Context-Free Trees

Jan Philipp **Wächter**

Department of Mathematics
University of Manchester

This research was supported by EPSRC

29 July 2026

Preliminaries

Preliminaries

- involutive alphabet: alphabet A with involution $\cdot^{-1}\colon A \to A$ (i.e. $(a^{-1})^{-1} = a$ for all $a \in A$)

Preliminaries

- involutive alphabet: alphabet A with involution $\cdot^{-1} \colon A \to A$ (i.e. $(a^{-1})^{-1} = a$ for all $a \in A$)
  $A^{-1} = \{a^{-1} \mid a \in A\}$: disjoint copy

Preliminaries

- involutive alphabet: alphabet A with involution $\cdot^{-1}\colon A \to A$ (i.e. $(a^{-1})^{-1} = a$ for all $a \in A$)
  $A^{-1} = \{a^{-1} \mid a \in A\}$: disjoint copy $A^{\pm 1} = A \uplus A^{-1}$: involutive alphabet

Preliminaries

- involutive alphabet: alphabet A with involution $\cdot^{-1}\colon A \to A$ (i.e. $(a^{-1})^{-1} = a$ for all $a \in A$)
  $A^{-1} = \{a^{-1} \mid a \in A\}$: disjoint copy $\quad A^{\pm 1} = A \uplus A^{-1}$: involutive alphabet
- A-graph: $\Gamma = (V, E)$ with $E \subseteq V \times A \times V$ (i.e. directed, A-edge-labeled)

Preliminaries

- involutive alphabet: alphabet A with involution $\cdot^{-1}\colon A \to A$ (i.e. $(a^{-1})^{-1} = a$ for all $a \in A$)
  $A^{-1} = \{a^{-1} \mid a \in A\}$: disjoint copy $\quad A^{\pm 1} = A \uplus A^{-1}$: involutive alphabet
- A-graph: $\Gamma = (V, E)$ with $E \subseteq V \times A \times V$ (i.e. directed, A-edge-labeled)
  Today: usually with a root

Preliminaries

- involutive alphabet: alphabet A with involution $\cdot^{-1}\colon A \to A$ (i.e. $(a^{-1})^{-1} = a$ for all $a \in A$)
  $A^{-1} = \{a^{-1} \mid a \in A\}$: disjoint copy $\quad A^{\pm 1} = A \uplus A^{-1}$: involutive alphabet
- A-graph: $\Gamma = (V, E)$ with $E \subseteq V \times A \times V$ (i.e. directed, A-edge-labeled)
  Today: usually with a root
- involutive A-graph Γ:

Preliminaries

- involutive alphabet: alphabet A with involution $\cdot^{-1}\colon A \to A$ (i.e. $(a^{-1})^{-1} = a$ for all $a \in A$)
  $A^{-1} = \{a^{-1} \mid a \in A\}$: disjoint copy $A^{\pm 1} = A \uplus A^{-1}$: involutive alphabet
- A-graph: $\Gamma = (V, E)$ with $E \subseteq V \times A \times V$ (i.e. directed, A-edge-labeled)
  Today: usually with a root
- involutive A-graph Γ: A involutive alphabet and

Preliminaries

- involutive alphabet: alphabet A with involution $\cdot^{-1}\colon A \to A$ (i.e. $(a^{-1})^{-1} = a$ for all $a \in A$)
  $A^{-1} = \{a^{-1} \mid a \in A\}$: disjoint copy $A^{\pm 1} = A \uplus A^{-1}$: involutive alphabet
- A-graph: $\Gamma = (V, E)$ with $E \subseteq V \times A \times V$ (i.e. directed, A-edge-labeled)
  Today: usually with a root
- involutive A-graph Γ: A involutive alphabet and $u \xrightarrow{a} v$ in Γ $\iff$ $u \xleftarrow{a^{-1}} v$ in Γ

Preliminaries

- involutive alphabet: alphabet A with involution $\cdot^{-1}\colon A \to A$ (i.e. $(a^{-1})^{-1} = a$ for all $a \in A$)
  $A^{-1} = \{a^{-1} \mid a \in A\}$: disjoint copy $A^{\pm 1} = A \uplus A^{-1}$: involutive alphabet
- A-graph: $\Gamma = (V, E)$ with $E \subseteq V \times A \times V$ (i.e. directed, A-edge-labeled)
  Today: usually with a root
- involutive A-graph Γ: A involutive alphabet and $u \xrightarrow{a} v$ in Γ $\iff$ $u \xleftarrow{a^{-1}} v$ in Γ
- deterministic A-graph Γ:

