(\Sigma, \mathcal{A}_{\mathbb{D}})) & & \pi_1^2(\mathrm{Def}(\Sigma, \mathcal{A}_{\mathbb{D}})) & \xleftarrow[{[\![\cdot]\!]}]{} & \text{m-expressions}
\end{array}
$$

$\mathrm{Rec}(\Sigma, X)$: class of recognizable weighted tree languages over X
$\mathrm{Def}(\Sigma, X)$: class of definable weighted tree languages over X

Theorem

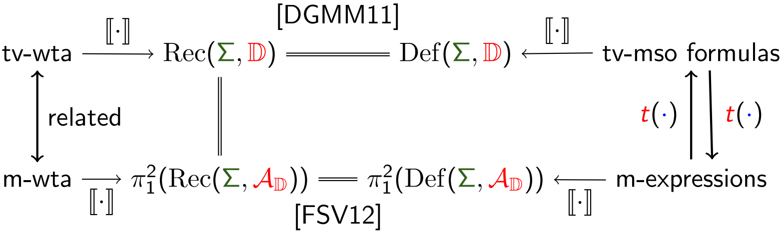
For every regular product tv-monoid $\mathbb{D} = (D, +, \diamond, 0, 1, \mathrm{Val})$



$\mathrm{Rec}(\Sigma, X)$: class of recognizable weighted tree languages over X
$\mathrm{Def}(\Sigma, X)$: class of definable weighted tree languages over X

Theorem

For every regular product tv-monoid $\mathbb{D} = (D, +, \diamond, 0, 1, \mathrm{Val})$



$\mathrm{Rec}(\Sigma, X)$: class of recognizable weighted tree languages over X
$\mathrm{Def}(\Sigma, X)$: class of definable weighted tree languages over X

Tree Valuation MSO $\rightsquigarrow$ Multioperator Expressions

$t(\exists x.([\mathrm{root}(x) \wedge 2] \vee 5))$

Tree Valuation MSO $\rightsquigarrow$ Multioperator Expressions

$t(\exists x.([\mathrm{root}(x) \wedge 2] \vee 5))$

$= \sum_x (t([\mathrm{root}(x) \wedge 2] \vee 5))$

Tree Valuation MSO $\rightsquigarrow$ Multioperator Expressions

$$t(\exists x.([\mathrm{root}(x) \wedge 2] \vee 5))$$
$$= \sum_x (t([\mathrm{root}(x) \wedge 2] \vee 5))$$
$$= \sum_x (t(\mathrm{root}(\mathrm{x}) \wedge 2) + t(5))$$

Tree Valuation MSO $\rightsquigarrow$ Multioperator Expressions

$$t(\exists x.([\mathrm{root}(x) \wedge 2] \vee 5))$$
$$= \sum_x (t([\mathrm{root}(x) \wedge 2] \vee 5))$$
$$= \sum_x (t(\mathrm{root}(\mathrm{x}) \wedge 2) + t(5))$$
$$= \sum_x ((\mathrm{root}(\mathrm{x}) \triangleright t(2)) + t(5))$$

Tree Valuation MSO $\rightsquigarrow$ Multioperator Expressions

$$t(\exists x.([\mathrm{root}(x) \wedge 2] \vee 5))$$
$$= \sum_x (t([\mathrm{root}(x) \wedge 2] \vee 5))$$
$$= \sum_x (t(\mathrm{root}(\mathrm{x}) \wedge 2) + t(5))$$
$$= \sum_x ((\mathrm{root}(\mathrm{x}) \triangleright t(2)) + t(5))$$

For every $d \in D$:

$t(d)$ simulates the automaton $\mathcal{N}_d$
recognizing d for any input

Tree Valuation MSO ⇝ Multioperator Expressions

$t(\exists x.([\mathrm{root}(x) \wedge 2] \vee 5))$
$= \sum_x (t([\mathrm{root}(x) \wedge 2] \vee 5))$
$= \sum_x (t(\mathrm{root}(\mathrm{x}) \wedge 2) + t(5))$
$= \sum_x ((\mathrm{root}(\mathrm{x}) \triangleright t(2)) + t(5))$

