Compositions of Bottom-Up Tree Series Transformations

Andreas Maletti [a]

Technische Universität Dresden
Fakultät Informatik
D–01062 Dresden, Germany

maletti@tcs.inf.tu-dresden.de

May 27, 2005

1. Motivation
2. Semirings, Tree Series, and Tree Series Substitution
3. Bottom-Up Tree Series Transducers
4. Composition Results

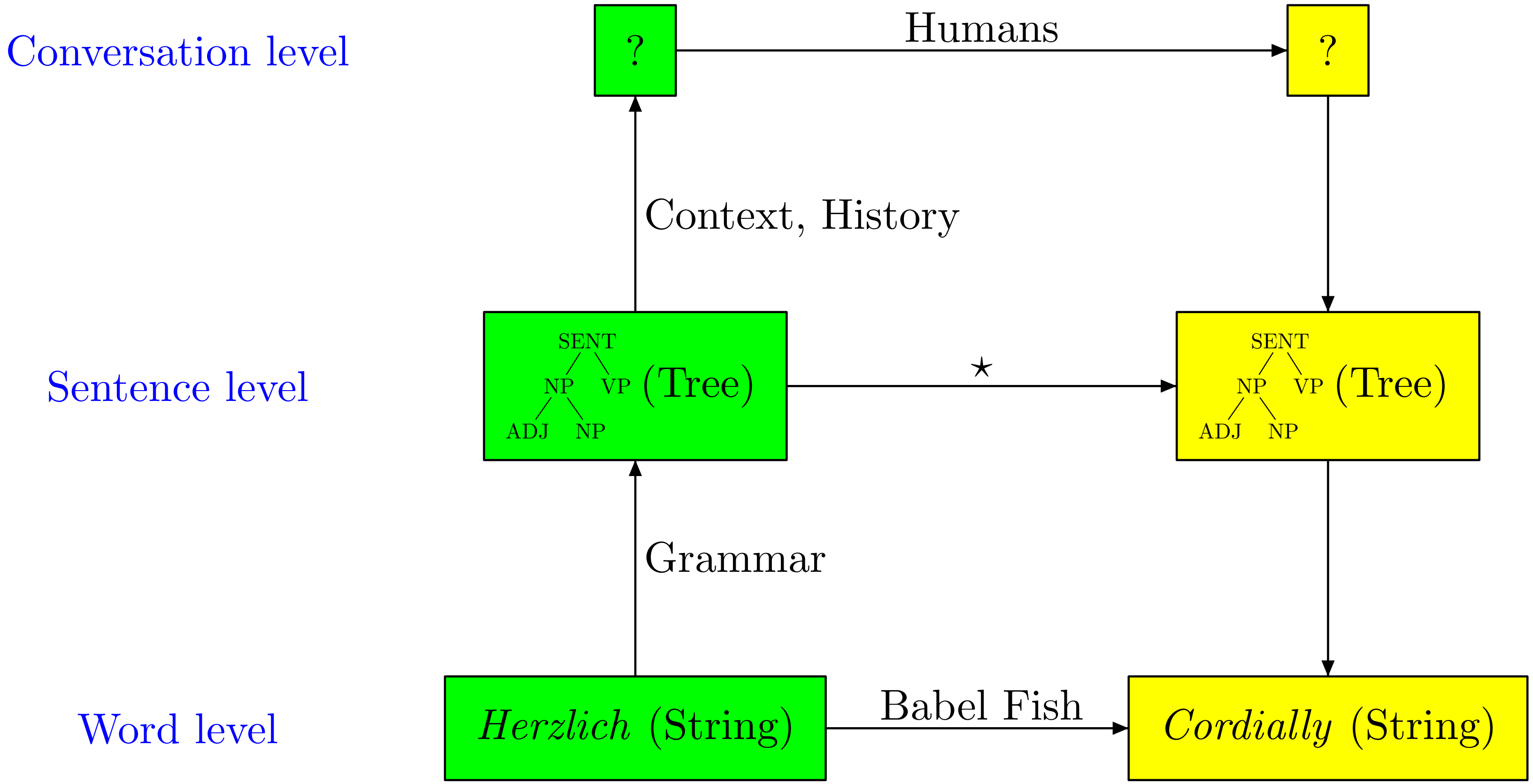
[a] Financially supported by the German Research Foundation (DFG, GK 334)

Motivation

Babel Fish Translation

German	English
Herzlich willkommen meine sehr geehrten Damen und Herren. Ich möchte mich vorab bei den Organisatoren für die vortrefflich geleistete Arbeit bedanken.	Cordially welcomely my very much honoured ladies and gentlemen. I would like to thank you first the supervisors for the splendid carried out work.
Herzlich willkommen meine sehr geehrten Damen und Herren. Ich möchte mich vorab bei den Organisatoren, *die diese Veranstaltung erst ermöglicht haben*, für die vortrefflich geleistete Arbeit bedanken.	Cordially welcomely my very much honoured ladies and gentlemen. I would like *me* first the supervisors, *who made this meeting possible only*, for *whom* splendid carried out work *thank you*.

