Workshop in honor of Javier Esparza
ETAPS 2024

A Uniform Framework for Language Inclusion Problems

Kyveli Doveri
Pierre Ganty
Chana Weil-Kennedy

institute imdea software

Language Inclusion Problem

$$L \subseteq^? M$$

for L and M formal languages

Language Inclusion Problem

$$L \subseteq^{?} M$$

for L and M formal languages

Classical approach: complementation

$$L_1 \subseteq L_2 \iff L_1 \cap \overline{L_2} = \varnothing$$

Language Inclusion Problem

$$L \subseteq^? M$$

for L and M formal languages

Classical approach: complementation

$$L_1 \subseteq L_2 \iff L_1 \cap \overline{L_2} = \emptyset$$

We present an algorithm framework avoiding explicit complementation

Algorithm Framework

$$L \subseteq^? M$$

for L and M formal languages

Algorithm Framework

$$L \subseteq^? M$$

for L and M formal languages

The idea is to select a **subset** $S \subseteq L$ such that

1. S is a finite set $\{s_1, \ldots, s_n\}$
2. S is effectively computable
3. S contains a counter-example if one exists

Algorithm Framework

$$L \subseteq^? M$$

for L and M formal languages

The idea is to select a **subset** $S \subseteq L$ such that

1. S is a finite set $\{s_1, \ldots, s_n\}$
2. S is effectively computable
3. S contains a counter-example if one exists

$$L \subseteq M \iff S \subseteq M$$

Algorithm Framework

$$L \subseteq^? M$$

for L and M formal languages

We define S via

a **fixpoint characterization** of L

+ a **quasi-order** $\preccurlyeq$ on words defined using M

Algorithm Framework

$$L \subseteq^? M$$

for L and M formal languages

We define S via

a **fixpoint characterization** of L

+ a **quasi-order** $\preccurlyeq$ on words defined using M

We choose $S \subseteq L$ a **basis** of L for $\preccurlyeq$, i.e.

$\forall w \in L, \exists s \in S$ such that $s \preccurlyeq w$

Algorithm Framework

$$L \subseteq^{?} M$$

for L and M formal languages

We choose $S \subseteq L$ a **basis** of L for $\preccurlyeq$

1. S is a finite set $\{s_1, \ldots, s_n\}$
2. S is effectively computable
3. S contains a counter-example if one exists

Algorithm Framework

$$L \subseteq^{?} M$$

for L and M formal languages

We choose $S \subseteq L$ a **basis** of L for $\preccurlyeq$

1. $\preccurlyeq$ is a wqo

$\Longrightarrow$

1. S is a finite set $\{s_1, \ldots, s_n\}$
2. S is effectively computable
3. S contains a counter-example if one exists

Algorithm Framework

$$L \subseteq^? M$$

for L and M formal languages

We choose $S \subseteq L$ a **basis** of L for $\preccurlyeq$

1. $\preccurlyeq$ is a wqo $\implies$ 1. S is a finite set $\{s_1, \ldots, s_n\}$
2. $\preccurlyeq$ is monotonic $\implies$ 2. S is effectively computable

3. S contains a counter-example if one exists

Algorithm Framework

$$L \subseteq^? M$$

for L and M formal languages

We choose $S \subseteq L$ a **basis** of L for $\preccurlyeq$

1. $\preccurlyeq$ is a wqo	$\Longrightarrow$	1. S is a finite set $\{s_1, \ldots, s_n\}$
2. $\preccurlyeq$ is monotonic	$\Longrightarrow$	2. S is effectively computable
3. M is $\preccurlyeq$-upward closed	$\Longrightarrow$	3. S contains a counter-example if one exists

Algorithm Framework

$$L \subseteq^? M$$

for $L = L(\mathscr{G})$ for $\mathscr{G}$ a context-free grammar and M a regular language

Algorithm Framework

$$\mathcal{G}: \quad X \to aXb$$

$$X \to \varepsilon$$

$$L(\mathcal{G}) = \{a^n b^n \mid n \geq 0\}$$

Algorithm Framework

$$\mathscr{G}: \quad X \to aXb$$

$$X \to \varepsilon$$

$$F_{\mathscr{G}} : W \mapsto a\,W\,b \cup \{\varepsilon\}$$

$$L(\mathscr{G}) = \{a^n b^n \mid n \geq 0\}$$

Algorithm Framework

$\mathcal{G}: \quad X \to aXb$

$X \to \varepsilon$

$L(\mathcal{G}) = \{a^n b^n \mid n \geq 0\}$

$F_{\mathcal{G}} : W \mapsto a\,W\,b \cup \{\varepsilon\}$

$F_{\mathcal{G}}(\emptyset) = \{\varepsilon\}$

$F_{\mathcal{G}}^2(\emptyset) = \{\varepsilon, ab\}$

$F_{\mathcal{G}}^3(\emptyset) = \{\varepsilon, ab, aabb\}$

...

lfp $F_{\mathcal{G}} = L(\mathcal{G})$

Algorithm Framework

Algorithm: // decide $L(\mathscr{G}) \subseteq M$
Compute $F_{\mathscr{G}}^{m}(\varnothing)$ until
$F_{\mathscr{G}}^{m-1}(\varnothing)$ is a basis of $F_{\mathscr{G}}^{m}(\varnothing)$ for $\preccurlyeq$.
Check $F_{\mathscr{G}}^{m-1}(\varnothing) \subseteq M$.

$$F_{\mathscr{G}} : W \mapsto a\,W\,b \cup \{\varepsilon\}$$

$$F_{\mathscr{G}}(\varnothing) = \{\varepsilon\}$$

$$F_{\mathscr{G}}^{2}(\varnothing) = \{\varepsilon, ab\}$$

$$F_{\mathscr{G}}^{3}(\varnothing) = \{\varepsilon, ab, aabb\}$$

...

lfp $F_{\mathscr{G}} = L(\mathscr{G})$

Algorithm Framework

Algorithm: // decide $L(\mathscr{G}) \subseteq M$
Compute $F^m_{\mathscr{G}}(\emptyset)$ until
$F^{m-1}_{\mathscr{G}}(\emptyset)$ is a basis of $F^m_{\mathscr{G}}(\emptyset)$ for $\preccurlyeq$.
Check $F^{m-1}_{\mathscr{G}}(\emptyset) \subseteq M$.

There is such an m because $\preccurlyeq$ is a wqo

$$F_{\mathscr{G}} : W \mapsto a\,W\,b \cup \{\varepsilon\}$$

$$F_{\mathscr{G}}(\emptyset) = \{\varepsilon\}$$

$$F^2_{\mathscr{G}}(\emptyset) = \{\varepsilon, ab\}$$

$$F^3_{\mathscr{G}}(\emptyset) = \{\varepsilon, ab, aabb\}$$

...

lfp $F_{\mathscr{G}} = L(\mathscr{G})$

Algorithm Framework

Algorithm: // decide $L(\mathcal{G}) \subseteq M$
Compute $F_{\mathcal{G}}^{m}(\emptyset)$ until
$F_{\mathcal{G}}^{m-1}(\emptyset)$ is a basis of $F_{\mathcal{G}}^{m}(\emptyset)$ for $\preccurlyeq$.
Check $F_{\mathcal{G}}^{m-1}(\emptyset) \subseteq M$.