For every $d \in D$:

$t(d)$ simulates the automaton $\mathcal{N}_d$ recognizing d for any input

Tree Valuation MSO $\rightsquigarrow$ Multioperator Expressions

$t(\exists x.([\mathrm{root}(x) \wedge 2] \vee 5))$
$= \sum_x (t([\mathrm{root}(x) \wedge 2] \vee 5))$
$= \sum_x (t(\mathrm{root}(\mathrm{x}) \wedge 2) + t(5))$
$= \sum_x ((\mathrm{root}(\mathrm{x}) \triangleright t(2)) + t(5))$

For every $d \in D$:

$t(d)$ simulates the automaton $\mathcal{N}_d$ recognizing d for any input

$t(\forall x.\psi)$:

Tree Valuation MSO $\rightsquigarrow$ Multioperator Expressions

$t(\exists x.([\mathrm{root}(x) \wedge 2] \vee 5))$

$= \sum_x (t([\mathrm{root}(x) \wedge 2] \vee 5))$

$= \sum_x (t(\mathrm{root}(x) \wedge 2) + t(5))$

$= \sum_x ((\mathrm{root}(x) \triangleright t(2)) + t(5))$

For every $d \in D$:

$t(d)$ simulates the automaton $\mathcal{N}_d$ recognizing d for any input

$t(\forall x.\psi)$:

$\mathrm{step}(\psi) = (d_1, \psi_1), \ldots, (d_n, \psi_n)$

Tree Valuation MSO $\rightsquigarrow$ Multioperator Expressions

$t(\exists x.([\mathrm{root}(x) \land 2] \lor 5))$
$= \sum_x (t([\mathrm{root}(x) \land 2] \lor 5))$
$= \sum_x (t(\mathrm{root}(\mathrm{x}) \land 2) + t(5))$
$= \sum_x ((\mathrm{root}(\mathrm{x}) \triangleright t(2)) + t(5))$

For every $d \in D$:

$t(d)$ simulates the automaton $\mathcal{N}_d$ recognizing d for any input

$t(\forall x.\psi)$:
$\mathrm{step}(\psi) = (d_1, \psi_1), \ldots, (d_n, \psi_n)$
$[\![\psi]\!](\xi) = \sum_{i \in [n]} \mathbb{1}_{\mathcal{L}(\psi_i)}(\xi) \diamond d_i$

Tree Valuation MSO $\rightsquigarrow$ Multioperator Expressions

$t(\exists x.([\mathrm{root}(x) \wedge 2] \vee 5))$

$= \sum_x (t([\mathrm{root}(x) \wedge 2] \vee 5))$

$= \sum_x (t(\mathrm{root}(\mathrm{x}) \wedge 2) + t(5))$

$= \sum_x ((\mathrm{root}(\mathrm{x}) \triangleright t(2)) + t(5))$

For every $d \in D$:

$t(d)$ simulates the automaton $\mathcal{N}_d$ recognizing d for any input

$t(\forall x.\psi)$:

$\mathrm{step}(\psi) = (d_1, \psi_1), \ldots, (d_n, \psi_n)$

$[\![\psi]\!](\xi) = \sum_{i \in [n]} \mathbb{1}_{\mathcal{L}(\psi_i)}(\xi) \diamond d_i$

$t(\forall x.\psi)$

Tree Valuation MSO $\rightsquigarrow$ Multioperator Expressions

$t(\exists x.([\mathrm{root}(x) \wedge 2] \vee 5))$
$= \sum_x (t([\mathrm{root}(x) \wedge 2] \vee 5))$
$= \sum_x (t(\mathrm{root}(\mathrm{x}) \wedge 2) + t(5))$
$= \sum_x ((\mathrm{root}(\mathrm{x}) \triangleright t(2)) + t(5))$

For every $d \in D$:

$t(d)$ simulates the automaton $\mathcal{N}_d$ recognizing d for any input

$t(\forall x.\psi)$:
$\mathrm{step}(\psi) = (d_1, \psi_1), \ldots, (d_n, \psi_n)$
$[\![\psi]\!](\xi) = \sum_{i \in [n]} \mathbb{1}_{\mathcal{L}(\psi_i)}(\xi) \diamond d_i$