Preliminaries

- involutive alphabet: alphabet A with involution $\cdot^{-1}\colon A \to A$ (i.e. $(a^{-1})^{-1} = a$ for all $a \in A$)
  $A^{-1} = \{a^{-1} \mid a \in A\}$: disjoint copy $A^{\pm 1} = A \uplus A^{-1}$: involutive alphabet
- A-graph: $\Gamma = (V, E)$ with $E \subseteq V \times A \times V$ (i.e. directed, A-edge-labeled)
  Today: usually with a root
- involutive A-graph Γ: A involutive alphabet and $u \xrightarrow{a} v$ in Γ $\iff$ $u \xleftarrow{a^{-1}} v$ in Γ
- deterministic A-graph Γ:

$$\begin{tikzcd} & v_1 \\ u \arrow[ur, "a"] \arrow[dr, "a"'] & \nparallel & \text{in } \Gamma \\ & v_2 \end{tikzcd}$$

Preliminaries

- involutive alphabet: alphabet A with involution $\cdot^{-1}\colon A \to A$ (i.e. $(a^{-1})^{-1} = a$ for all $a \in A$)
  $A^{-1} = \{a^{-1} \mid a \in A\}$: disjoint copy $\quad A^{\pm 1} = A \uplus A^{-1}$: involutive alphabet
- A-graph: $\Gamma = (V, E)$ with $E \subseteq V \times A \times V$ (i.e. directed, A-edge-labeled)
  Today: usually with a root
- involutive A-graph Γ: A involutive alphabet and $u \xrightarrow{a} v$ in $\Gamma \iff u \xleftarrow{a^{-1}} v$ in Γ
- deterministic A-graph Γ:

$$\begin{tikzcd} & v_1 \arrow[dd, phantom, "\neq"] \\ u \arrow[ur, "a"] \arrow[dr, "a"'] & \\ & v_2 \end{tikzcd} \quad \text{in } \Gamma$$

Preliminaries

- involutive alphabet: alphabet A with involution $\cdot^{-1}\colon A \to A$ (i.e. $(a^{-1})^{-1} = a$ for all $a \in A$)
  $A^{-1} = \{a^{-1} \mid a \in A\}$: disjoint copy $\quad A^{\pm 1} = A \uplus A^{-1}$: involutive alphabet
- A-graph: $\Gamma = (V, E)$ with $E \subseteq V \times A \times V$ (i.e. directed, A-edge-labeled)
  Today: usually with a root
- involutive A-graph Γ: A involutive alphabet and $u \xrightarrow{a} v$ in Γ $\iff$ $u \xleftarrow{a^{-1}} v$ in Γ
- deterministic A-graph Γ:

$$\begin{tikzcd} & v_1 \arrow[d, phantom, "\neq"] \\ u \arrow[ur, "a"] \arrow[dr, "a"'] & \\ & v_2 \end{tikzcd} \quad \text{in } \Gamma$$

- co-deterministic A-graph Γ:

Preliminaries

- involutive alphabet: alphabet A with involution $\cdot^{-1} \colon A \to A$ (i.e. $(a^{-1})^{-1} = a$ for all $a \in A$)
  $A^{-1} = \{a^{-1} \mid a \in A\}$: disjoint copy $\quad A^{\pm 1} = A \uplus A^{-1}$: involutive alphabet
- A-graph: $\Gamma = (V, E)$ with $E \subseteq V \times A \times V$ (i.e. directed, A-edge-labeled)
  Today: usually with a root
- involutive A-graph Γ: A involutive alphabet and $u \xrightarrow{a} v$ in $\Gamma \iff u \xleftarrow{a^{-1}} v$ in Γ
- deterministic A-graph Γ:

$$\begin{tikzcd} & v_1 \arrow[d, phantom, "\neq"] \\ u \arrow[ur, "a"] \arrow[dr, "a"'] & \\ & v_2 \end{tikzcd} \quad \text{in } \Gamma$$

- co-deterministic A-graph Γ:

$$\begin{tikzcd} & u_1 \arrow[dl, "a"'] \arrow[d, phantom, "\neq"] \\ v & \\ & u_2 \arrow[ul, "a"] \end{tikzcd} \quad \text{in } \Gamma$$