There is such an m because $\preccurlyeq$ is a wqo

$\preccurlyeq$ is **monotonic** if

$$u \preccurlyeq v \implies \forall w, w' \in \Sigma^* \; wuw' \preccurlyeq wvw'$$

$$F_{\mathcal{G}} : W \mapsto a\,W\,b \cup \{\varepsilon\}$$

$$F_{\mathcal{G}}(\emptyset) = \{\varepsilon\}$$

$$F_{\mathcal{G}}^{2}(\emptyset) = \{\varepsilon, ab\}$$

$$F_{\mathcal{G}}^{3}(\emptyset) = \{\varepsilon, ab, aabb\}$$

...

lfp $F_{\mathcal{G}} = L(\mathcal{G})$

Algorithm Framework

Algorithm: // decide $L(\mathcal{G}) \subseteq M$
Compute $F_{\mathcal{G}}^{m}(\emptyset)$ until
$F_{\mathcal{G}}^{m-1}(\emptyset)$ is a basis of $F_{\mathcal{G}}^{m}(\emptyset)$ for $\preccurlyeq$.
Check $F_{\mathcal{G}}^{m-1}(\emptyset) \subseteq M$.

$$F_{\mathcal{G}} : W \mapsto a\,W\,b \cup \{\varepsilon\}$$

$$F_{\mathcal{G}}(\emptyset) = \{\varepsilon\}$$

There is such an m because $\preccurlyeq$ is a wqo

$$F_{\mathcal{G}}^{2}(\emptyset) = \{\varepsilon, ab\}$$

$$F_{\mathcal{G}}^{3}(\emptyset) = \{\varepsilon, ab, aabb\}$$

...

$\preccurlyeq$ is **monotonic** if

$$u \preccurlyeq v \implies \forall w, w' \in \Sigma^* \; wuw' \preccurlyeq wvw'$$

lfp $F_{\mathcal{G}} = L(\mathcal{G})$

By monotonicity of $\preccurlyeq$, $F_{\mathcal{G}}^{m-1}(\emptyset)$ is a basis of lfp $F_{\mathcal{G}}$ for $\preccurlyeq$

Algorithm Framework

Algorithm: // decide $L(\mathscr{G}) \subseteq M$
Compute $F_{\mathscr{G}}^{m}(\varnothing)$ until
$F_{\mathscr{G}}^{m-1}(\varnothing)$ is a basis of $F_{\mathscr{G}}^{m}(\varnothing)$ for $\preccurlyeq$.
Check $F_{\mathscr{G}}^{m-1}(\varnothing) \subseteq M$.

$$F_{\mathscr{G}} : W \mapsto a\,W\,b \cup \{\varepsilon\}$$

$$F_{\mathscr{G}}(\varnothing) = \{\varepsilon\}$$

$$F_{\mathscr{G}}^{2}(\varnothing) = \{\varepsilon, ab\}$$

$$F_{\mathscr{G}}^{3}(\varnothing) = \{\varepsilon, ab, aabb\}$$

...

lfp $F_{\mathscr{G}} = L(\mathscr{G})$

There is such an m because $\preccurlyeq$ is a wqo

$\preccurlyeq$ is **monotonic** if

$$u \preccurlyeq v \implies \forall w, w' \in \Sigma^* \; wuw' \preccurlyeq wvw'$$

By monotonicity of $\preccurlyeq$, $F_{\mathscr{G}}^{m-1}(\varnothing)$ is a basis of lfp $F_{\mathscr{G}}$ for $\preccurlyeq$

Algorithm Instantiated

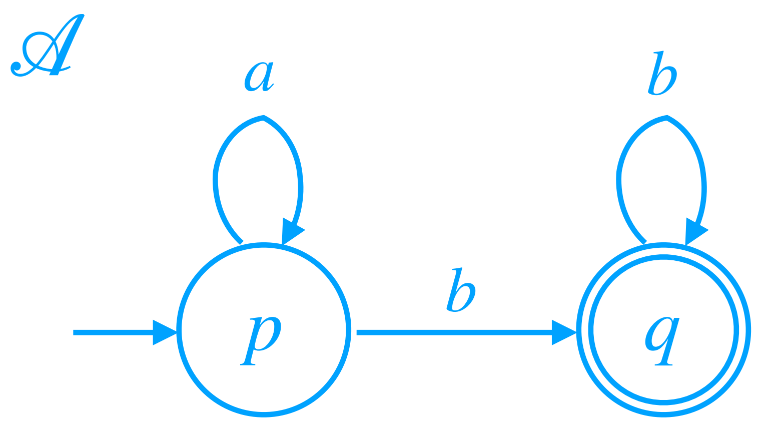
Let $u \in \Sigma^*$, and $\mathscr{A}$ an NFA

$$context_{\mathscr{A}}(u) = \{\ (q_1, q_2) \mid q_1 \xrightarrow{u}_{\mathscr{A}} q_2\ \}$$

Algorithm Instantiated

Let $u \in \Sigma^*$, and $\mathscr{A}$ an NFA

$$context_{\mathscr{A}}(u) = \{\ (q_1, q_2) \mid q_1 \xrightarrow{u}_{\mathscr{A}} q_2\ \}$$



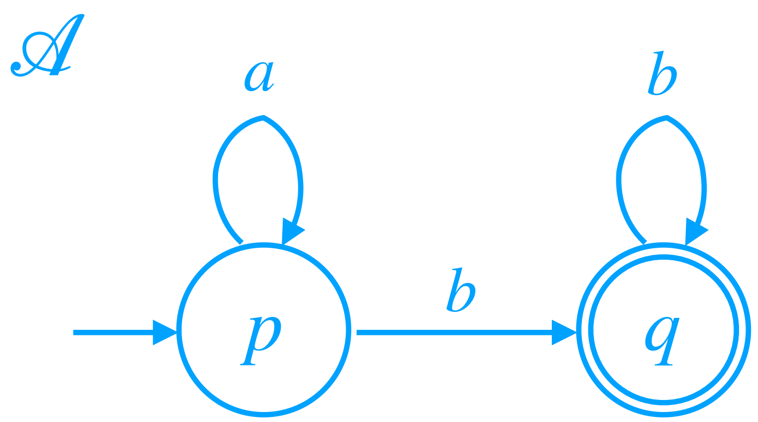
$L(\mathscr{A}) = a^*b^+$

Algorithm Instantiated

Let $u \in \Sigma^*$, and $\mathscr{A}$ an NFA

$$context_{\mathscr{A}}(u) = \{\ (q_1, q_2) \mid q_1 \xrightarrow{u}_{\mathscr{A}} q_2\ \}$$

$$context_{\mathscr{A}}(\varepsilon) = \{\ (p, p),\ (q, q)\ \}$$

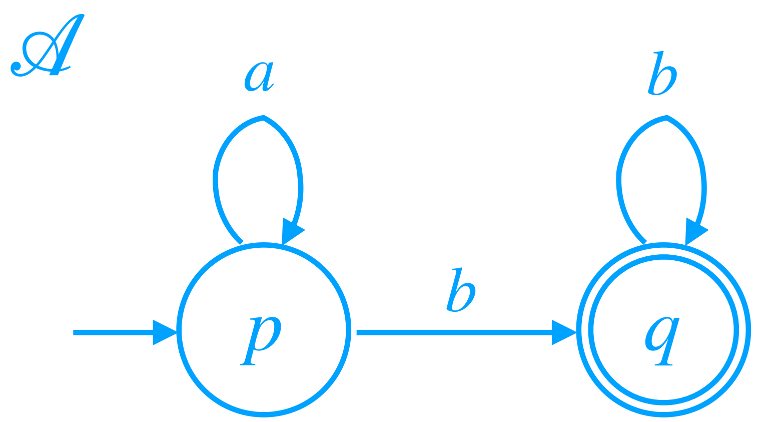


$L(\mathscr{A}) = a^*b^+$

Algorithm Instantiated

Let $u \in \Sigma^*$, and $\mathscr{A}$ an NFA

$$context_{\mathscr{A}}(u) = \{\ (q_1, q_2) \mid q_1 \xrightarrow{u}_{\mathscr{A}} q_2\ \}$$