Motivation


Conversation level
Sentence level
Word level
?
Humans
?
Context, History
SENT
NP
VP
ADJ
NP
(Tree)
⋆
SENT
NP
VP
ADJ
NP
(Tree)
Grammar
Herzlich (String)
Babel Fish
Cordially (String)

Motivation

- Automatic translation is widely used (even Microsoft uses it to translate English documentation into German)

  Der Wert des Punkts, der zurückgegeben wird, wird in Bildschirmkoordinaten angegeben, während das Aufrufen von CWnd::GetCurrentMessage statt eine Nachricht manuell erstellen, sinnvoll schien. Die Quick-Info führt einen Treffertest aus, um wann festzulegen, den Test den Punkt von innerhalb der Grenze des Client-Rechtecks jeder zugeordneter Tools den weitergeleiteten Nachrichtenherbsten fehlzuschlagen und die Quick-Info nicht angezeigt werden.

- Dictionaries are very powerful word-to-word translators; leave few words untranslated

- Outcome is nevertheless usually unhappy and ungrammatical

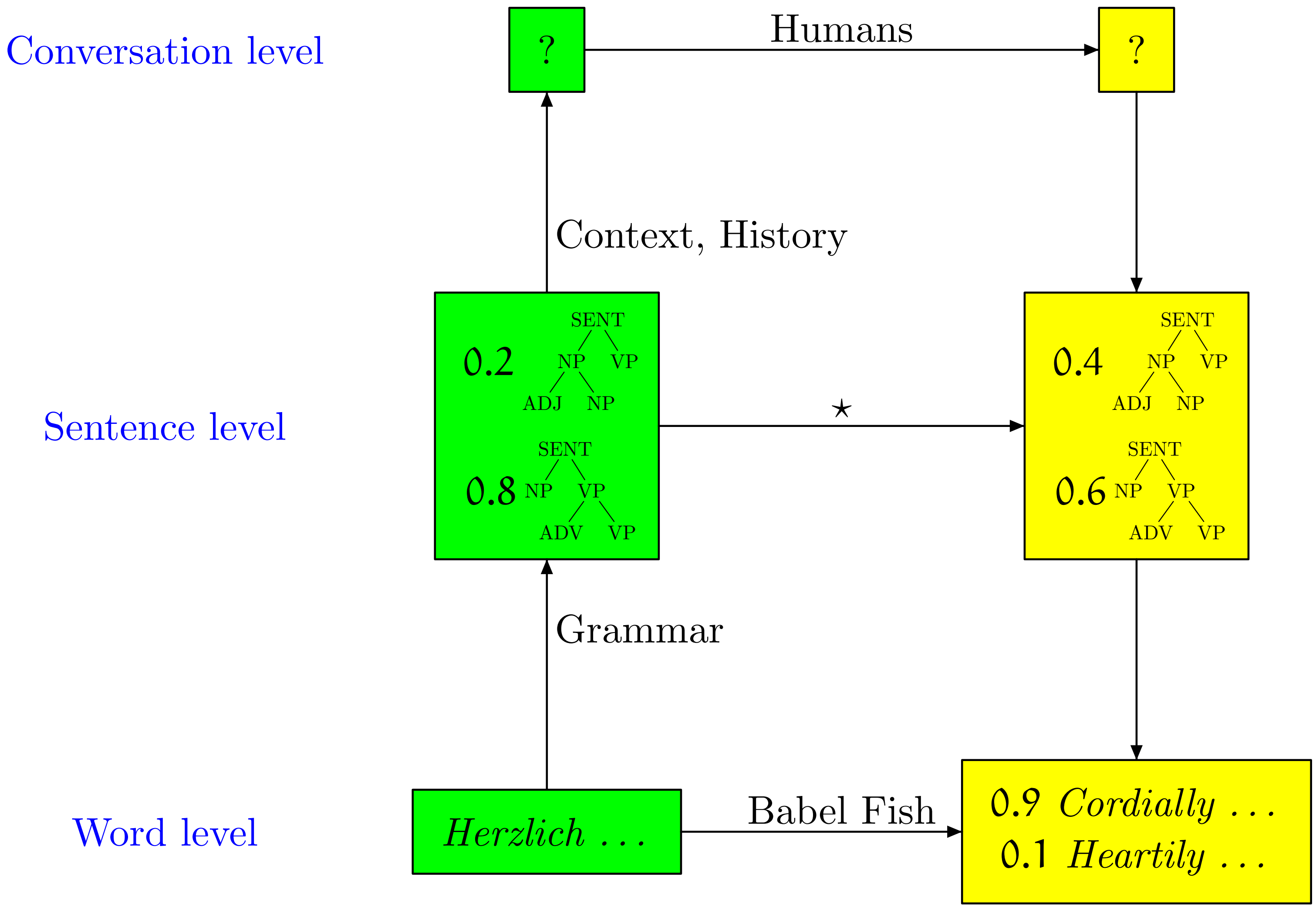
- Post-processing necessary

Motivation

Major problem: Ambiguity of natural language

Common approach:
- "Soft output" (results equipped with a probability)
- Human choses the correct translation among the more likely ones

