
Collapse of the MTT Hierarchy for Functions of Linear Size Increase

Sebastian Maneth

EPF Lausanne

`sebastian.maneth@epfl.ch`

Motivation of Tree Transducers

- Inspired by Syntax-Directed Compilers:

Attribute Grammars

[Knuth 1968]

Macro Tree Transducer – MTT

[EngelfrietVogler 1985]

- Today: formal model for XML Query Languages

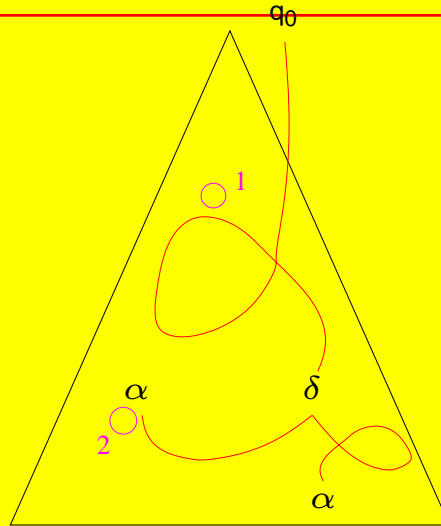
n-Pebble Tree Transducer – n-PTT

[MiloSuciuVianu2003]

Tree Transducer Inclusions

Attributed ATT

n-Pebble n-PTT



Macro MTT
(only down!)

$\langle q, \sigma \rangle (y_1, y_2) \rightarrow$

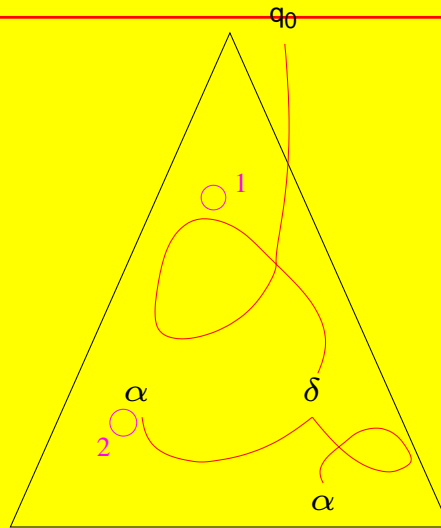
$\langle p, \downarrow_2 \rangle (y_1, \langle p', \downarrow_1 \rangle (y_2))$

$ATT = 0\text{-PTT} \subseteq n\text{-PTT}$
 $\subseteq MTT^n$ [EngelfrietM.2003]

Tree Transducer Inclusions

Attributed ATT

n-Pebble n-PTT



Macro MTT
(only down!)

$\langle q, \sigma \rangle (y_1, y_2) \rightarrow$

$\langle p, \downarrow_2 \rangle (y_1, \langle p', \downarrow_1 \rangle (y_2))$

Thus: XML Queries can be modeled by

compositions of MTTs.

Efficient Macro Tree Transducers

Look at Functions of Linear Size Increase:

$$\exists c \forall s : |\tau(s)| \leq c \cdot |s|.$$

$$\begin{aligned} \text{MTT} \cap \text{LSI} &= \text{MSOTT} \quad [\text{EngelfrietM03}] \\ &= \text{ATT}_{\text{sur}}^{\text{R}} \quad [\text{BloemEngelfriet00}] \\ &= \text{MTT}_{\text{fc}}^{\text{R}} \quad [\text{EngelfrietM99}] \\ &= \text{C}. \end{aligned}$$

Efficient Macro Tree Transducers

Look at Functions of Linear Size Increase:

$$\exists c \forall s : |\tau(s)| \leq c \cdot |s|.$$

Nice Properties of $C := \text{MTT} \cap \text{LSI}$

- linear time computable (w.r.t. size of input tree)
- precomposable with attribute grammars:

$$C \circ \text{ATT}^R \subseteq \text{ATT}^R \quad [\text{GanzingerGiegerich84/88}]$$

Efficient Macro Tree Transducers

Look at Functions of Linear Size Increase:

$$\exists c \forall s : |\tau(s)| \leq c \cdot |s|.$$

Proof of $\text{MTT} \cap \text{LSI} \subseteq \text{MTT}_{fc}^R$ (ca. 3 years):

“If reduced and **NOT** fc ,
then **NOT** linear size increase.” (heavy pumping..)