$$context_{\mathscr{A}}(\varepsilon) = \{\ (p,p),\ (q,q)\ \}$$

$context_{\mathscr{A}}(w) = \{\ (p,q)\ \}$, for all $w \in a^+b^+$

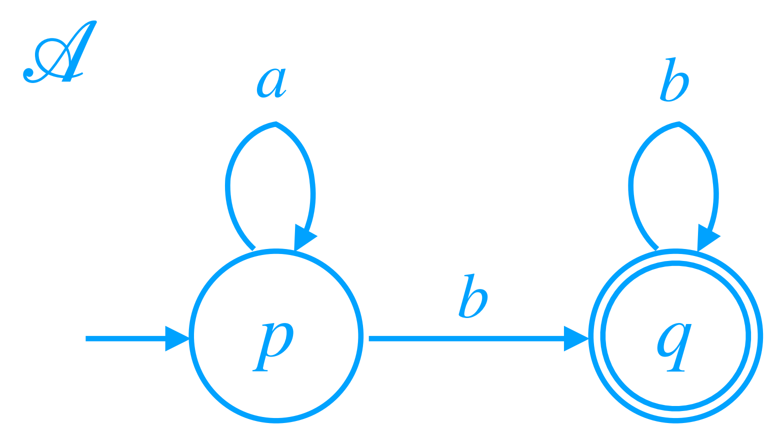
$L(\mathscr{A}) = a^*b^+$

Algorithm Instantiated

Let $u \in \Sigma^*$, and $\mathscr{A}$ an NFA

$$context_{\mathscr{A}}(u) = \{\ (q_1, q_2) \mid q_1 \xrightarrow{u}_{\mathscr{A}} q_2\ \}$$

$$u \preccurlyeq v \iff context_{\mathscr{A}}(u) \subseteq context_{\mathscr{A}}(v)$$



$$context_{\mathscr{A}}(\varepsilon) = \{\ (p,p),\ (q,q)\ \}$$

$context_{\mathscr{A}}(w) = \{\ (p,q)\ \}$, for all $w \in a^+b^+$

$L(\mathscr{A}) = a^*b^+$

Algorithm Instantiated

Let $u \in \Sigma^*$, and $\mathscr{A}$ an NFA

$$context_{\mathscr{A}}(u) = \{\ (q_1, q_2) \mid q_1 \xrightarrow{u}_{\mathscr{A}} q_2\ \}$$

$$u \preccurlyeq v \iff context_{\mathscr{A}}(u) \subseteq context_{\mathscr{A}}(v)$$

1. $\preccurlyeq$ is a wqo
2. $\preccurlyeq$ is monotonic
3. $L(\mathscr{A})$ is $\preccurlyeq$-upward closed

Algorithm Instantiated

$$\{a^n b^n \mid n \geq 0\} \subseteq^? a^* b^+$$

Algorithm Instantiated

$$\{a^n b^n \mid n \geq 0\} \subseteq^? a^* b^+$$

$$F_{\mathscr{G}} : W \mapsto a\,W\,b \cup \{\varepsilon\}$$

$$F_{\mathscr{G}}(\varnothing) = \{\varepsilon\}$$

basis for $\preccurlyeq$?

$$F^2_{\mathscr{G}}(\varnothing) = \{\varepsilon, ab\}$$

basis for $\preccurlyeq$?

$$F^3_{\mathscr{G}}(\varnothing) = \{\varepsilon, ab, aabb\}$$

...

lfp $F_{\mathscr{G}} = L(\mathscr{G})$

Algorithm Instantiated

$$\{a^n b^n \mid n \geq 0\} \subseteq^? a^* b^+$$

$$F_{\mathscr{G}} : W \mapsto a\,W\,b \cup \{\varepsilon\}$$

basis for $\preccurlyeq$?

$$F_{\mathscr{G}}(\varnothing) = \{\varepsilon\}$$

$$F^2_{\mathscr{G}}(\varnothing) = \{\varepsilon, ab\}$$

$$F^3_{\mathscr{G}}(\varnothing) = \{\varepsilon, ab, aabb\}$$

...

lfp $F_{\mathscr{G}} = L(\mathscr{G})$

Algorithm Instantiated

$$\{a^n b^n \mid n \geq 0\} \subseteq^? a^* b^+$$

$context_{\mathscr{A}}(\varepsilon) = \{\ (p, p),\ (q, q)\ \}$

$context_{\mathscr{A}}(w) = \{\ (p, q)\ \}$, for all $w \in a^+ b^+$

$F_{\mathscr{G}} : W \mapsto a\,W\,b \cup \{\varepsilon\}$

$F_{\mathscr{G}}(\varnothing) = \{\varepsilon\}$

basis for $\preccurlyeq$?

$F^2_{\mathscr{G}}(\varnothing) = \{\varepsilon, ab\}$

$F^3_{\mathscr{G}}(\varnothing) = \{\varepsilon, ab, aabb\}$

...

lfp $F_{\mathscr{G}} = L(\mathscr{G})$

Algorithm Instantiated

$$\{a^n b^n \mid n \geq 0\} \subseteq^? a^* b^+$$

$context_{\mathscr{A}}(\varepsilon) = \{\ (p, p),\ (q, q)\ \}$

$context_{\mathscr{A}}(w) = \{\ (p, q)\ \}$, for all $w \in a^+ b^+$

$F_{\mathscr{G}} : W \mapsto a\, W\, b \cup \{\varepsilon\}$

$\varepsilon \preccurlyeq \varepsilon$

$\varepsilon \not\preccurlyeq ab$

basis for $\preccurlyeq$?

$F_{\mathscr{G}}(\varnothing) = \{\varepsilon\}$

$F^2_{\mathscr{G}}(\varnothing) = \{\varepsilon, ab\}$

$F^3_{\mathscr{G}}(\varnothing) = \{\varepsilon, ab, aabb\}$

...

lfp $F_{\mathscr{G}} = L(\mathscr{G})$

Algorithm Instantiated

$$\{a^n b^n \mid n \geq 0\} \subseteq^? a^* b^+$$

$context_{\mathscr{A}}(\varepsilon) = \{\ (p,p),\ (q,q)\ \}$

$context_{\mathscr{A}}(w) = \{\ (p,q)\ \}$, for all $w \in a^+b^+$

$\varepsilon \preccurlyeq \varepsilon$

$\varepsilon \not\preccurlyeq ab$

$ab \preccurlyeq ab$

$ab \preccurlyeq aabb$

$F_{\mathscr{G}} : W \mapsto a\,W\,b \cup \{\varepsilon\}$

$F_{\mathscr{G}}(\varnothing) = \{\varepsilon\}$

$F^2_{\mathscr{G}}(\varnothing) = \{\varepsilon, ab\}$

basis for $\preccurlyeq$?

$F^3_{\mathscr{G}}(\varnothing) = \{\varepsilon, ab, aabb\}$

...

lfp $F_{\mathscr{G}} = L(\mathscr{G})$

Algorithm Instantiated

$$\{a^n b^n \mid n \geq 0\} \subseteq^? a^* b^+$$

$context_{\mathscr{A}}(\varepsilon) = \{\ (p,p),\ (q,q)\ \}$

$context_{\mathscr{A}}(w) = \{\ (p,q)\ \}$, for all $w \in a^+ b^+$

$F_{\mathscr{G}} : W \mapsto a\,W\,b \cup \{\varepsilon\}$

$\varepsilon \preccurlyeq \varepsilon$

$\varepsilon \not\preccurlyeq ab$

$ab \preccurlyeq ab$

$ab \preccurlyeq aabb$

$F_{\mathscr{G}}(\varnothing) = \{\varepsilon\}$

$F^2_{\mathscr{G}}(\varnothing) = \{\varepsilon, ab\}$

basis for $\preccurlyeq$ ✅

$F^3_{\mathscr{G}}(\varnothing) = \{\varepsilon, ab, aabb\}$

…

lfp $F_{\mathscr{G}} = L(\mathscr{G})$

Algorithm Instantiated

$$\{a^n b^n \mid n \geq 0\} \subseteq^? a^* b^+$$

$context_{\mathscr{A}}(\varepsilon) = \{\ (p,p), (q,q)\ \}$

$context_{\mathscr{A}}(w) = \{\ (p,q)\ \}$, for all $w \in a^+b^+$

$\varepsilon \preccurlyeq \varepsilon$

$\varepsilon \not\preccurlyeq ab$

$ab \preccurlyeq ab$

$ab \preccurlyeq aabb$

$F_{\mathscr{G}} : W \mapsto a\,W\,b \cup \{\varepsilon\}$

$F_{\mathscr{G}}(\varnothing) = \{\varepsilon\}$

$F^2_{\mathscr{G}}(\varnothing) = \{\varepsilon, ab\}$

basis for $\preccurlyeq$ ✅

$F^3_{\mathscr{G}}(\varnothing) = \{\varepsilon, ab, aabb\}$

...