Motivation


Conversation level
Sentence level
Word level
?
Humans
?
Context, History
0.2
SENT
NP VP
ADJ NP
0.8
SENT
NP VP
ADV VP
⋆
0.4
SENT
NP VP
ADJ NP
0.6
SENT
NP VP
ADV VP
Grammar
Herzlich . . .
Babel Fish
0.9 Cordially . . .
0.1 Heartily . . .

Motivation

Tree series transducers are a straightforward generalization of

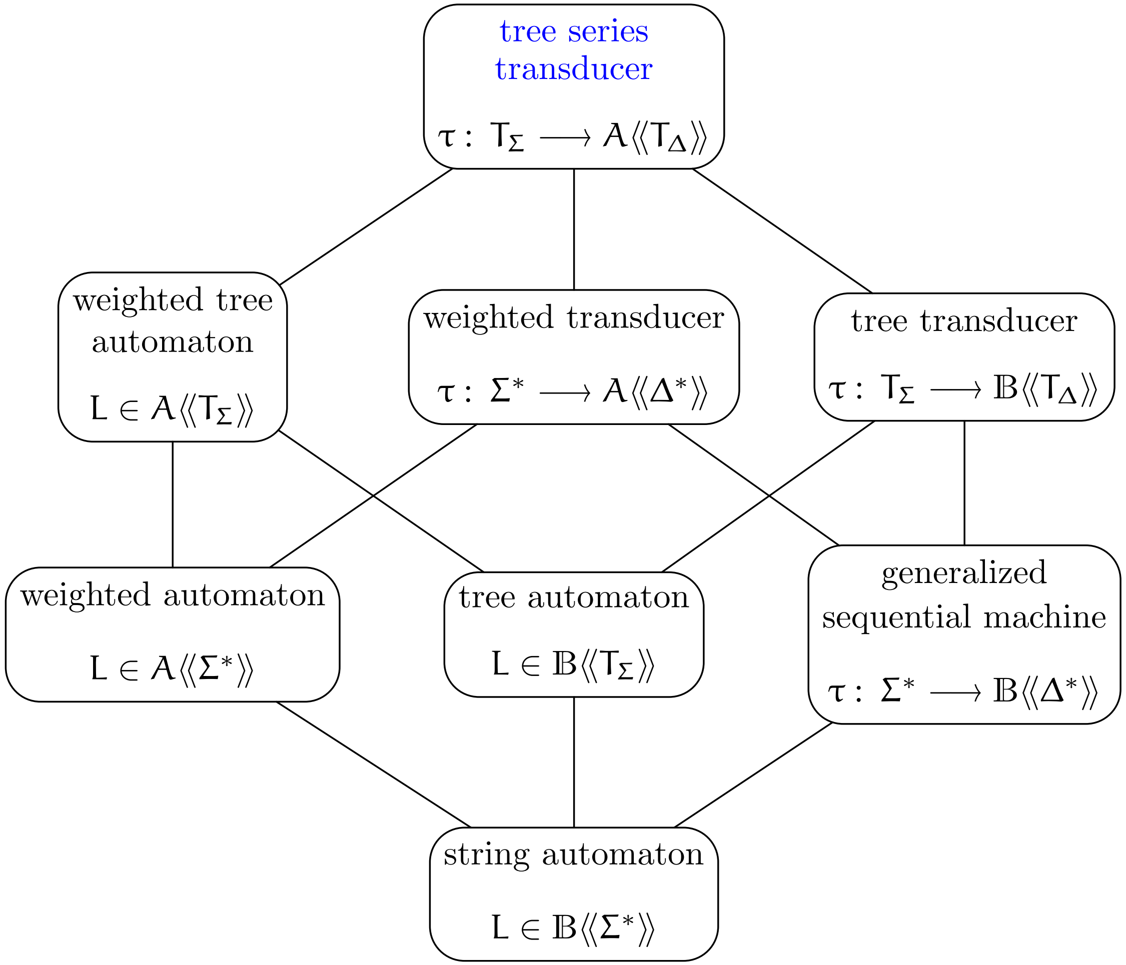
(i) tree transducers, which are applied in

- syntax-directed semantics,
- functional programming, and
- XML querying,

(ii) weighted automata, which are applied in

- (tree) pattern matching,
- image compression and speech-to-text processing.