$t(\forall x.\psi)$
$= \sum_{X_1} \ldots \sum_{X_n}$

Tree Valuation MSO $\rightsquigarrow$ Multioperator Expressions

$t(\exists x.([\mathrm{root}(x) \wedge 2] \vee 5))$
$= \sum_x (t([\mathrm{root}(x) \wedge 2] \vee 5))$
$= \sum_x (t(\mathrm{root(x)} \wedge 2) + t(5))$
$= \sum_x ((\mathrm{root(x)} \triangleright t(2)) + t(5))$

For every $d \in D$:

$t(d)$ simulates the automaton $\mathcal{N}_d$ recognizing d for any input

$t(\forall x.\psi)$:
$\mathrm{step}(\psi) = (d_1, \psi_1), \ldots, (d_n, \psi_n)$
$[\![\psi]\!](\xi) = \sum_{i \in [n]} \mathbb{1}_{\mathcal{L}(\psi_i)}(\xi) \diamond d_i$

$t(\forall x.\psi)$
$= \sum_{X_1} \ldots \sum_{X_n}$
$(\forall x.(\bigwedge_{i \in [n]} (x \in X_i) \Leftrightarrow \psi_i))$

Tree Valuation MSO $\rightsquigarrow$ Multioperator Expressions

$t(\exists x.([\mathrm{root}(x) \wedge 2] \vee 5))$
$= \sum_x (t([\mathrm{root}(x) \wedge 2] \vee 5))$
$= \sum_x (t(\mathrm{root}(\mathrm{x}) \wedge 2) + t(5))$
$= \sum_x ((\mathrm{root}(\mathrm{x}) \triangleright t(2)) + t(5))$

For every $d \in D$:

$t(d)$ simulates the automaton $\mathcal{N}_d$ recognizing d for any input

$t(\forall x.\psi)$:
$\mathrm{step}(\psi) = (d_1, \psi_1), \dots, (d_n, \psi_n)$
$[\![\psi]\!](\xi) = \sum_{i \in [n]} \mathbb{1}_{\mathcal{L}(\psi_i)}(\xi) \diamond d_i$

$t(\forall x.\psi)$
$= \sum_{X_1} \dots \sum_{X_n}$
$(\forall x.(\bigwedge_{i \in [n]} (x \in X_i) \Leftrightarrow \psi_i))$
$\triangleright H(\omega)$

Tree Valuation MSO $\rightsquigarrow$ Multioperator Expressions

$$t(\exists x.([\mathrm{root}(x) \wedge 2] \vee 5))$$
$$= \textstyle\sum_x (t([\mathrm{root}(x) \wedge 2] \vee 5))$$
$$= \textstyle\sum_x (t(\mathrm{root}(\mathrm{x}) \wedge 2) + t(5))$$
$$= \textstyle\sum_x ((\mathrm{root}(\mathrm{x}) \triangleright t(2)) + t(5))$$

For every $d \in D$:

$t(d)$ simulates the automaton $\mathcal{N}_d$ recognizing d for any input

$t(\forall x.\psi)$:
$$\mathrm{step}(\psi) = (d_1, \psi_1), \ldots, (d_n, \psi_n)$$
$$[\![\psi]\!](\xi) = \textstyle\sum_{i \in [n]} \mathbb{1}_{\mathcal{L}(\psi_i)}(\xi) \diamond d_i$$

$$t(\forall x.\psi)$$
$$= \textstyle\sum_{X_1} \ldots \sum_{X_n}$$
$$(\forall x.(\textstyle\bigwedge_{i \in [n]} (x \in X_i) \Leftrightarrow \psi_i))$$
$$\triangleright H(\omega)$$
$$\omega_{(\sigma, U)} = \mathrm{valtop}_{d_U}{}^{(k)}$$

Tree Valuation MSO $\rightsquigarrow$ Multioperator Expressions

$t(\exists x.([\mathrm{root}(x) \wedge 2] \vee 5))$

$= \sum_x (t([\mathrm{root}(x) \wedge 2] \vee 5))$

$= \sum_x (t(\mathrm{root}(x) \wedge 2) + t(5))$

$= \sum_x ((\mathrm{root}(x) \triangleright t(2)) + t(5))$

For every $d \in D$:

$t(d)$ simulates the automaton $\mathcal{N}_d$ recognizing d for any input

$t(\forall x.\psi)$:

$\mathrm{step}(\psi) = (d_1, \psi_1), \ldots, (d_n, \psi_n)$

$[\![\psi]\!](\xi) = \sum_{i \in [n]} \mathbb{1}_{\mathcal{L}(\psi_i)}(\xi) \diamond d_i$

$t(\forall x.\psi)$

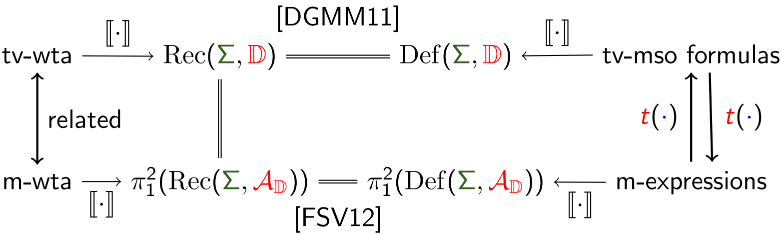
$$= \sum_{X_1} \ldots \sum_{X_n} (\forall x.(\bigwedge_{i \in [n]} (x \in X_i) \Leftrightarrow \psi_i)) \triangleright H(\omega)$$

$\omega_{(\sigma, U)} = \mathrm{valtop}_{d_U}{}^{(k)}$

$$d_U = \sum_{\substack{i \in [n] \\ X_i \in U}} d_i$$

Theorem

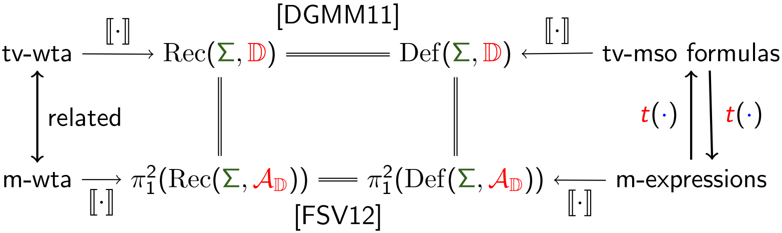
For every regular product tv-monoid $\mathbb{D} = (D, +, \diamond, 0, 1, \mathrm{Val})$



$\mathrm{Rec}(\Sigma, X)$: class of recognizable weighted tree languages over X
$\mathrm{Def}(\Sigma, X)$: class of definable weighted tree languages over X

Theorem

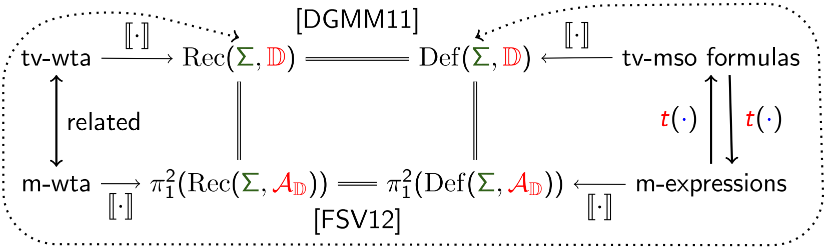
For every regular product tv-monoid $\mathbb{D} = (D, +, \diamond, 0, 1, \mathrm{Val})$



$\mathrm{Rec}(\Sigma, X)$: class of recognizable weighted tree languages over X
$\mathrm{Def}(\Sigma, X)$: class of definable weighted tree languages over X

Theorem

For every regular product tv-monoid $\mathbb{D} = (D, +, \diamond, 0, 1, \mathrm{Val})$



$\mathrm{Rec}(\Sigma, X)$: class of recognizable weighted tree languages over X
$\mathrm{Def}(\Sigma, X)$: class of definable weighted tree languages over X