lfp $F_{\mathscr{G}} = L(\mathscr{G})$

Algorithm Instantiated

$$\{a^n b^n \mid n \geq 0\} \subseteq^? a^* b^+$$

$context_{\mathscr{A}}(\varepsilon) = \{\ (p,p),\ (q,q)\ \}$

$context_{\mathscr{A}}(w) = \{\ (p,q)\ \}$, for all $w \in a^+ b^+$

$\varepsilon \preccurlyeq \varepsilon$

$\varepsilon \not\preccurlyeq ab$

$ab \preccurlyeq ab$

$ab \preccurlyeq aabb$

$F_{\mathscr{G}} : W \mapsto a\, W\, b \cup \{\varepsilon\}$

$F_{\mathscr{G}}(\varnothing) = \{\varepsilon\}$

$F^2_{\mathscr{G}}(\varnothing) = \{\varepsilon, ab\}$

basis for $\preccurlyeq$ ✅

$F^3_{\mathscr{G}}(\varnothing) = \{\varepsilon, ab, aabb\}$

...

lfp $F_{\mathscr{G}} = L(\mathscr{G})$

$$\{\varepsilon, ab\} \subseteq a^* b^+$$

Algorithm Instantiated

$$\{a^n b^n \mid n \geq 0\} \subseteq^? a^* b^+$$

$context_{\mathscr{A}}(\varepsilon) = \{\ (p,p),\ (q,q)\ \}$

$context_{\mathscr{A}}(w) = \{\ (p,q)\ \}$, for all $w \in a^+ b^+$

$\varepsilon \preccurlyeq \varepsilon$

$\varepsilon \not\preccurlyeq ab$

$ab \preccurlyeq ab$

$ab \preccurlyeq aabb$

$F_{\mathscr{G}} : W \mapsto a\,W\,b \cup \{\varepsilon\}$

$F_{\mathscr{G}}(\varnothing) = \{\varepsilon\}$

$F^2_{\mathscr{G}}(\varnothing) = \{\varepsilon, ab\}$

basis for $\preccurlyeq$ ✅

$F^3_{\mathscr{G}}(\varnothing) = \{\varepsilon, ab, aabb\}$

...

lfp $F_{\mathscr{G}} = L(\mathscr{G})$

$\{\varepsilon, ab\} \subseteq a^* b^+$ ❌

Algorithm Instantiated

$\{a^n b^n \mid n \geq 0\} \subseteq^? a^* b^+$ ❌

$context_{\mathscr{A}}(\varepsilon) = \{\ (p, p),\ (q, q)\ \}$

$context_{\mathscr{A}}(w) = \{\ (p, q)\ \}$, for all $w \in a^+ b^+$

$F_{\mathscr{G}} : W \mapsto a\,W\,b \cup \{\varepsilon\}$

$\varepsilon \preccurlyeq \varepsilon$

$\varepsilon \not\preccurlyeq ab$

$ab \preccurlyeq ab$

$ab \preccurlyeq aabb$

$F_{\mathscr{G}}(\varnothing) = \{\varepsilon\}$

$F^2_{\mathscr{G}}(\varnothing) = \{\varepsilon, ab\}$

basis for $\preccurlyeq$ ✅

$F^3_{\mathscr{G}}(\varnothing) = \{\varepsilon, ab, aabb\}$

...

lfp $F_{\mathscr{G}} = L(\mathscr{G})$

$\{\varepsilon, ab\} \subseteq a^* b^+$ ❌

Algorithm Instantiated

Let $u \in \Sigma^*$, and L a regular language

$$envt_L(u) = \{ \ (w_1, w_2) \mid w_1 u w_2 \in L \ \}$$

$$u \ll v \iff envt_L(u) \subseteq envt_L(v)$$

1. $\ll$ is a wqo
2. $\ll$ is monotonic
3. L is $\ll$-upward closed

Algorithm Instantiated

Let $u \in \Sigma^*$, and L a regular language

$$envt_L(u) = \{\ (w_1, w_2) \mid w_1 u w_2 \in L\ \}$$

$$u \ll v \iff envt_L(u) \subseteq envt_L(v)$$

1. $\ll$ is a wqo
2. $\ll$ is monotonic
3. L is $\ll$-upward closed

$\ll$ is the coarsest order such that this is true for the case CFL into REG

State-Based Algorithm

$$\{a^n b^n \mid n \geq 0\} \subseteq^? a^* b^+$$

$$F_{\mathscr{G}} : W \mapsto a\, W\, b \cup \{\varepsilon\}$$

$$F_{\mathscr{G}}(\varnothing) = \{\varepsilon\}$$

$$F^2_{\mathscr{G}}(\varnothing) = \{\varepsilon, ab\}$$

$$F^3_{\mathscr{G}}(\varnothing) = \{\varepsilon, ab, aabb\}$$

...

lfp $F_{\mathscr{G}} = L(\mathscr{G})$

State-Based Algorithm

$$\{a^n b^n \mid n \geq 0\} \subseteq^? a^* b^+$$

$$F_{\mathscr{G}} : W \mapsto a\,W\,b \cup \{\varepsilon\}$$

$\varepsilon \preccurlyeq \varepsilon$

$\varepsilon \not\preccurlyeq ab$

basis for $\preccurlyeq$
?

$$F_{\mathscr{G}}(\varnothing) = \{\varepsilon\}$$

$$F^2_{\mathscr{G}}(\varnothing) = \{\varepsilon, ab\}$$

$$F^3_{\mathscr{G}}(\varnothing) = \{\varepsilon, ab, aabb\}$$

...

lfp $F_{\mathscr{G}} = L(\mathscr{G})$

State-Based Algorithm

$$\{a^n b^n \mid n \geq 0\} \subseteq^{?} a^*b^+$$

$context_{\mathscr{A}}(\varepsilon) = \{\ (p,p),\ (q,q)\ \}$

$context_{\mathscr{A}}(w) = \{\ (p,q)\ \}$, for all $w \in a^+b^+$

$context_{\mathscr{A}}(\varepsilon) \subseteq context_{\mathscr{A}}(\varepsilon)$

$context_{\mathscr{A}}(ab) \nsubseteq context_{\mathscr{A}}(\varepsilon)$

$F_{\mathscr{G}} : W \mapsto a\,W\,b \cup \{\varepsilon\}$

basis for $\preccurlyeq$?