Generalization Hierarchy

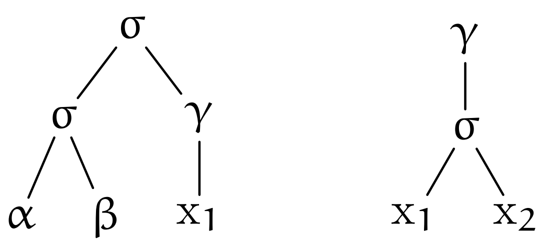


Trees

Σ ranked alphabet, $\Sigma_k \subseteq \Sigma$ symbols of rank k, $X = \{ x_i \mid i \in \mathbb{N}_+ \}$

- $T_\Sigma(X)$ set of Σ-trees indexed by X,
- $T_\Sigma = T_\Sigma(\emptyset)$,
- $t \in T_\Sigma(X)$ is *linear* (resp., *nondeleting*) in $Y \subseteq X$, if every $y \in Y$ occurs at most (resp., at least) once in t,
- $t[t_1, \ldots, t_k]$ denotes the tree substitution of t_i for x_i in t

Examples: $\Sigma = \{\sigma^{(2)}, \gamma^{(1)}, \alpha^{(0)}, \beta^{(0)}\}$ and $Y = \{x_1, x_2\}$



$\sigma\big(\sigma(\alpha, \beta), \gamma(x_1)\big)$ $\quad$ $\gamma\big(\sigma(x_1, x_2)\big)$

Semirings

A *semiring* is an algebraic structure $\mathcal{A} = (A, \oplus, \odot)$

- $(A, \oplus)$ is a commutative monoid with neutral element $\mathbb{0}$,
- $(A, \odot)$ is a monoid with neutral element $\mathbb{1}$,
- $\mathbb{0}$ is absorbing wrt. $\odot$, and
- $\odot$ distributes over $\oplus$ (from left and right).

Examples:

- semiring of non-negative integers $\mathbb{N}_\infty = (\mathbb{N} \cup \{\infty\}, +, \cdot)$
- Boolean semiring $\mathbb{B} = (\{0, 1\}, \vee, \wedge)$
- tropical semiring $\mathbb{T} = (\mathbb{N} \cup \{\infty\}, \min, +)$
- any ring, field, etc.

Properties of Semirings

We say that $\mathcal{A}$ is

- *commutative*, if $\odot$ is commutative,
- *idempotent*, if $a \oplus a = a$,
- *complete*, if there is an operation $\bigoplus_I : A^I \longrightarrow A$ such that
  1. $\bigoplus_{i \in \{m,n\}} a_i = a_m \oplus a_n$,
  2. $\bigoplus_{i \in I} a_i = \bigoplus_{j \in J} \left(\bigoplus_{i \in I_j} a_i \right)$, if $I = \bigcup_{j \in J} I_j$ is a (generalized) partition of I, and
  3. $\left(\bigoplus_{i \in I} a_i \right) \odot \left(\bigoplus_{j \in J} b_j \right) = \bigoplus_{i \in I, j \in J} (a_i \odot b_j)$.

Semiring	Commutative	Idempotent	Complete
$\mathbb{N}_\infty$	YES	no	YES
$\mathbb{B}$	YES	YES	YES
$\mathbb{T}$	YES	YES	YES

Tree Series

$\mathcal{A} = (A, \oplus, \odot)$ semiring, Σ ranked alphabet

Mappings $\varphi : T_\Sigma(X) \longrightarrow A$ are also called *tree series*

- the set of all tree series is $A\langle\langle T_\Sigma(X)\rangle\rangle$,
- the *coefficient* of $t \in T_\Sigma(X)$ in φ, i.e., $\varphi(t)$, is denoted by (φ, t),
- the *sum* is defined pointwise $(\varphi_1 \oplus \varphi_2, t) = (\varphi_1, t) \oplus (\varphi_2, t)$,
- the *support* of φ is $\mathrm{supp}(\varphi) = \{ t \in T_\Sigma(X) \mid (\varphi, t) \neq 0 \}$,
- φ is *linear* (resp., *nondeleting* in $Y \subseteq X$), if $\mathrm{supp}(\varphi)$ is a set of trees, which are linear (resp., nondeleting in Y),
- the series φ with $\mathrm{supp}(\varphi) = \emptyset$ is denoted by $\widetilde{0}$.