Preliminaries

- involutive alphabet: alphabet A with involution $\cdot^{-1} \colon A \to A$ (i.e. $(a^{-1})^{-1} = a$ for all $a \in A$)
  $A^{-1} = \{a^{-1} \mid a \in A\}$: disjoint copy $\quad A^{\pm 1} = A \uplus A^{-1}$: involutive alphabet
- A-graph: $\Gamma = (V, E)$ with $E \subseteq V \times A \times V$ (i.e. directed, A-edge-labeled)
  Today: usually with a root
- involutive A-graph Γ: A involutive alphabet and $u \xrightarrow{a} v$ in Γ $\iff$ $u \xleftarrow{a^{-1}} v$ in Γ
- deterministic A-graph Γ:

$$\begin{array}{ccc} & \overset{a}{\nearrow} & v_1 \\ u & & \nparallel \\ & \underset{a}{\searrow} & v_2 \end{array} \quad \text{in } \Gamma$$

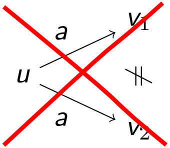
- co-deterministic A-graph Γ:

$$\begin{array}{ccc} & \overset{a}{\swarrow} & u_1 \\ v & & \nparallel \\ & \underset{a}{\nwarrow} & u_2 \end{array} \quad \text{in } \Gamma$$

- inverse A-graph: involutive + strongly connected + deterministic + co-deterministic

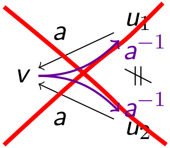
Preliminaries

- involutive alphabet: alphabet A with involution $\cdot^{-1}\colon A \to A$ (i.e. $(a^{-1})^{-1} = a$ for all $a \in A$)
  $A^{-1} = \{a^{-1} \mid a \in A\}$: disjoint copy $\quad A^{\pm 1} = A \uplus A^{-1}$: involutive alphabet
- A-graph: $\Gamma = (V, E)$ with $E \subseteq V \times A \times V$ (i.e. directed, A-edge-labeled)
  Today: usually with a root
- involutive A-graph Γ: A involutive alphabet and $u \xrightarrow{a} v$ in Γ $\iff$ $u \xleftarrow{a^{-1}} v$ in Γ
- deterministic A-graph Γ:



$\not\parallel$ in Γ

- co-deterministic A-graph Γ:

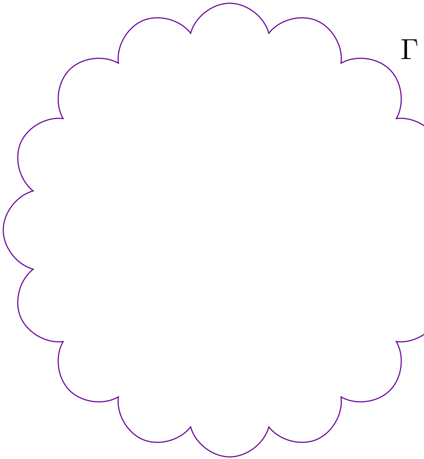


$\not\parallel$ in Γ

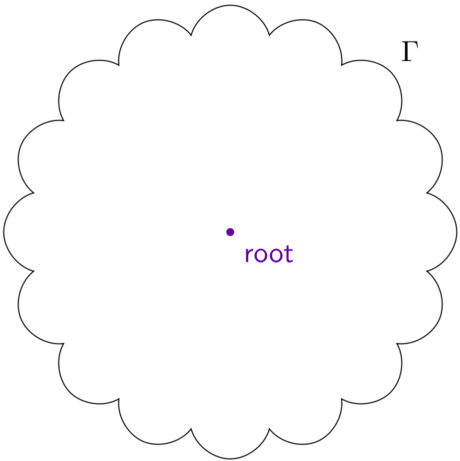
- inverse A-graph: involutive + strongly connected + deterministic + co-deterministic