Efficient Macro Tree Transducers

Look at Functions of Linear Size Increase:

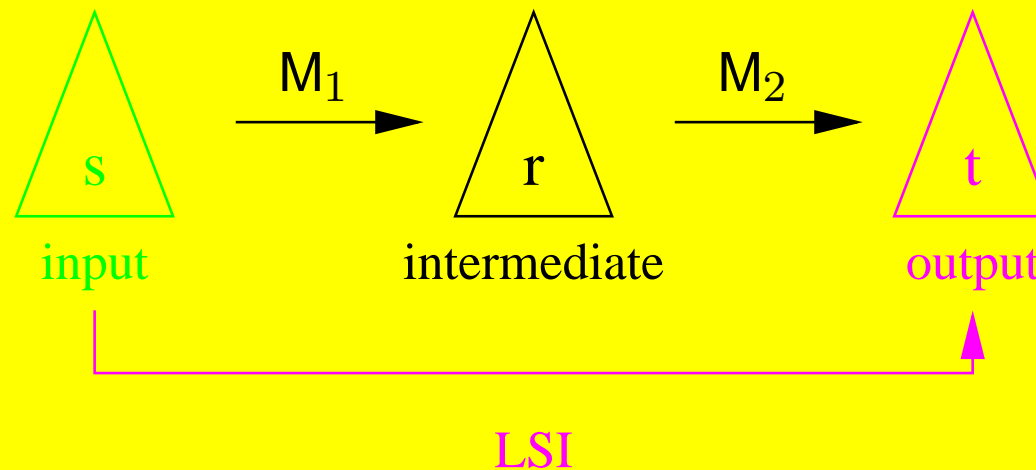
$$\exists c \forall s : |\tau(s)| \leq c \cdot |s|.$$

C is decidable subclass of MTT:

- (1) make the MTT reduced.
- (2) decide the **finite copying (fc)** property.

New Problem: Compositions

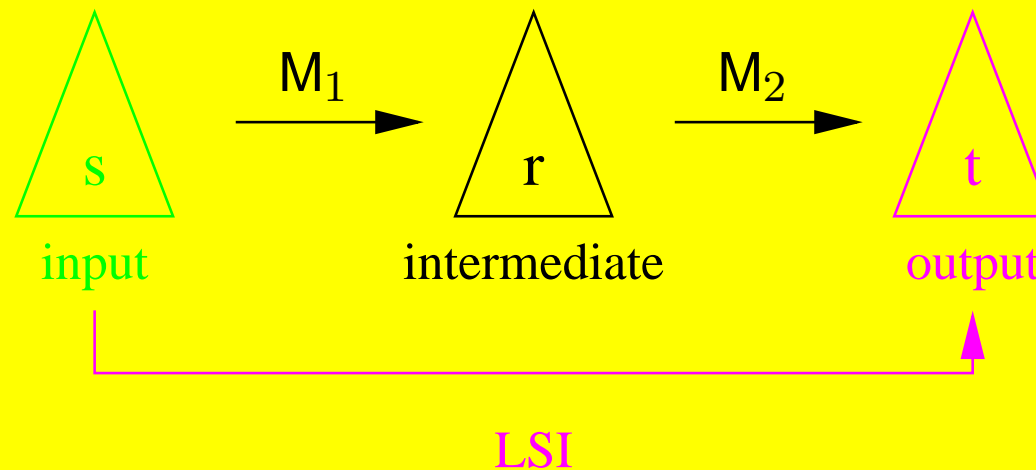
$$\text{MTT}^* \cap \text{LSI} = ???$$



We **CANNOT** reduce M_1 and M_2 and then check finite copying!!

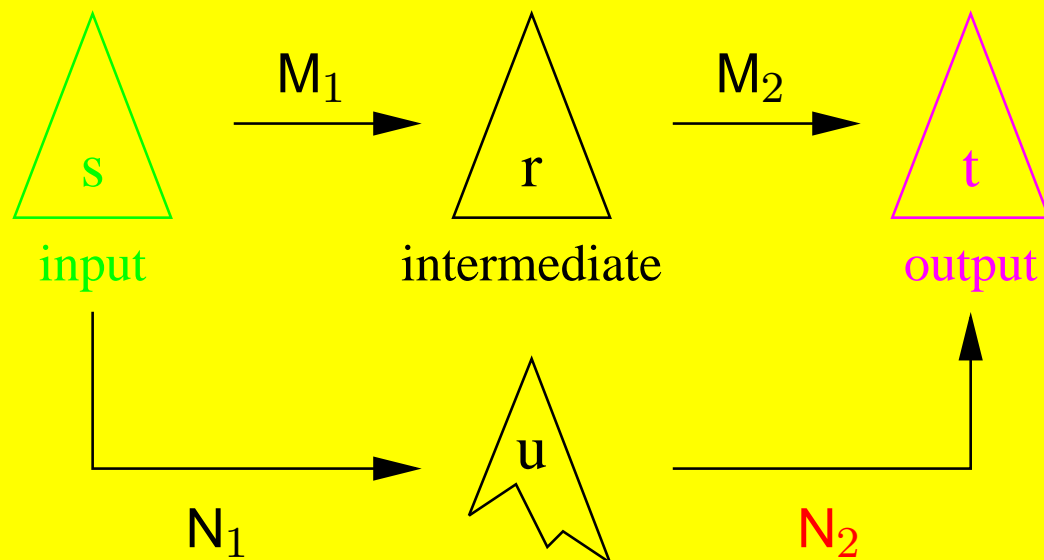
New Problem: Compositions

$$\text{MTT}^* \cap \text{LSI} = ???$$



- WHY?**
- (1) $|r|$ might be NON-linear in $|s|$
 - (2) M_2 need only be LSI on $\text{out}(M_1)$!