Example: $\varphi = 1\ \alpha + 1\ \beta + 3\ \sigma(\alpha, \alpha) + \ldots + 3\ \sigma(\beta, \beta) + 5\ \sigma(\alpha, \sigma(\alpha, \alpha)) + \ldots$

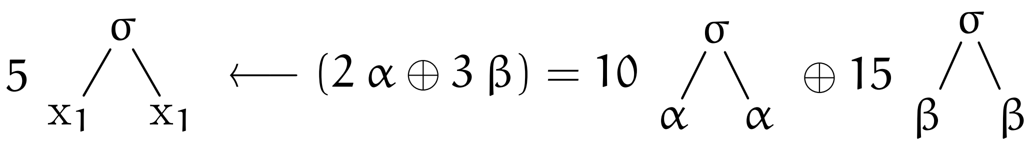
Tree Series Substitution

$\mathcal{A} = (A, \oplus, \odot)$ complete semiring, $\varphi, \psi_1, \ldots, \psi_k \in A\langle\langle T_\Sigma(X)\rangle\rangle$

Pure substitution of $(\psi_1, \ldots, \psi_k)$ into φ:

$$\varphi \longleftarrow (\psi_1, \ldots, \psi_k) = \bigoplus_{\substack{t \in \text{supp}(\varphi), \\ (\forall i \in [k]):\, t_i \in \text{supp}(\psi_i)}} (\varphi, t) \odot (\psi_1, t_1) \odot \cdots \odot (\psi_k, t_k)\; t[t_1, \ldots, t_k]$$

Example: $5\, \sigma(x_1, x_1) \longleftarrow (2\, \alpha \oplus 3\, \beta) = 10\, \sigma(\alpha, \alpha) \oplus 15\, \sigma(\beta, \beta)$



Tree Series Transducers

Definition: A *(bottom-up) tree series transducer* (tst) is a system $M = (Q, \Sigma, \Delta, \mathcal{A}, F, \mu)$

- Q is a non-empty set of *states*,
- Σ and Δ are input and output ranked alphabets,
- $\mathcal{A} = (A, \oplus, \odot)$ is a complete semiring,
- $F \in A\langle\langle T_\Delta(X_1)\rangle\rangle^Q$ is a vector of linear and nondeleting tree series, also called *final output*,
- *tree representation* $\mu = (\,\mu_k\,)_{k\in\mathbb{N}}$ with $\mu_k : \Sigma_k \longrightarrow A\langle\langle T_\Delta(X_k)\rangle\rangle^{Q\times Q^k}$.

If Q is finite and $\mu_k(\sigma)_{q,\vec{q}}$ is polynomial, then M is called *finite*.

Semantics of Tree Series Transducers

Mapping $r : \mathrm{pos}(t) \longrightarrow Q$ is a *run* of M on the input tree $t \in T_\Sigma$

$\mathrm{Run}(t)$ set of all runs on t

Evaluation mapping: $\mathrm{eval}_r : \mathrm{pos}(t) \longrightarrow A\langle\langle T_\Delta \rangle\rangle$ defined for every $k \in \mathbb{N}$, $\mathrm{lab}_t(p) \in \Sigma_k$ by

$$\mathrm{eval}_r(p) = \mu_k(\mathrm{lab}_t(p))_{r(p),r(p\cdot 1)\dots r(p\cdot k)} \longleftarrow \big(\mathrm{eval}_r(p\cdot 1), \dots, \mathrm{eval}_r(p\cdot k)\big)$$

Tree-series transformation induced by M is $\|M\| : A\langle\langle T_\Sigma \rangle\rangle \longrightarrow A\langle\langle T_\Delta \rangle\rangle$ defined

$$\|M\|(\varphi) = \bigoplus_{t \in T_\Sigma} \Big(\bigoplus_{r \in \mathrm{Run}(t)} \mathrm{eval}_r(\varepsilon) \Big)$$

Semantics — Example

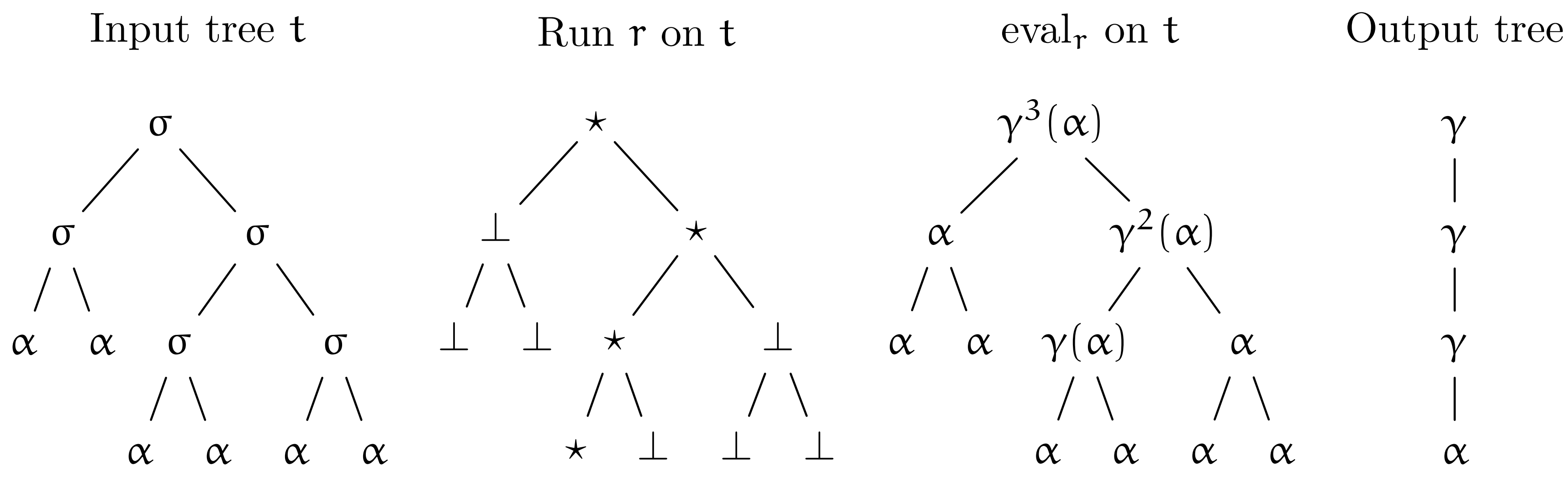
$M = (Q, \Sigma, \Delta, \mathbb{N}_\infty, F, \mu)$ with