End-Cones and End-Isomorphisms

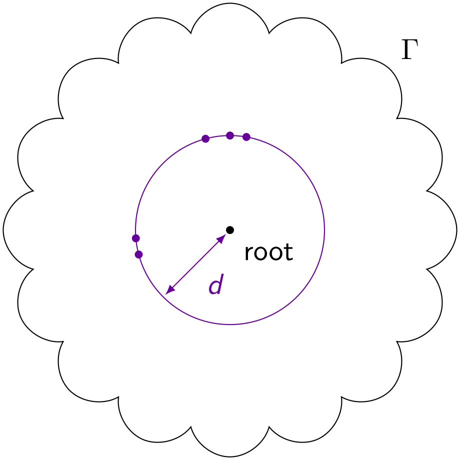
End-Cones and End-Isomorphisms



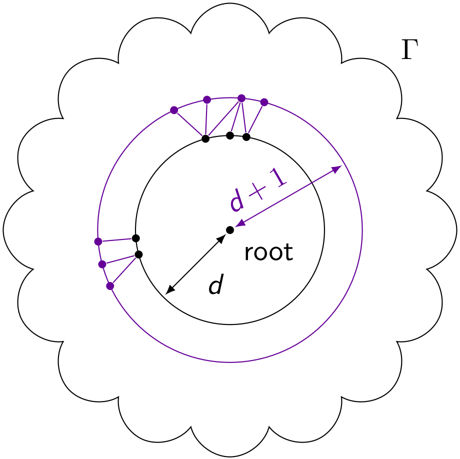
End-Cones and End-Isomorphisms



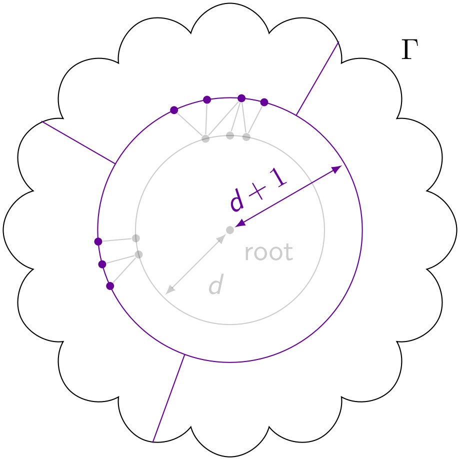
End-Cones and End-Isomorphisms



End-Cones and End-Isomorphisms

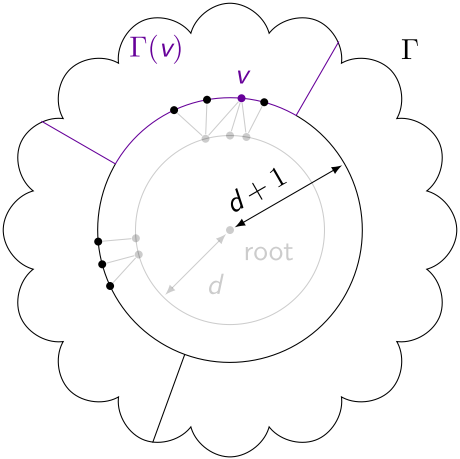


End-Cones and End-Isomorphisms



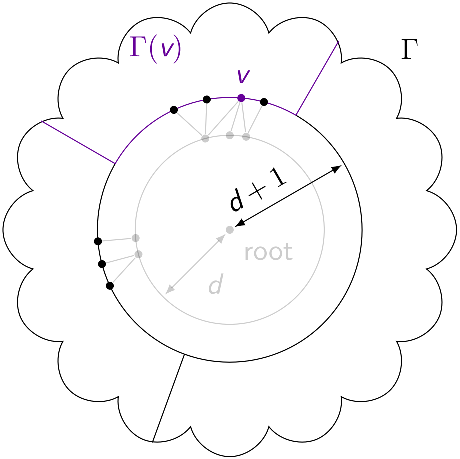
"end-cone"

End-Cones and End-Isomorphisms

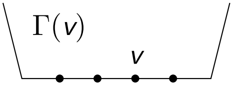


"end-cone" of v

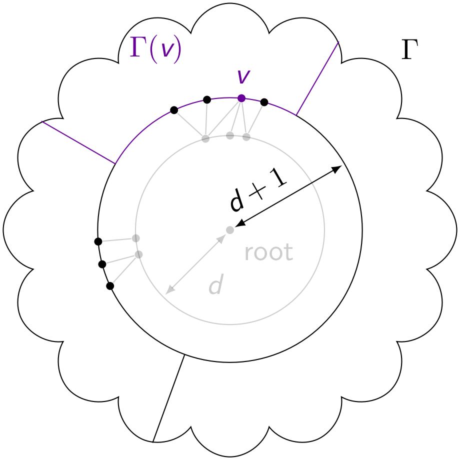
End-Cones and End-Isomorphisms



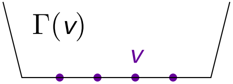
“end–cone” of v



End-Cones and End-Isomorphisms

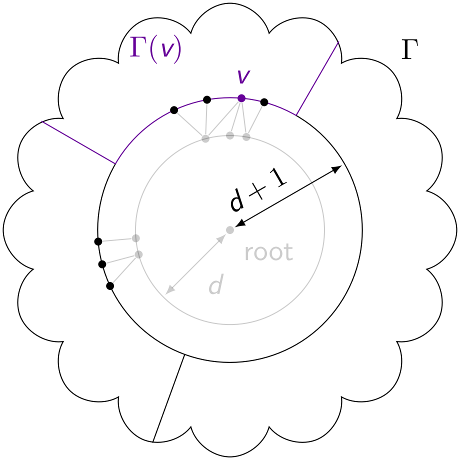


"end-cone" of v

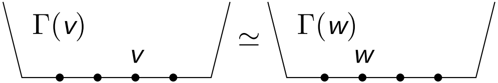


"frontier points"