$F_{\mathscr{G}}(\varnothing) = \{\varepsilon\}$

$F^2_{\mathscr{G}}(\varnothing) = \{\varepsilon, ab\}$

$F^3_{\mathscr{G}}(\varnothing) = \{\varepsilon, ab, aabb\}$

...

lfp $F_{\mathscr{G}} = L(\mathscr{G})$

State-Based Algorithm

$$\{a^n b^n \mid n \geq 0\} \subseteq^? a^* b^+$$

$$F_{\mathscr{G}} : W \mapsto a\,W\,b \cup \{\varepsilon\}$$

$$\varepsilon \preccurlyeq \varepsilon$$

$$\varepsilon \not\preccurlyeq ab$$

$$ab \preccurlyeq ab$$

$$ab \preccurlyeq aabb$$

$$F_{\mathscr{G}}(\varnothing) = \{\varepsilon\}$$

$$F^2_{\mathscr{G}}(\varnothing) = \{\varepsilon, ab\}$$

basis for $\preccurlyeq$?

$$F^3_{\mathscr{G}}(\varnothing) = \{\varepsilon, ab, aabb\}$$

...

lfp $F_{\mathscr{G}} = L(\mathscr{G})$

State-Based Algorithm

$$\{a^n b^n \mid n \geq 0\} \subseteq^? a^* b^+$$

$context_{\mathscr{A}}(\varepsilon) = \{\ (p,p),\ (q,q)\ \}$

$context_{\mathscr{A}}(w) = \{\ (p,q)\ \}$, for all $w \in a^+b^+$

$context_{\mathscr{A}}(\varepsilon) \subseteq context_{\mathscr{A}}(\varepsilon)$

$context_{\mathscr{A}}(ab) \nsubseteq context_{\mathscr{A}}(\varepsilon)$

$context_{\mathscr{A}}(ab) \subseteq context_{\mathscr{A}}(ab)$

$context_{\mathscr{A}}(aabb) \subseteq context_{\mathscr{A}}(ab)$

$F_{\mathscr{G}} : W \mapsto a\,W\,b \cup \{\varepsilon\}$

$F_{\mathscr{G}}(\varnothing) = \{\varepsilon\}$

$F^2_{\mathscr{G}}(\varnothing) = \{\varepsilon, ab\}$

basis for $\preccurlyeq$?

$F^3_{\mathscr{G}}(\varnothing) = \{\varepsilon, ab, aabb\}$

...

lfp $F_{\mathscr{G}} = L(\mathscr{G})$

State-Based Algorithm

$$\{a^n b^n \mid n \geq 0\} \subseteq^? a^* b^+$$

→ store the contexts

$$F_{\mathscr{G}} : W \mapsto a\,W\,b \cup \{\varepsilon\}$$

$$F_{\mathscr{G}}(\varnothing) = \{\varepsilon\}$$

$$F_{\mathscr{G}}^2(\varnothing) = \{\varepsilon, ab\}$$

$$F_{\mathscr{G}}^3(\varnothing) = \{\varepsilon, ab, aabb\}$$

...

lfp $F_{\mathscr{G}} = L(\mathscr{G})$

State-Based Algorithm

$$\{a^n b^n \mid n \geq 0\} \subseteq^? a^* b^+$$

→ store the contexts

$$F_{\mathscr{G}} : W \mapsto a\,W\,b \cup \{\varepsilon\}$$

$$F_{\mathscr{G}}(\varnothing) = \{\varepsilon\}$$

$$F^2_{\mathscr{G}}(\varnothing) = \{\varepsilon, ab\}$$

$$F^3_{\mathscr{G}}(\varnothing) = \{\varepsilon, ab, aabb\}$$

...

lfp $F_{\mathscr{G}} = L(\mathscr{G})$

$$context_{\mathscr{A}}(ww') = \{(p,q) \mid$$
$$\exists p' .\ (p,p') \in context(w) \wedge (p',q) \in context(w')\}$$

State-Based Algorithm

$$\{a^n b^n \mid n \geq 0\} \subseteq^? a^* b^+$$

- store the contexts
- update contexts using contexts

$$context_{\mathscr{A}}(ww') = \{(p,q) \mid$$

$$\exists p' .\ (p,p') \in context(w) \land (p',q) \in context(w')\}$$

$$F_{\mathscr{G}} : W \mapsto a\,W\,b \cup \{\varepsilon\}$$

$$F_{\mathscr{G}}(\varnothing) = \{\varepsilon\}$$

$$F^2_{\mathscr{G}}(\varnothing) = \{\varepsilon, ab\}$$

$$F^3_{\mathscr{G}}(\varnothing) = \{\varepsilon, ab, aabb\}$$

...

lfp $F_{\mathscr{G}} = L(\mathscr{G})$

State-Based Algorithm

$$\{a^n b^n \mid n \geq 0\} \subseteq^? a^* b^+$$

- store the contexts
- update contexts using contexts

$$F_{\mathscr{G}} : W \mapsto a\,W\,b \cup \{\varepsilon\}$$

$$F_{\mathscr{G}}(\varnothing) = \{\varepsilon\}$$

$$F^2_{\mathscr{G}}(\varnothing) = \{\varepsilon, ab\}$$

$$F^3_{\mathscr{G}}(\varnothing) = \{\varepsilon, ab, aabb\}$$

...

lfp $F_{\mathscr{G}} = L(\mathscr{G})$

$$F^2_{\mathscr{G}}(\varnothing) = \{\varepsilon, ab\} \subseteq^? a^* b^+$$

State-Based Algorithm

$$\{a^n b^n \mid n \geq 0\} \subseteq^? a^* b^+$$

- store the contexts
- update contexts using contexts

$$F_{\mathscr{G}} : W \mapsto a\,W\,b \cup \{\varepsilon\}$$

$$F_{\mathscr{G}}(\emptyset) = \{\varepsilon\}$$

$$F^2_{\mathscr{G}}(\emptyset) = \{\varepsilon, ab\}$$

$$F^3_{\mathscr{G}}(\emptyset) = \{\varepsilon, ab, aabb\}$$

...

$$\text{lfp } F_{\mathscr{G}} = L(\mathscr{G})$$

$$F^2_{\mathscr{G}}(\emptyset) = \{\varepsilon, ab\} \subseteq^? a^* b^+$$

$$(q_i, q_f) \nsubseteq context_{\mathscr{A}}(\varepsilon)$$

$$(q_i, q_f) \subseteq context_{\mathscr{A}}(ab)$$

State-Based Algorithm

$$\{a^n b^n \mid n \geq 0\} \subseteq^? a^* b^+$$

- store the contexts
- update contexts using contexts
- do final membership tests using contexts

$$F_{\mathscr{G}} : W \mapsto a\,W\,b \cup \{\varepsilon\}$$

$$F_{\mathscr{G}}(\varnothing) = \{\varepsilon\}$$

$$F^2_{\mathscr{G}}(\varnothing) = \{\varepsilon, ab\}$$

$$F^3_{\mathscr{G}}(\varnothing) = \{\varepsilon, ab, aabb\}$$

...

lfp $F_{\mathscr{G}} = L(\mathscr{G})$

$$F^2_{\mathscr{G}}(\varnothing) = \{\varepsilon, ab\} \subseteq^? a^* b^+$$

$$(q_i, q_f) \nsubseteq context_{\mathscr{A}}(\varepsilon)$$

$$(q_i, q_f) \subseteq context_{\mathscr{A}}(ab)$$

State-Based Algorithm

$$\{a^n b^n \mid n \geq 0\} \subseteq^? a^* b^+$$

- store the contexts
- update contexts using contexts
- do final membership tests using contexts

Don't store words, only contexts

$$F_{\mathscr{G}} : W \mapsto a\,W\,b \cup \{\varepsilon\}$$

$$F_{\mathscr{G}}(\varnothing) = \{\varepsilon\}$$

$$F^2_{\mathscr{G}}(\varnothing) = \{\varepsilon, ab\}$$

$$F^3_{\mathscr{G}}(\varnothing) = \{\varepsilon, ab, aabb\}$$

...