- $Q = \{\perp, \star\}$,
- $\Sigma = \{\sigma^{(2)}, \alpha^{(0)}\}$ and $\Delta = \{\gamma^{(1)}, \alpha^{(0)}\}$,
- $F_\perp = \widetilde{0}$ and $F_\star = 1\, x_1$,
- and tree representation

$$\mu_0(\alpha)_\perp = 1\,\alpha \qquad \mu_0(\alpha)_\star = 1\,\alpha$$

$$\mu_2(\sigma)_{\perp,\perp\perp} = 1\,\alpha \qquad \mu_2(\sigma)_{\star,\star\perp} = 1\,x_1 \qquad \mu_2(\sigma)_{\star,\perp\star} = 1\,x_2$$

Semantics — Example (cont.)



$\|M\|(1\ t) = 2\,\gamma(\alpha) \oplus 4\,\gamma^3(\alpha)$

Known Results

- $\text{nl-BOT}_{\text{ts-ts}}(\mathcal{A}) \circ \text{nl-BOT}_{\text{ts-ts}}(\mathcal{A}) = \text{nl-BOT}_{\text{ts-ts}}(\mathcal{A})$
- $\text{nl-BOT}_{\text{ts-ts}}(\mathcal{A}) \circ \text{h-BOT}_{\text{ts-ts}}(\mathcal{A}) \subseteq \text{BOT}_{\text{ts-ts}}(\mathcal{A})$
- $\text{p-BOT}_{\text{ts-ts}}(\mathcal{A}) \subseteq \text{nlp-BOT}_{\text{ts-ts}}(\mathcal{A}) \circ \text{h-BOT}_{\text{ts-ts}}(\mathcal{A})$

Corollary: For every commutative and $\aleph_0$-complete semiring

$$\text{nlp-BOT}_{\text{ts-ts}}(\mathcal{A}) \circ \text{p-BOT}_{\text{ts-ts}}(\mathcal{A}) = \text{p-BOT}_{\text{ts-ts}}(\mathcal{A}) \; .$$

Extension

$(Q, \Sigma, \Delta, \mathcal{A}, F, \mu)$ tree series transducer, $\vec{q} \in Q^k$, $q \in Q$, $\varphi \in A\langle\langle T_\Sigma(X_k)\rangle\rangle$

Definition: We define $h_\mu^{\vec{q}} : T_\Sigma(X_k) \longrightarrow A\langle\langle T_\Delta(X_k)\rangle\rangle^Q$

$$h_\mu^{\vec{q}}(x_i)_q = \begin{cases} 1\, x_i & \text{, if } q = q_i \\ \widetilde{0} & \text{, otherwise} \end{cases}$$

$$h_\mu^{\vec{q}}(\sigma(t_1, \dots, t_k))_q = \bigoplus_{p_1, \dots, p_k \in Q} \mu_k(\sigma)_{q, p_1 \dots p_k} \longleftarrow (h_\mu^{\vec{q}}(t_1)_{p_1}, \dots, h_\mu^{\vec{q}}(t_k)_{p_k})$$

We define $h_\mu^{\vec{q}} : A\langle\langle T_\Sigma(X_k)\rangle\rangle \longrightarrow A\langle\langle T_\Delta(X_k)\rangle\rangle^Q$ by

$$h_\mu^{\vec{q}}(\varphi)_q = \bigoplus_{t \in T_\Sigma(X_k)} (\varphi, t) \odot h_\mu^{\vec{q}}(t)_q$$