End-Cones and End-Isomorphisms

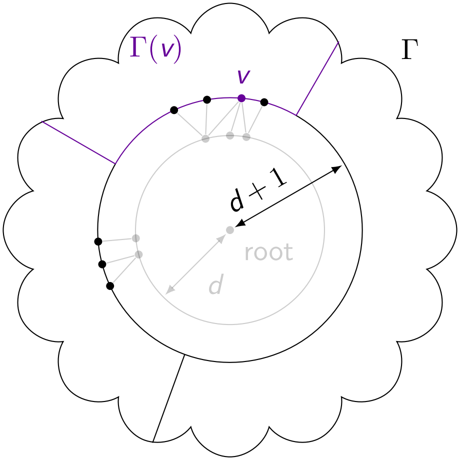


"end-cone" of v

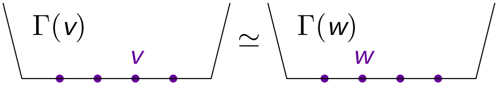


"frontier points"

End-Cones and End-Isomorphisms



"end-cone" of v



"frontier points"

"end isomorphism":

maps frontier points to frontier points

Context-Free Graphs

Definition (Muller, Schupp; 1985)

A context-free graph is a

Context-Free Graphs

Definition (Muller, Schupp; 1985)

A context-free graph is a

- rooted,

Definition (Muller, Schupp; 1985)

A context-free graph is a

- rooted,
- connected and

Context-Free Graphs

Definition (Muller, Schupp; 1985)

A context-free graph is a

- rooted,
- connected and
- involutive

Context-Free Graphs

Definition (Muller, Schupp; 1985)

A context-free graph is a

- rooted,
- connected and
- involutive
- A-graph for

Context-Free Graphs

Definition (Muller, Schupp; 1985)

A context-free graph is a

- rooted,
- connected and
- involutive
- A-graph for
- finite alphabet A with

Definition (Muller, Schupp; 1985)

A context-free graph is a

- rooted,
- connected and
- involutive
- A-graph for
- finite alphabet A with
- uniformly bounded degree and

Context-Free Graphs

Definition (Muller, Schupp; 1985)

A context-free graph is a

- rooted,
- connected and
- involutive
- A-graph for
- finite alphabet A with
- uniformly bounded degree and
- finitely many end-isomorphism classes.

Context-Free Graphs

Definition (Muller, Schupp; 1985)

A context-free graph is a

- rooted,
- connected and
- involutive
- A-graph for
- finite alphabet A with
- uniformly bounded degree and
- finitely many end-isomorphism classes.

Corollary (Muller, Schupp; 1985)

Being context-free is independent of the choice of the root.

Context-Free Graphs

Definition (Muller, Schupp; 1985)

A context-free graph is a

- rooted,
- connected and
- involutive
- A-graph for
- finite alphabet A with
- uniformly bounded degree and
- finitely many end-isomorphism classes.

Corollary (Muller, Schupp; 1985)

Being context-free is independent of the choice of the root.

Theorem (Muller, Schupp; 1985)

$$\begin{aligned} &\Gamma \text{ is context-free} \\ \iff &\Gamma = \Delta^{\pm 1} \text{ for a configuration graph of a PDA} \end{aligned}$$

Context-Free Graphs

Definition (Muller, Schupp; 1985)

A context-free graph is a

- rooted,
- connected and
- involutive
- *A*-graph for
- finite alphabet A with
- uniformly bounded degree and
- finitely many end-isomorphism classes.

Corollary (Muller, Schupp; 1985)

Being context-free is independent of the choice of the root.

Theorem (Muller, Schupp; 1985)

$$\Gamma \text{ is context-free}$$
$$\iff \Gamma = \Delta^{\pm 1} \text{ for a configuration graph of a PDA}$$

nodes: $Q \times \Gamma^*$

edges: $(p, A\gamma) \xrightarrow{a} (q, \beta\gamma)$

if $(p, A, a) \to (q, \beta) \in \delta$

Context-free graphs appear naturally as

Context-Free Graphs in the Wild

Context-free graphs appear naturally as

- Cayley graphs of context-free groups and

Context-Free Graphs in the Wild

Context-free graphs appear naturally as

- Cayley graphs of context-free groups and
- Schützenberger graphs of some finitely presented inverse monoids.

Context-Free Graphs in the Wild

Context-free graphs appear naturally as

- Cayley graphs of context-free groups and
- Schützenberger graphs of some finitely presented inverse monoids.

These are inverse graphs!