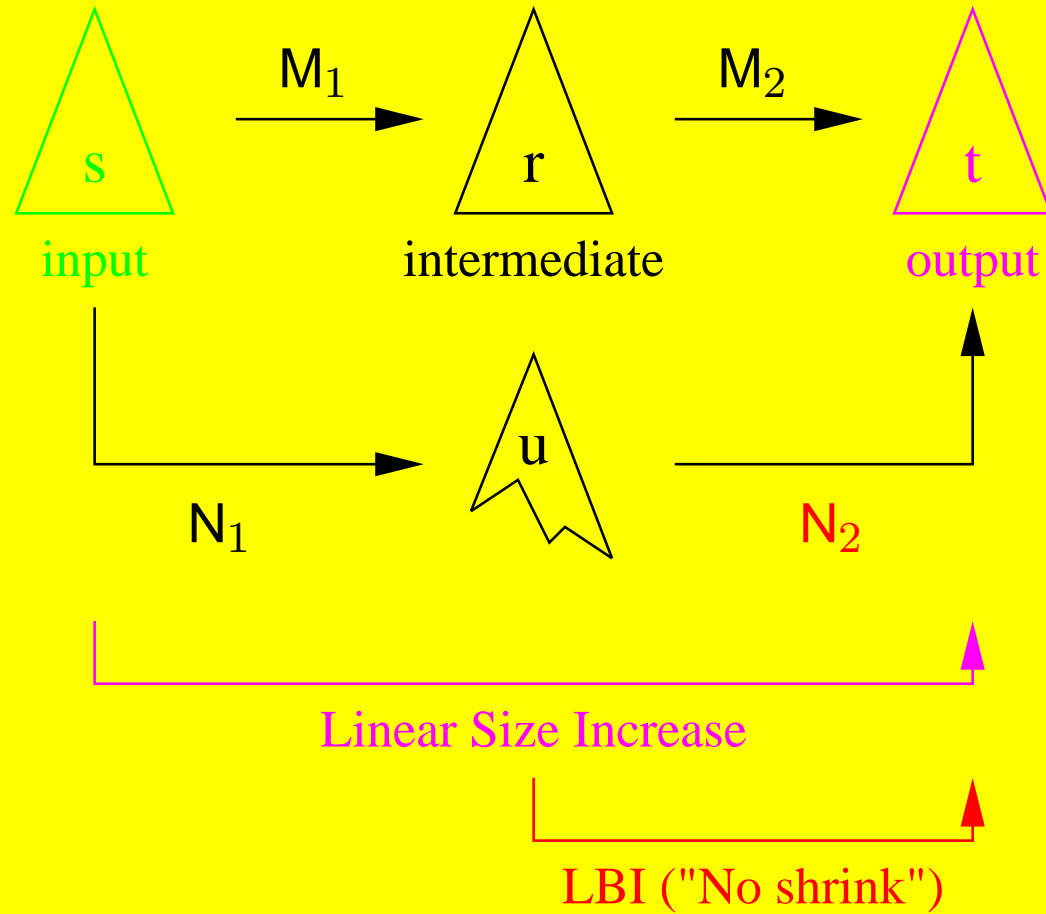
New Idea: Linear Bounded Input



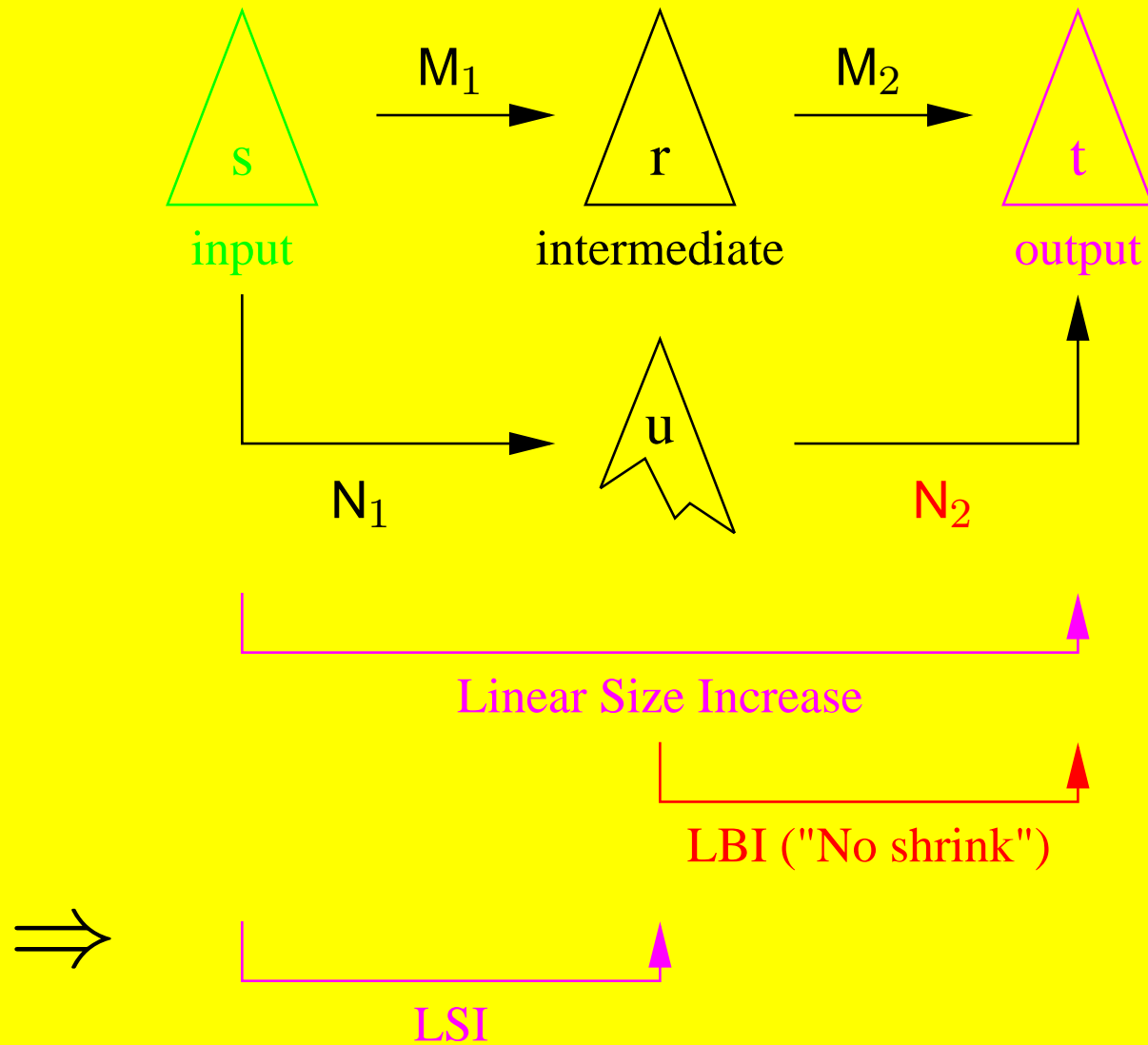
N_2 has Linear Bounded Input (LBI)

$$\exists c \forall u : |u| \leq c \cdot |N_2(u)|.$$

New Idea: Linear Bounded Input



New Idea: Linear Bounded Input



Main Result

Theorem. $MTT^* \cap LSI = MTT \cap LSI = C.$

Proof:

$$\begin{aligned} MTT^{n+1} \cap LSI &\subseteq MTT \circ (LMTT^n \cap LBI) \cap LSI && \text{[M02, Kanpur]} \\ &\subseteq (MTT \cap LSI) \circ LMTT^n \cap LSI && \text{(prev. slide)} \\ &\subseteq C \circ ATT^R \circ LMTT^{n-1} \cap LSI && \text{(prev. result)} \\ &\subseteq MTT^n \cap LSI && \text{(composition)} \\ &\subseteq MTT \cap LSI = C && \text{(induction)} \end{aligned}$$

XML Queries/Translations

XML documents correspond to **unranked** trees!

⇒ Macro **Forest** Transducers (**MFTs**) [PerstSeidl04]

Corrollary:

For a composition of **MFTs** or of PTTs, it is decidable whether or not it is **LSI**.

If it is, then an equivalent MTT (working on binary encodings) can be construced effectively.

Conclusion / Future Work

For tree / XML translations we do NOT
need **attribute grammars** anymore!

Think Functional!

Conclusion / Future Work

For tree / XML translations we do NOT need **attribute grammars** anymore!

Think Functional!

WHY?

- Linear Size Computability of **LMTT**: Can easily be proved via lazy evaluation.
- Composition: $MTT_{sur}^R = C \circ LMTT \subseteq MTT$.
Must hold. Direct construction **to be done..**

Conclusion / Future Work

For tree / XML translations we do NOT
need **attribute grammars** anymore!

Think Functional!

THE END