Composition Construction

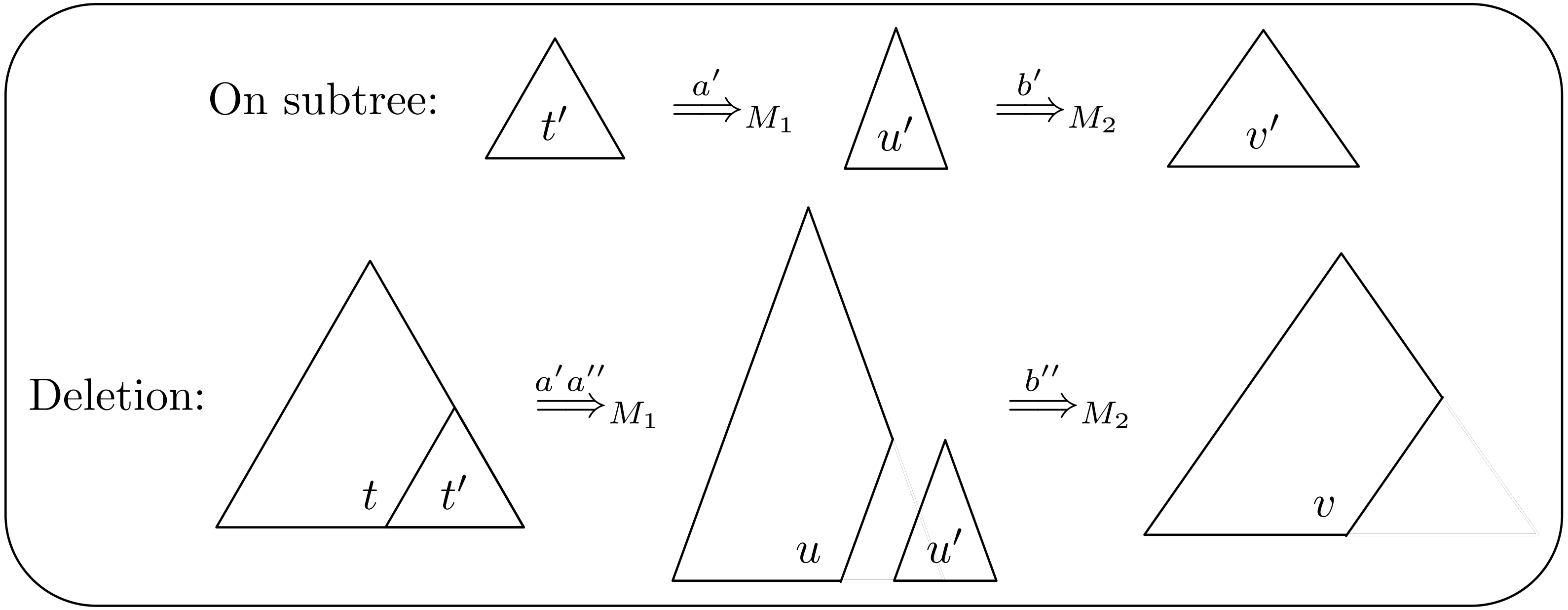
$M_1 = (Q_1, \Sigma, \Delta, \mathcal{A}, F_1, \mu_1)$ and $M_2 = (Q_2, \Delta, \Gamma, \mathcal{A}, F_2, \mu_2)$ tree series transducer

Definition: The *product of* M_1 and M_2, denoted by $M_1 \cdot M_2$, is the tree series transducer

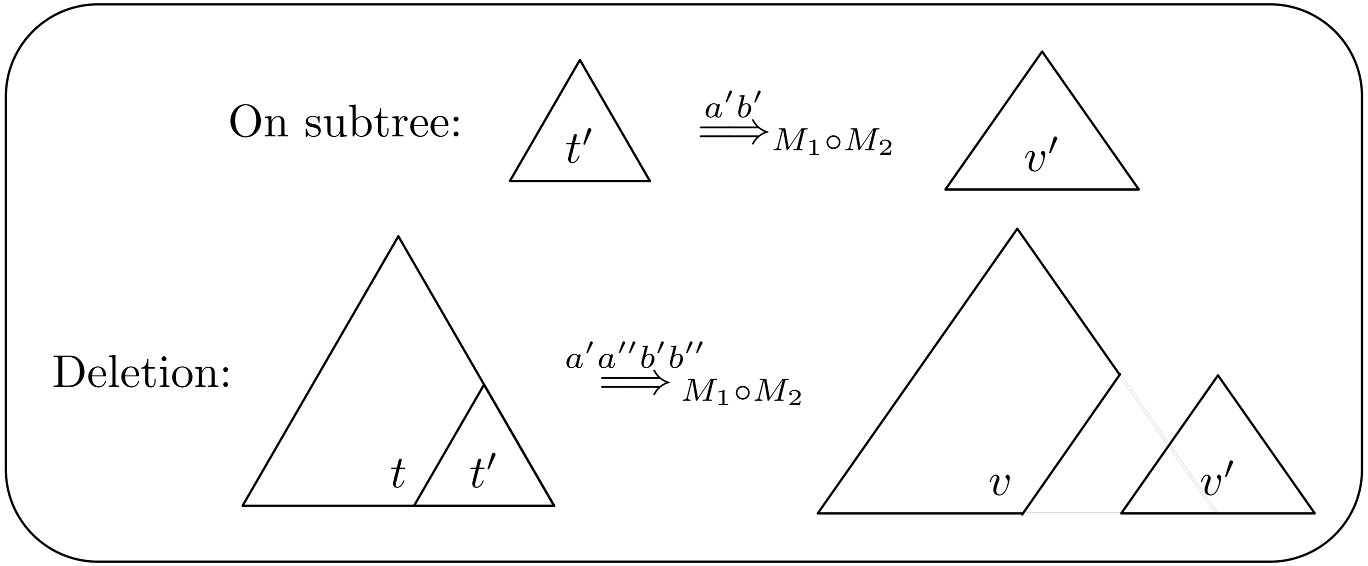
$$M = (Q_1 \times Q_2, \Sigma, \Gamma, \mathcal{A}, F, \mu)$$