Context-Free Graphs in the Wild

Context-free graphs appear naturally as

- Cayley graphs of context-free groups and
- Schützenberger graphs of some finitely presented inverse monoids.

These are inverse graphs!

Γ: inverse $A^{\pm 1}$-graph with root u Then:

Context-Free Graphs in the Wild

Context-free graphs appear naturally as

- Cayley graphs of context-free groups and
- Schützenberger graphs of some finitely presented inverse monoids.

These are inverse graphs!

Γ: inverse $A^{\pm 1}$-graph with root u Then:

$$\begin{aligned} \Gamma \text{ is context-free} &\iff \Gamma \text{ has finitely many end-isomorphism classes} && \text{(definition)} \\ &\iff \Gamma \text{ is the configuration graph of some PDA} && \text{(Muller, Schupp, 1990)} \end{aligned}$$

Context-Free Graphs in the Wild

Context-free graphs appear naturally as

- Cayley graphs of context-free groups and
- Schützenberger graphs of some finitely presented inverse monoids.

These are inverse graphs!

Γ: inverse $A^{\pm 1}$-graph with root u Then:

$$
\begin{aligned}
\Gamma \text{ is context-free} &\iff \Gamma \text{ has finitely many end-isomorphism classes} && \text{(definition)}\\
&\iff \Gamma \text{ is the configuration graph of some PDA} && \text{(Muller, Schupp, 1990)}\\
&\iff \mathscr{L}(\Gamma; u, u) \text{ is deterministic context-free} && \text{(Rodaro, 2024)}
\end{aligned}
$$

Context-Free Graphs in the Wild

Context-free graphs appear naturally as

- Cayley graphs of context-free groups and
- Schützenberger graphs of some finitely presented inverse monoids.

These are inverse graphs!

Γ: inverse $A^{\pm 1}$-graph with root u Then:

Γ is context-free $\iff$ Γ has finitely many end-isomorphism classes (definition)

$\iff$ Γ is the configuration graph of some PDA (Muller, Schupp, 1990)

$\iff$ $\mathscr{L}(\Gamma; u, u)$ is deterministic context-free (Rodaro, 2024)

labels of cycles at u

Context-Free Graphs in the Wild

Context-free graphs appear naturally as

- Cayley graphs of context-free groups and
- Schützenberger graphs of some finitely presented inverse monoids.

These are inverse graphs!

Some details on ε-transitions and $\perp$ omitted

Γ: inverse $A^{\pm 1}$-graph with root u Then:

Γ is context-free $\iff$ Γ has finitely many end-isomorphism classes (definition)

$\iff$ Γ is the configuration graph of some PDA (Muller, Schupp, 1990)

$\iff$ $\mathscr{L}(\Gamma; u, u)$ is deterministic context-free (Rodaro, 2024)

labels of cycles at u

Context-Free Graphs in the Wild

Context-free graphs appear naturally as

- Cayley graphs of context-free groups and
- Schützenberger graphs of some finitely presented inverse monoids.

These are inverse graphs!

Some details on ε-transitions and $\perp$ omitted

Γ: inverse $A^{\pm 1}$-graph with root u Then:

Γ is context-free $\iff$ Γ has finitely many end-isomorphism classes (definition)

$\iff$ Γ is the configuration graph of some PDA (Muller, Schupp, 1990)

$\iff$ $\mathscr{L}(\Gamma; u, u)$ is deterministic context-free (Rodaro, 2024)

$\implies$ Γ is tree-like

labels of cycles at u

Context-Free Graphs in the Wild

Context-free graphs appear naturally as

- Cayley graphs of context-free groups and
- Schützenberger graphs of some finitely presented inverse monoids.

These are inverse graphs!

Some details on ε-transitions and $\perp$ omitted

Γ: inverse $A^{\pm 1}$-graph with root u Then:

Γ is context-free	$\iff$	Γ has finitely many end-isomorphism classes	(definition)
	$\iff$	Γ is the configuration graph of some PDA	(Muller, Schupp, 1990)
	$\iff$	$\mathscr{L}(\Gamma; u, u)$ is deterministic context-free	(Rodaro, 2024)
	$\iff$	Γ is tree-like	

labels of cycles at u

"with some regularity condition"

Tree-Like Graphs

Tree-Like Graphs

A directed graph is tree-like

Tree-Like Graphs

A directed graph is tree-like
$\iff$ it is quasi-isometric to a tree

Tree-Like Graphs

A directed graph is tree-like
$\iff$ it is quasi-isometric to a tree
$\iff$ it has a strong tree decomposition