Related Work

- Transformations for different logic restrictions
  - simplifies constructions for some cases
- Simulation of M-Languages by TV-Languages
  - based on [Droste, Götze, Märcker, Meinecke (2011)]
  - no equality, i.e. $\mathrm{Rec}(\Sigma, \mathcal{A}) \subset \mathrm{Rec}(\Sigma, \mathbb{D}_{\mathcal{A}})$

List of References

Manfred Droste, Doreen Götze, Steffen Märcker, and Ingmar Meinecke. "Weighted Tree Automata over Valuation Monoids and Their Characterization by Weighted Logics". In: *Algebraic Foundations in Computer Science*. Ed. by Werner Kuich and George Rahonis. Vol. 7020. Lecture Notes in Computer Science. Springer Berlin Heidelberg, 2011, pp. 30–55.

Zoltán Fülöp, Torsten Stüber, and Heiko Vogler. "A Büchi-like theorem for weighted tree automata over multioperator monoids". In: *Theory of Computing Systems* 50(2) (2012). Published online 28.10.2010, pp. 241–278.

Multioperator Expressions $\rightsquigarrow$ Tree Valuation MSO

- $t(H(\omega)) = \forall x.\psi^{H(\omega)}$, where $\omega_{(\sigma,U)} = \mathrm{valtop}_d{}^{(k)}$

Multioperator Expressions $\rightsquigarrow$ Tree Valuation MSO

- $t(H(\omega)) = \forall x.\psi^{H(\omega)}$, where $\omega_{(\sigma,U)} = \mathrm{valtop}_d{}^{(k)}$

$$\psi^{H(\omega)} = \bigvee_{\sigma\in\Sigma} \bigvee_{V_1\subseteq\mathcal{U}_1} \bigvee_{V_2\subseteq\mathcal{U}_2} \psi^{\mathrm{value}}_{\sigma,V_1,V_2,\omega}$$

Multioperator Expressions $\rightsquigarrow$ Tree Valuation MSO

- $t(H(\omega)) = \forall x . \psi^{H(\omega)}$, where $\omega_{(\sigma,U)} = \mathrm{valtop}_{d}{}^{(k)}$

$$\psi^{H(\omega)} = \bigvee_{\sigma\in\Sigma} \bigvee_{V_1\subseteq\mathcal{U}_1} \bigvee_{V_2\subseteq\mathcal{U}_2} \psi^{\mathrm{value}}_{\sigma,V_1,V_2,\omega}$$

$$\psi^{\mathrm{value}}_{\sigma,V_1,V_2,\omega} = \psi^{\mathrm{label}}_{\sigma,V_1,V_2} \wedge d_{(\sigma,V_1\cup V_2)}$$

Multioperator Expressions $\rightsquigarrow$ Tree Valuation MSO

- $t(H(\omega)) = \forall x.\psi^{H(\omega)}$, where $\omega_{(\sigma,U)} = \mathrm{valtop}_d{}^{(k)}$

$$\psi^{H(\omega)} = \bigvee_{\sigma\in\Sigma} \bigvee_{V_1\subseteq\mathcal{U}_1} \bigvee_{V_2\subseteq\mathcal{U}_2} \psi^{\mathrm{value}}_{\sigma,V_1,V_2,\omega}$$

$$\psi^{\mathrm{value}}_{\sigma,V_1,V_2,\omega} = \psi^{\mathrm{label}}_{\sigma,V_1,V_2} \wedge d_{(\sigma,V_1\cup V_2)}$$

$$\psi^{\mathrm{label}}_{\sigma,V_1,V_2} = \mathrm{label}_\sigma(x) \wedge \bigwedge_{v\in V_1} x = v \wedge \bigwedge_{v\in\mathcal{U}_1\setminus V_1} x \neq v \wedge \bigwedge_{X\in V_2} x \in X \wedge \bigwedge_{X\in\mathcal{U}_2\setminus V_2} x \notin X$$