lfp $F_{\mathscr{G}} = L(\mathscr{G})$

State-Based Algorithm

$$\{a^n b^n \mid n \geq 0\} \subseteq^? a^* b^+$$

- store the contexts
- update contexts using contexts
- do final membership tests using contexts

$$F_{\mathscr{G}} : W \mapsto a\,W\,b \cup \{\varepsilon\}$$

$$F_{\mathscr{G}}(\emptyset) = \{\varepsilon\}$$

$$F^2_{\mathscr{G}}(\emptyset) = \{\varepsilon, ab\}$$

$$F^3_{\mathscr{G}}(\emptyset) = \{\varepsilon, ab, aabb\}$$

...

lfp $F_{\mathscr{G}} = L(\mathscr{G})$

Don't store words, only contexts

We obtain an **antichain** algorithm

e.g. [De Wulf, Doyen, Henzinger, Raskin, CAV'06]

e.g. [Holík, Meyer, NETYS'15]

Saturation Algorithm

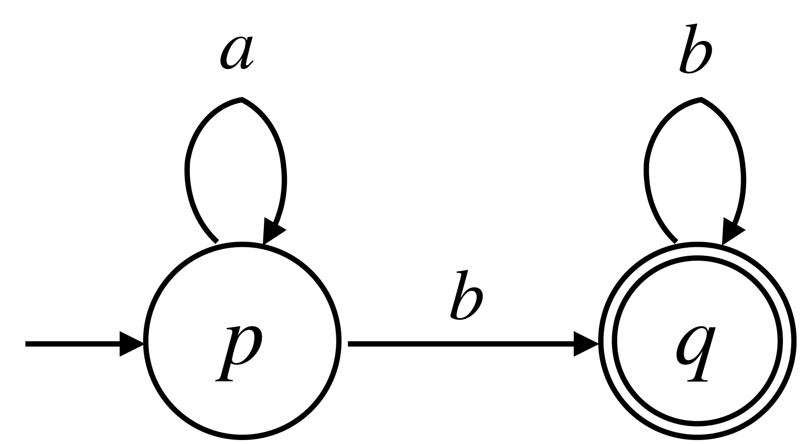
[Esparza, Rossmanith, Schwoon: *A Uniform Framework for Problems on Context-Free Grammars.* EATCS Bulletin, 2000] has the following algorithm

$\mathcal{G}$

$$X \to aXb$$

$$X \to \varepsilon$$

$\mathcal{A}$



Saturation Algorithm

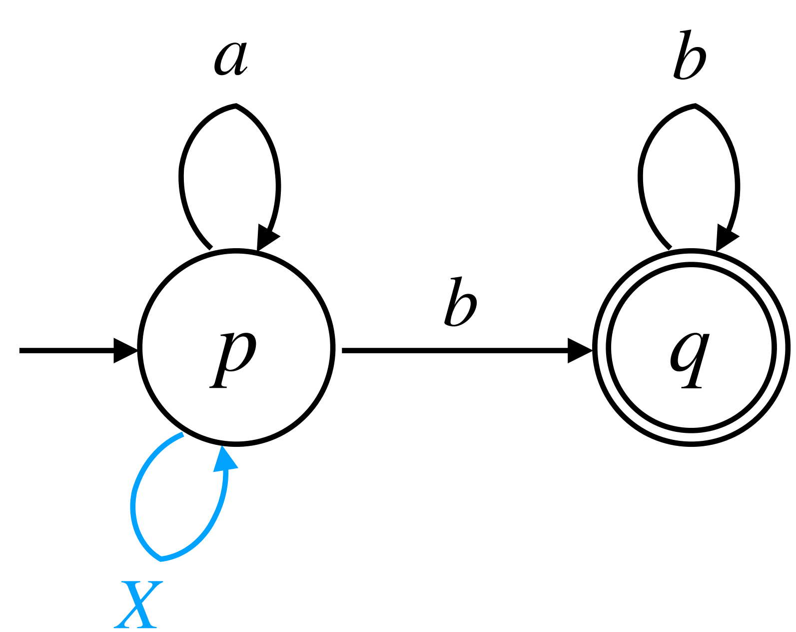
[Esparza, Rossmanith, Schwoon: *A Uniform Framework for Problems on Context-Free Grammars.* EATCS Bulletin, 2000] has the following algorithm

$\mathscr{G}$

$$X \to aXb$$

$$X \to \varepsilon$$

$\mathscr{A}$



Saturation Algorithm

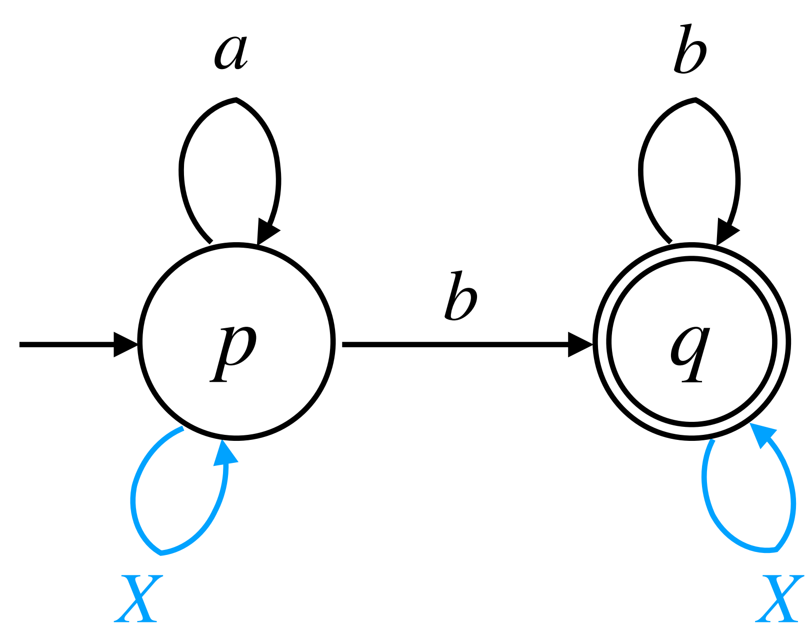
[Esparza, Rossmanith, Schwoon: *A Uniform Framework for Problems on Context-Free Grammars.* EATCS Bulletin, 2000] has the following algorithm

$\mathscr{G}$

$$X \to aXb$$

$$X \to \varepsilon$$

$\mathscr{A}$



Saturation Algorithm

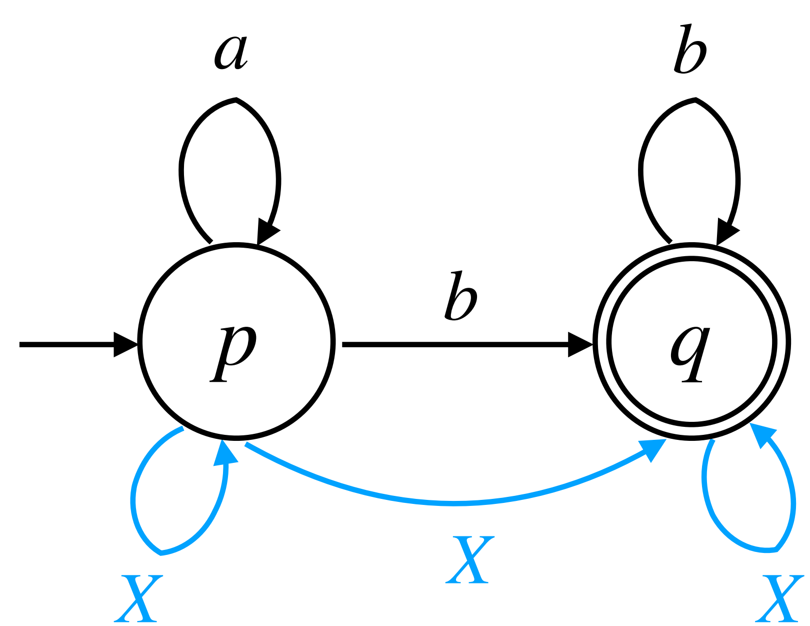
[Esparza, Rossmanith, Schwoon: *A Uniform Framework for Problems on Context-Free Grammars.* EATCS Bulletin, 2000] has the following algorithm

