Learning XML : VPAs and Discrimination Trees

Cinzia Di Giusto, Davide Fissore, Etienne Lozes

AFFILIATION

Why VPA ?

For ∀ Non-Deterministic VPA V_1, there ∃ a Deterministic VPA V_2 such that $L(V_1) = L(V_2)$
→ Every binary operation between 2 VPA is decidable !

Note :
Push symbols ⇔ Open tags
Pop symbols ⇔ Close tags

VPAs

VPA := Visibly pushdown automata.
They can recognize context free languagages.
The alphabet is :

$$\hat{\Sigma} = \Sigma_{call} \uplus \Sigma_{ret} \uplus \Sigma_{int}$$

XML

XML (eXtensible Markup Language) is a standard format for data exchange.
XML representable w/**VPA**!

Acceptance for XML :
Empty stack + final states

And Communication ?

Arthur : Does w ∈ U ?
Merlin : Yes/No

Arthur creates a conjecture C.
Arthur : Does C = U ?
Merlin : if C = U → Yes
else → a counter-example

What is Learning ?

Dana Angluin's framework :

The Learner wants to learn a language U
The Teacher knows U

«Canonical» VPA

Regular automataon have a unique minimal representant, this is not true for VPA

k-SEVPA

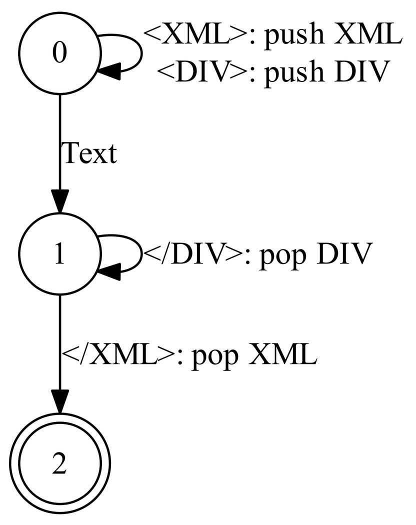
Single entry VPA are VPAs where states are partitioned into **k** modules.
Each module has only one entry for call transitions

An XML grammar to LEARN

G :=
d(XML) = Text + DIV
d(DIV) = Text + DIV

d : X → <X> RULE </X>

Example:
<XML><DIV>Text</DIV></XML> ∈ **G**



The learning phase

In Visibly Pushdown Language (VPL), we can adapt the Myhill-Nerode congruence :
two words $(\omega_1, \omega_2) \in \hat{\Sigma}^2$ are equivalent if

$\forall (u_1, u_2) \in \mathrm{WM}(\hat{\Sigma})$
$u_1 \cdot \omega_1 \cdot u_2 \in L \leftrightarrow u_1 \cdot \omega_2 \cdot u_2 \in L$

$\mathrm{WM}(\hat{\Sigma})$ It is a couple of words u_1, u_2 such that in $u = u_1 \circ u_2$ each call symbol has a corresponding ret symbol

Discrimination Tree

From WM, we can build a particular binary tree called Discrimination tree.
Inner Nodes contain a couple (u_1, u_2) and leaves are labelled with a string.

Leaves meaning

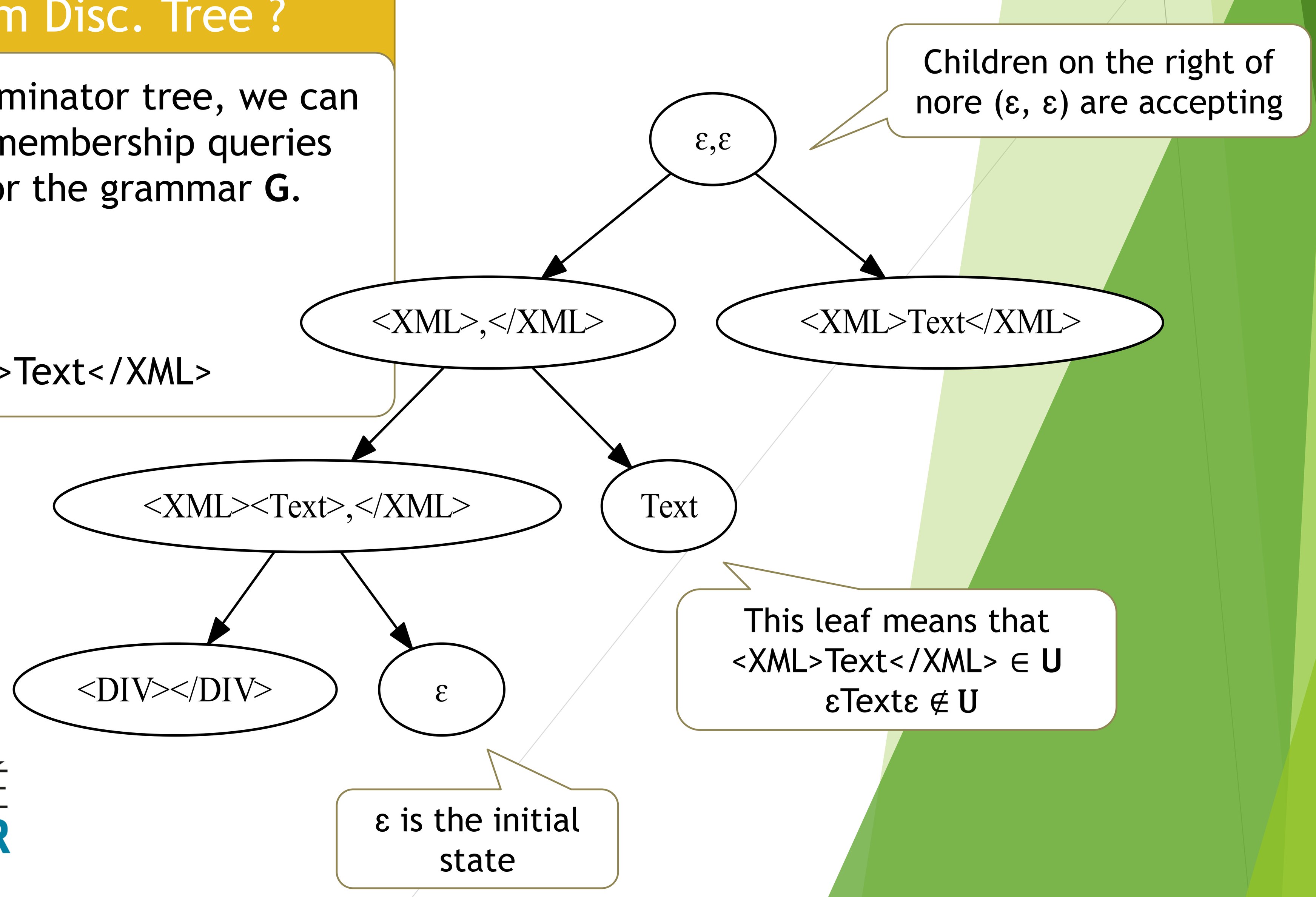
Leaves represent the VPA states and throught Membership questions, we build the corresponding VPA

LCA

The LCA L (Lowest Common Anchestor) of two leaves l1, l2 is the unique inner node such that l1 is on the right of L ↔ l2 is on the left of L

VPA from Disc. Tree ?

From this discriminator tree, we can build, through membership queries the same VPA for the grammar **G**.
Where:
state 0 := ε
state 1 := Text
state 2 := <XML>Text</XML>



References

Demo

DEMO_

DIGITAL SYSTEMS FOR HUMANS
GRADUATE SCHOOL AND RESEARCH

UNIVERSITÉ CÔTE D'AZUR

i3S sophia antipolis