Multioperator Expressions $\rightsquigarrow$ Tree Valuation MSO

- $t(H(\omega)) = \forall x.\psi^{H(\omega)}$, where $\qquad \omega_{(\sigma,U)} = \mathrm{valtop}_d{}^{(k)}$

$$\psi^{H(\omega)} = \bigvee_{\sigma\in\Sigma} \bigvee_{V_1\subseteq\mathcal{U}_1} \bigvee_{V_2\subseteq\mathcal{U}_2} \psi^{\mathrm{value}}_{\sigma,V_1,V_2,\omega}$$

$$\psi^{\mathrm{value}}_{\sigma,V_1,V_2,\omega} = \psi^{\mathrm{label}}_{\sigma,V_1,V_2} \wedge d_{(\sigma,V_1\cup V_2)}$$

$$\psi^{\mathrm{label}}_{\sigma,V_1,V_2} = \mathrm{label}_\sigma(x) \wedge \bigwedge_{v\in V_1} x = v \wedge \bigwedge_{v\in\mathcal{U}_1\setminus V_1} x \neq v \wedge \bigwedge_{X\in V_2} x \in X \wedge \bigwedge_{X\in\mathcal{U}_2\setminus V_2} x \notin X$$

- $t(e_1 + e_2) = t(e_1) \vee t(e_2)$,

Multioperator Expressions $\rightsquigarrow$ Tree Valuation MSO

- $t(H(\omega)) = \forall x.\psi^{H(\omega)}$, where $\qquad \omega_{(\sigma,U)} = \mathrm{valtop}_d{}^{(k)}$

$$\psi^{H(\omega)} = \bigvee_{\sigma\in\Sigma} \bigvee_{V_1\subseteq\mathcal{U}_1} \bigvee_{V_2\subseteq\mathcal{U}_2} \psi^{\mathrm{value}}_{\sigma,V_1,V_2,\omega}$$

$$\psi^{\mathrm{value}}_{\sigma,V_1,V_2,\omega} = \psi^{\mathrm{label}}_{\sigma,V_1,V_2} \wedge d_{(\sigma,V_1\cup V_2)}$$

$$\psi^{\mathrm{label}}_{\sigma,V_1,V_2} = \mathrm{label}_\sigma(x) \wedge \bigwedge_{v\in V_1} x = v \wedge \bigwedge_{v\in\mathcal{U}_1\setminus V_1} x \neq v \\ \wedge \bigwedge_{X\in V_2} x \in X \wedge \bigwedge_{X\in\mathcal{U}_2\setminus V_2} x \notin X$$

- $t(e_1 + e_2) = t(e_1) \vee t(e_2)$,
- $t(\sum_x e') = \exists x.t(e')$,

Multioperator Expressions $\rightsquigarrow$ Tree Valuation MSO

- $t(H(\omega)) = \forall x.\psi^{H(\omega)}$, where $\qquad \omega_{(\sigma,U)} = \mathrm{valtop}_d{}^{(k)}$

$$\psi^{H(\omega)} = \bigvee_{\sigma\in\Sigma} \bigvee_{V_1\subseteq\mathcal{U}_1} \bigvee_{V_2\subseteq\mathcal{U}_2} \psi^{\mathrm{value}}_{\sigma,V_1,V_2,\omega}$$

$$\psi^{\mathrm{value}}_{\sigma,V_1,V_2,\omega} = \psi^{\mathrm{label}}_{\sigma,V_1,V_2} \wedge d_{(\sigma,V_1\cup V_2)}$$

$$\psi^{\mathrm{label}}_{\sigma,V_1,V_2} = \mathrm{label}_\sigma(x) \wedge \bigwedge_{v\in V_1} x = v \wedge \bigwedge_{v\in\mathcal{U}_1\setminus V_1} x \neq v \wedge \bigwedge_{X\in V_2} x \in X \wedge \bigwedge_{X\in\mathcal{U}_2\setminus V_2} x \notin X$$

- $t(e_1 + e_2) = t(e_1) \vee t(e_2)$,
- $t(\sum_x e') = \exists x.t(e')$,
- $t(\sum_X e') = \exists X.t(e')$, and