$\mathscr{G}$

$$X \to aXb$$

$$X \to \varepsilon$$

$\mathscr{A}$



Saturation Algorithm

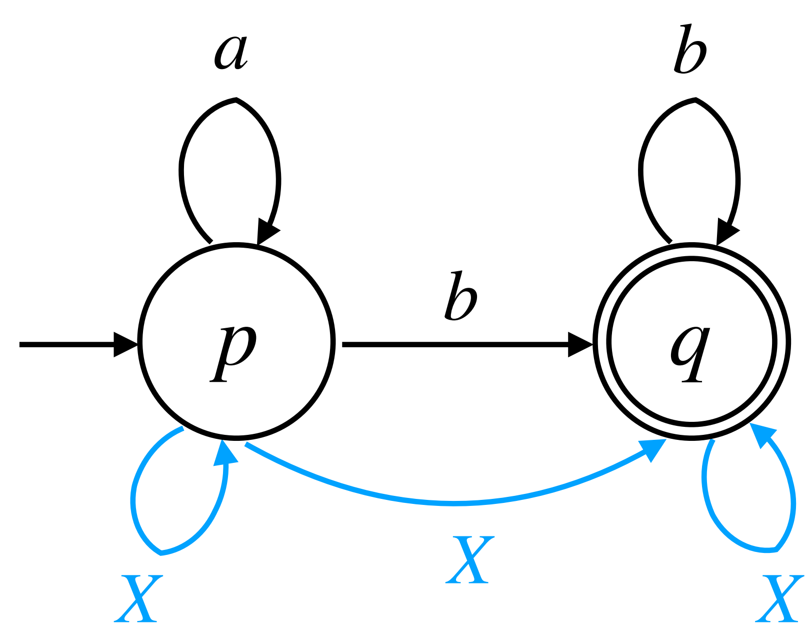
[Esparza, Rossmanith, Schwoon: *A Uniform Framework for Problems on Context-Free Grammars.* EATCS Bulletin, 2000] has the following algorithm

$\mathscr{G}$

$$X \to aXb$$

$$X \to \varepsilon$$

$\mathscr{A}$



Applications to emptiness, finiteness, …

SLP Case

An SLP is a grammar generating exactly one string

$$\mathcal{G}: \; X_1 \to X_0 X_0$$
$$X_0 \to a$$

SLP Case

An SLP is a grammar generating exactly one string

$$\mathcal{G} : X_1 \to X_0 X_0$$
$$X_0 \to a$$

$$F_{\mathcal{G}}\begin{pmatrix}\varnothing \\ \varnothing\end{pmatrix} = \begin{pmatrix}\varnothing \\ a\end{pmatrix} \qquad F^2_{\mathcal{G}}\begin{pmatrix}\varnothing \\ \varnothing\end{pmatrix} = \begin{pmatrix}aa \\ a\end{pmatrix}$$

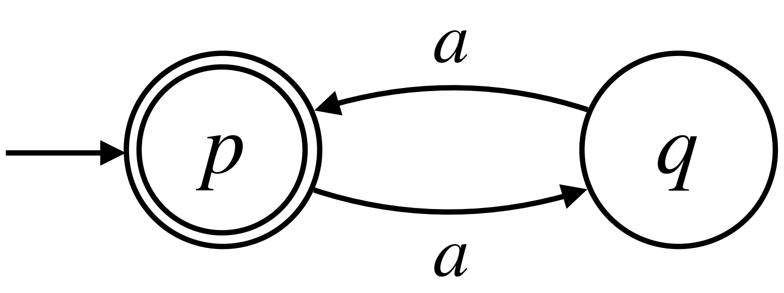
SLP Case

An SLP is a grammar generating exactly one string

$$\mathcal{G} : X_1 \to X_0 X_0$$
$$X_0 \to a$$

$$F_{\mathcal{G}}\begin{pmatrix}\varnothing \\ \varnothing\end{pmatrix} = \begin{pmatrix}\varnothing \\ a\end{pmatrix} \qquad F^2_{\mathcal{G}}\begin{pmatrix}\varnothing \\ \varnothing\end{pmatrix} = \begin{pmatrix}aa \\ a\end{pmatrix}$$

$$\{aa\} \subseteq^? (aa)^*$$



SLP Case

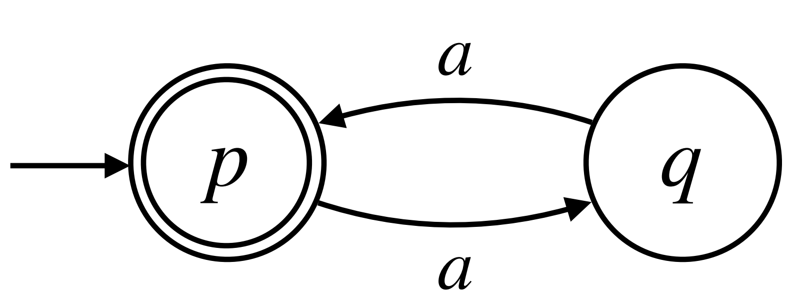
An SLP is a grammar generating exactly one string

$$\mathscr{G} : X_1 \to X_0 X_0$$
$$X_0 \to a$$

$$F_{\mathscr{G}} \begin{pmatrix} \varnothing \\ \varnothing \end{pmatrix} = \begin{pmatrix} \varnothing \\ a \end{pmatrix} \qquad F^2_{\mathscr{G}} \begin{pmatrix} \varnothing \\ \varnothing \end{pmatrix} = \begin{pmatrix} aa \\ a \end{pmatrix}$$

$$\{aa\} \subseteq^? (aa)^*$$

$$context_{\mathscr{A}}(a) = \{\ (p,q),\ (q,p)\ \}$$



SLP Case

An SLP is a grammar generating exactly one string

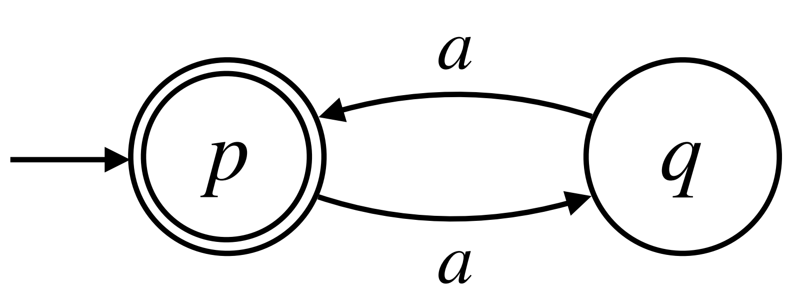
$$\mathscr{G}: X_1 \to X_0X_0$$
$$X_0 \to a$$

$$F_{\mathscr{G}}\begin{pmatrix}\varnothing\\ \varnothing\end{pmatrix} = \begin{pmatrix}\varnothing\\ a\end{pmatrix} \qquad F^2_{\mathscr{G}}\begin{pmatrix}\varnothing\\ \varnothing\end{pmatrix} = \begin{pmatrix}aa\\ a\end{pmatrix}$$