- $F_{pq} = \bigoplus_{i \in Q_2} (F_2)_i \longleftarrow h^q_{\mu_2}\big((F_1)_p\big)_i$
- $\mu_k(\sigma)_{pq,(p_1q_1,\ldots,p_kq_k)} = h^{q_1\cdots q_k}_{\mu_2}\big((\mu_1)_k(\sigma)_{p,p_1\ldots p_k}\big)_q.$

Composition



Composition (cont.)



Another Product

Definition:

- $\bot \notin Q_2$, $Q_2' = Q_2 \cup \{\bot\}$, and $\alpha \in \Delta_0$
- construct $M_2' = (Q_2', \Gamma, \Delta, \mathcal{A}, F_2', \mu_2')$ with $(F_2')_q = (F_2)_q$; $q \in Q_2$ and $(F_2')_\bot = \widetilde{0}$ and tree representation μ_2'

$$(\mu_2')_k(\gamma)_{q,q_1\dots q_k} = (\mu_2)_k(\gamma)_{q,q_1\dots q_k}$$
$$(\mu_2')_k(\gamma)_{\bot,\bot\dots\bot} = 1\ \alpha\ .$$

- construct $(M_1 \circ M_2) = (Q_1 \times Q_2', \Sigma, \Delta, \mathcal{A}, F, \mu)$ with

$$F_{(p,q)} = \sum_{q' \in Q_2'} (F_2)_{q'} \longleftarrow \left(h^{q}_{\mu_2'}((F_1)_p)_{q'}\right)$$

$$\mu_k(\sigma)_{(p,q),(p_1,q_1)\dots(p_k,q_k)} = h^{q_1\dots q_k}_{\mu_2'}\Bigg(\sum_{\substack{t \in T_\Gamma(k),\\ (\forall i \in [k]):\, i \notin \mathrm{var}(t) \iff q_i = \bot}} ((\mu_1)_k(\sigma)_{p,p_1\dots p_k}, t)\ t\Bigg)_q$$

$$\mu_k(\sigma)_{(p,\bot),(p_1,\bot)\dots(p_k,\bot)} = h^{\bot\dots\bot}_{\mu_2'}\left((\mu_1)_k(\sigma)_{p,p_1\dots p_k}\right)_\bot$$

Main Theorem

$\mathcal{A}$ commutative and complete semiring

Main Theorem

- $\text{l-BOT}_{\text{ts-ts}}(\mathcal{A}) \circ \text{BOT}_{\text{ts-ts}}(\mathcal{A}) = \text{BOT}_{\text{ts-ts}}(\mathcal{A})$.
- $\text{BOT}_{\text{ts-ts}}(\mathcal{A}) \circ \text{db-BOT}_{\text{ts-ts}}(\mathcal{A}) = \text{BOT}_{\text{ts-ts}}(\mathcal{A})$,
- $\text{BOT}_{\text{ts-ts}}(\mathcal{A}) \circ \text{d-BOT}_{\text{ts-ts}}(\mathcal{A}) = \text{BOT}_{\text{ts-ts}}(\mathcal{A})$, provided that $\mathcal{A}$ is multiplicatively idempotent.

References

[Borchardt 04] B. Borchardt: *Code Selection by Tree Series Transducers*. CIAA'04, Kingston, Canada, 2004. to appear

[Engelfriet et al 02] J. Engelfriet, Z. Fülöp, and H. Vogler: *Bottom-up and Top-down Tree Series Transformations*. Journal of Automata, Languages, and Combinatorics 7:11–70, 2002

[Fülöp et al 03] Z. Fülöp and H. Vogler: *Tree Series Transformations that Respect Copying*. Theory of Computing Systems 36:247–293, 2003

[Kuich 99] W. Kuich: *Tree Transducers and Formal Tree Series*. Acta Cybernetica 14:135–149, 1999