Tree-Like Graphs

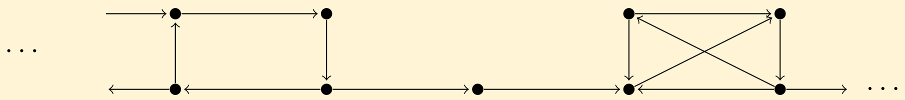
A directed graph is tree-like
$\iff$ it is quasi-isometric to a tree
$\iff$ it has a strong tree decomposition

Example

Tree-Like Graphs

A directed graph is tree-like
$\iff$ it is quasi-isometric to a tree
$\iff$ it has a strong tree decomposition

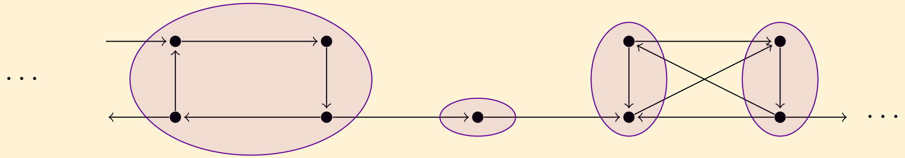
Example



Tree-Like Graphs

A directed graph is tree-like
$\iff$ it is quasi-isometric to a tree
$\iff$ it has a strong tree decomposition

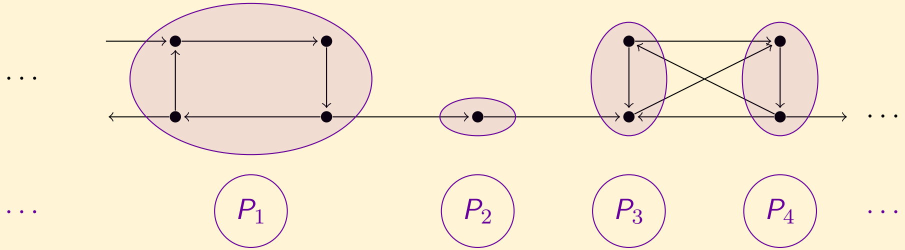
Example



Tree-Like Graphs

A directed graph is tree-like
$\iff$ it is quasi-isometric to a tree
$\iff$ it has a strong tree decomposition

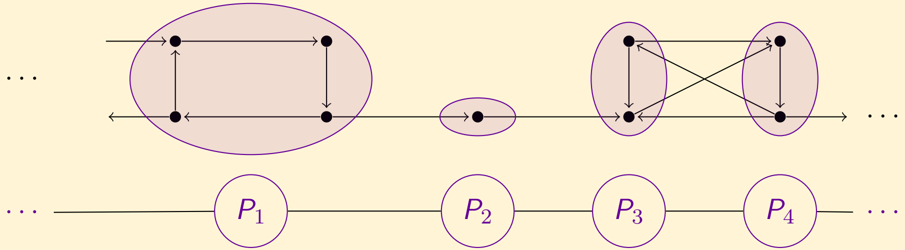
Example



Tree-Like Graphs

A directed graph is tree-like
$\iff$ it is quasi-isometric to a tree
$\iff$ it has a strong tree decomposition

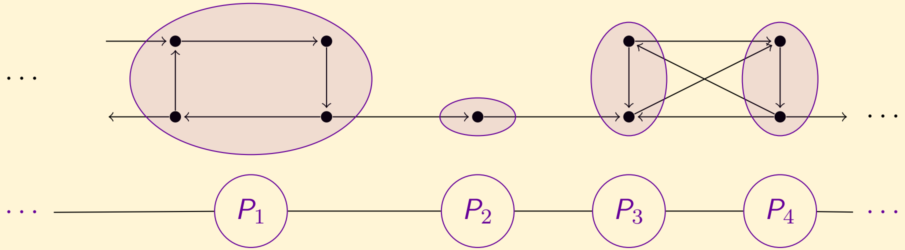
Example



Tree-Like Graphs

A directed graph is tree-like
$\iff$ it is quasi-isometric to a tree
$\iff$ it has a strong tree decomposition

Example



Idea: This tree needs to be "context-free" in some sense!

pDFAs Generating Trees

pDFAs Generating Trees

pDFA: partial deterministic finite automaton over $A^{\pm 1}$

pDFAs Generating Trees

pDFA: partial deterministic finite automaton over $A^{\pm 1}$

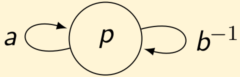
Every state p of a pDFA generates an involutive tree $\Gamma(p)$ via its tree of runs/path space.

pDFAs Generating Trees

pDFA: partial deterministic finite automaton over $A^{\pm 1}$

Every state p of a pDFA generates an involutive tree $\Gamma(p)$ via its tree of runs/path space.