Multioperator Expressions $\rightsquigarrow$ Tree Valuation MSO

- $t(H(\omega)) = \forall x.\psi^{H(\omega)}$, where $\quad \omega_{(\sigma,U)} = \mathrm{valtop}_d{}^{(k)}$

$$\psi^{H(\omega)} = \bigvee_{\sigma\in\Sigma} \bigvee_{V_1\subseteq\mathcal{U}_1} \bigvee_{V_2\subseteq\mathcal{U}_2} \psi^{\mathrm{value}}_{\sigma,V_1,V_2,\omega}$$

$$\psi^{\mathrm{value}}_{\sigma,V_1,V_2,\omega} = \psi^{\mathrm{label}}_{\sigma,V_1,V_2} \wedge d_{(\sigma,V_1\cup V_2)}$$

$$\psi^{\mathrm{label}}_{\sigma,V_1,V_2} = \mathrm{label}_\sigma(x) \wedge \bigwedge_{v\in V_1} x = v \wedge \bigwedge_{v\in\mathcal{U}_1\setminus V_1} x \neq v \wedge \bigwedge_{X\in V_2} x \in X \wedge \bigwedge_{X\in\mathcal{U}_2\setminus V_2} x \notin X$$

- $t(e_1 + e_2) = t(e_1) \vee t(e_2)$,
- $t(\sum_x e') = \exists x.t(e')$,
- $t(\sum_X e') = \exists X.t(e')$, and
- $t(\varphi \triangleright e') = \varphi \wedge t(e')$.

Tree Valuation MSO $\rightsquigarrow$ Multioperator Expressions

- Boolean formula β: $t(\beta) = \beta \triangleright t(1)$.
- $d \in D$: simulates automaton that recognizes d for any input.
- $t(\varphi_1 \vee \varphi_2) = t(\varphi_1) + t(\varphi_2)$.
- $\varphi = \varphi_1 \wedge \varphi_2$ is strongly $\wedge$-restricted:
  - φ_1 (or φ_2) Boolean: $t(\varphi_1 \wedge \varphi_2) = \varphi_1 \triangleright t(\varphi_2)$
  - φ_1 and φ_2 almost Boolean, i.e. $\mathrm{step}(\varphi_i) = (a^i_1, \psi^i_1) \dots (a^i_n, \psi^i_n)$ then $t(\varphi_1 \wedge \varphi_2) = \sum^{+}_{\substack{i\in[n]\\ j\in[m]}} (\psi^1_i \wedge \psi^2_j) \triangleright t\big(a^1_i \diamond a^2_j\big)$
- $t(\exists x.\psi) = \sum_x t(\psi)$
- $t(\exists X.\psi) = \sum_X t(\psi)$
- $\varphi = \forall x.\psi$ is $\forall$-restricted: ψ almost Boolean, i.e. $\mathrm{step}(\psi) = (d_1, \psi_1) \dots (d_n, \psi_n)$. Let $\mathcal{U} = \{X_1, \dots, X_n\}$ and $(\omega^{\psi})_{(\sigma,U)} = \mathrm{valtop}_{d_U}{}^{(k)}$ where $d_U = \sum_{\substack{i\in[n]\\ X_i\in U}} d_i$

$$t(\forall x.\psi) = \textstyle\sum_{X_1} \dots \sum_{X_n} \big(\forall x.(\bigwedge_{i\in[n]} (x \in X_i) \Leftrightarrow \psi_i)\big) \triangleright H(\omega^{\psi})$$

The Automaton $\mathcal{N}_d$

- Runs of the automaton are simulated by a partition of variables $X_{q_1 \dots q_k, \sigma, q}$ representing the transition taken at each position.
- $\varphi_{\mathrm{partition}}$ checks for a valid run representation
- $\varphi_{\mathrm{final}} = \forall z. \big(\mathrm{root}(z) \Rightarrow \bigwedge_{\substack{X_{q_1 \dots q_k, \sigma, q} \\ q \notin F}} (z \notin X_{q_1 \dots q_k, \sigma, q})\big)$
- We construct: $t(d) = \sum_{X_1} \dots \sum_{X_n} (\varphi_{\mathrm{partition}} \wedge \varphi_{\mathrm{final}}) \triangleright H(\omega^d)$ where $(\omega^d)_{(\sigma, \{X_{q_1 \dots q_k, \sigma, q}\})} = \mathrm{valtop}_{\mu_\sigma(q_1 \dots q_k, q)}$