$$\{aa\} \subseteq^? (aa)^*$$

$$context_{\mathscr{A}}(a) = \{\ (p,q),\ (q,p)\ \}$$

$$context_{\mathscr{A}}(aa) = \{\ (p,p),\ (q,q)\ \}$$



SLP Case

An SLP is a grammar generating exactly one string

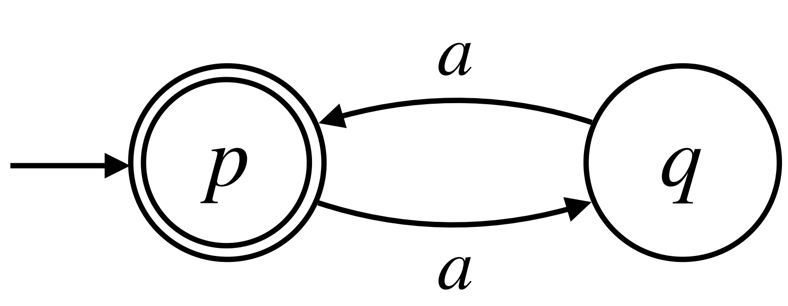
$$\mathcal{G}: X_1 \to X_0 X_0$$
$$X_0 \to a$$

$$F_{\mathcal{G}}\begin{pmatrix}\varnothing\\ \varnothing\end{pmatrix} = \begin{pmatrix}\varnothing\\ a\end{pmatrix} \qquad F^2_{\mathcal{G}}\begin{pmatrix}\varnothing\\ \varnothing\end{pmatrix} = \begin{pmatrix}aa\\ a\end{pmatrix}$$

$$\{aa\} \subseteq^? (aa)^*$$

$$context_{\mathcal{A}}(a) = \{\ (p,q),\ (q,p)\ \}$$

$$context_{\mathcal{A}}(aa) = \{\ (p,p),\ (q,q)\ \}$$



SLP Case

An SLP is a grammar generating exactly one string

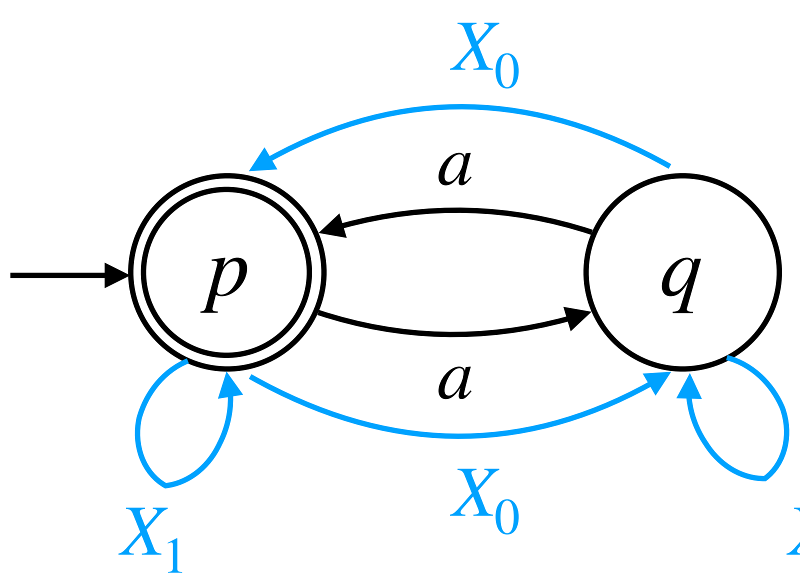
$$\mathcal{G}: X_1 \to X_0 X_0$$
$$X_0 \to a$$

$$F_{\mathcal{G}}\begin{pmatrix}\varnothing\\ \varnothing\end{pmatrix} = \begin{pmatrix}\varnothing\\ a\end{pmatrix} \qquad F^2_{\mathcal{G}}\begin{pmatrix}\varnothing\\ \varnothing\end{pmatrix} = \begin{pmatrix}aa\\ a\end{pmatrix}$$

$$\{aa\} \subseteq^? (aa)^*$$

$$context_{\mathscr{A}}(a) = \{\ (p,q),\ (q,p)\ \}$$

$$context_{\mathscr{A}}(aa) = \{\ (p,p),\ (q,q)\ \}$$



SLP Case

An SLP is a grammar generating exactly one string

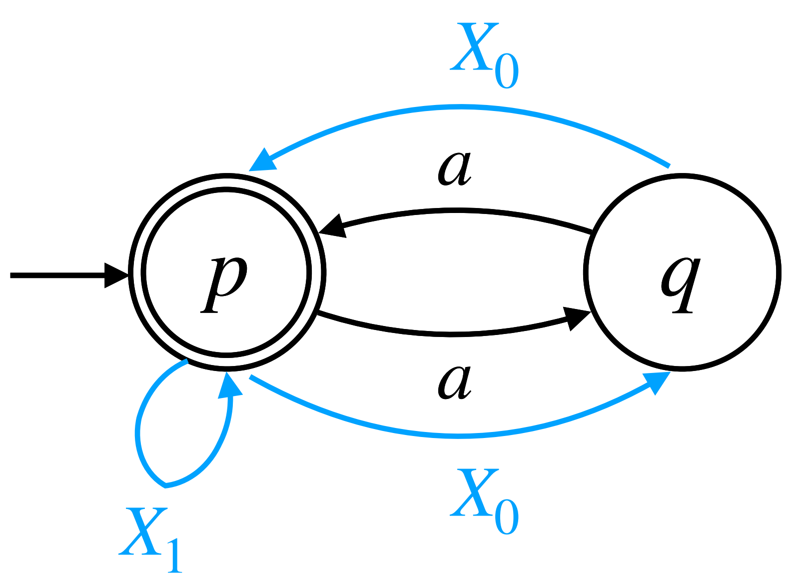
$$\mathcal{G}: X_1 \to X_0 X_0$$
$$X_0 \to a$$

$$F_{\mathcal{G}}\begin{pmatrix}\varnothing\\ \varnothing\end{pmatrix} = \begin{pmatrix}\varnothing\\ a\end{pmatrix} \qquad F^2_{\mathcal{G}}\begin{pmatrix}\varnothing\\ \varnothing\end{pmatrix} = \begin{pmatrix}aa\\ a\end{pmatrix}$$

$$\{aa\} \subseteq^? (aa)^*$$

$$context_{\mathscr{A}}(a) = \{\ (p,q),\ (q,p)\ \}$$

$$context_{\mathscr{A}}(aa) = \{\ (p,p),\ (q,q)\ \}$$



Final check: is there a X_1-arc from q_i to F?

SLP Case

The case SLP $\subseteq$ REG has applications to regular expression matching for text compressed using SLP.

SLP Case

The case SLP $\subseteq$ REG has applications to regular expression matching for text compressed using SLP.

$$\begin{aligned} \mathcal{G} :\ & X_1 \to X_0 X_0 \\ & X_0 \to a \end{aligned}$$

$$L(\mathcal{G}) = \{aa\} \subseteq^? (aa)^*$$

SLP Case

The case SLP ⊆ REG has applications to regular expression matching for text compressed using SLP.

$$\mathcal{G} : \; X_1 \to X_0 X_0$$
$$X_0 \to a$$

$$L(\mathcal{G}) = \{aa\} \subseteq^? (aa)^*$$

$$\mathcal{G}_n : X_n \to X_{n-1} X_{n-1}$$
$$X_{n-1} \to X_{n-2} X_{n-2}$$
$$\ldots$$
$$X_0 \to a$$

$$L(\mathcal{G}_n) = \{a^{2^n}\} \subseteq^? (aa)^*$$

Other Applications of the Framework

- REG $\subseteq$ REG
- REG $\subseteq$ Petri net traces
- ω-REG $\subseteq$ ω-REG, now part of SPOT!
- ω-CFL $\subseteq$ ω-REG
- ω-VPL $\subseteq$ ω-VPL
- operator-precedence languages [Henzinger et al., ICALP'23]

Other Applications of the Framework

- REG ⊆ REG
- REG ⊆ Petri net traces
- ω-REG ⊆ ω-REG, now part of SPOT!
- ω-CFL ⊆ ω-REG
- ω-VPL ⊆ ω-VPL
- operator-precedence languages [Henzinger et al., ICALP'23]

Thank you for listening!

Questions?