Example



pDFAs Generating Trees

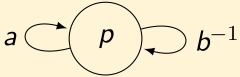
pDFA: partial deterministic finite automaton over $A^{\pm 1}$

Every state p of a pDFA generates an involutive tree $\Gamma(p)$ via its tree of runs/path space.

Example

p ε

This is the root!

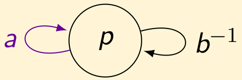


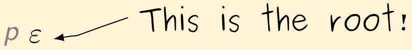
pDFAs Generating Trees

pDFA: partial deterministic finite automaton over $A^{\pm 1}$

Every state p of a pDFA generates an involutive tree $\Gamma(p)$ via its tree of runs/path space.

Example



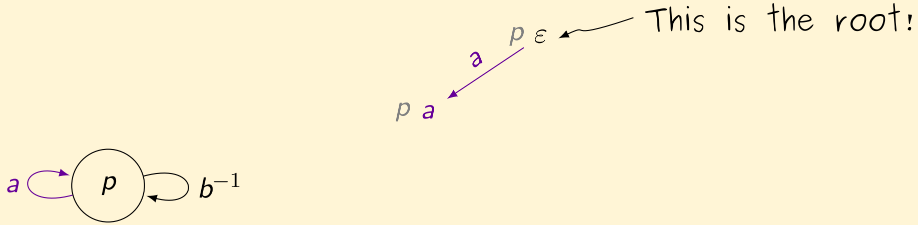


pDFAs Generating Trees

pDFA: partial deterministic finite automaton over $A^{\pm 1}$

Every state p of a pDFA generates an involutive tree $\Gamma(p)$ via its tree of runs/path space.

Example

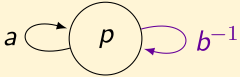


pDFAs Generating Trees

pDFA: partial deterministic finite automaton over $A^{\pm 1}$

Every state p of a pDFA generates an involutive tree $\Gamma(p)$ via its tree of runs/path space.

Example

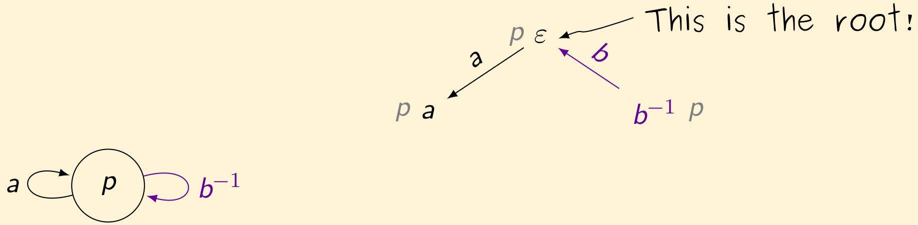


pDFAs Generating Trees

pDFA: partial deterministic finite automaton over $A^{\pm 1}$

Every state p of a pDFA generates an involutive tree $\Gamma(p)$ via its tree of runs/path space.

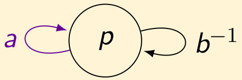
Example

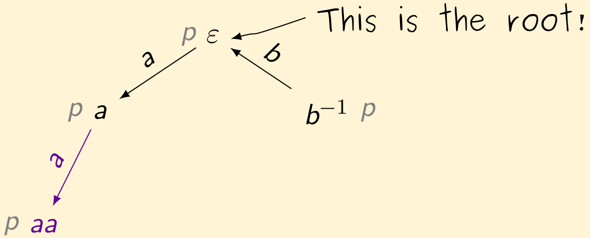


pDFA: partial deterministic finite automaton over $A^{\pm 1}$

Every state p of a pDFA generates an involutive tree $\Gamma(p)$ via its tree of runs/path space.

Example

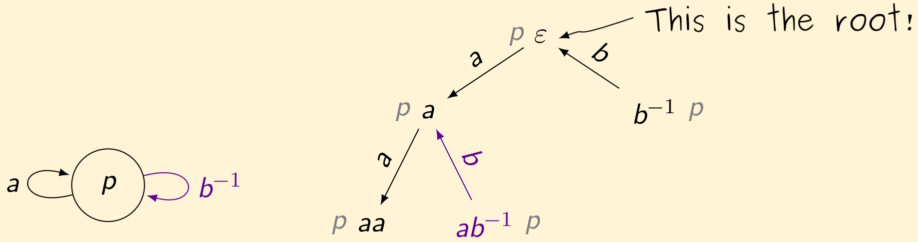




pDFAs Generating Trees

pDFA: partial deterministic finite automaton over $A^{\pm 1}$
Every state p of a pDFA generates an involutive tree $\Gamma(p)$ via its tree of runs/path space.

Example

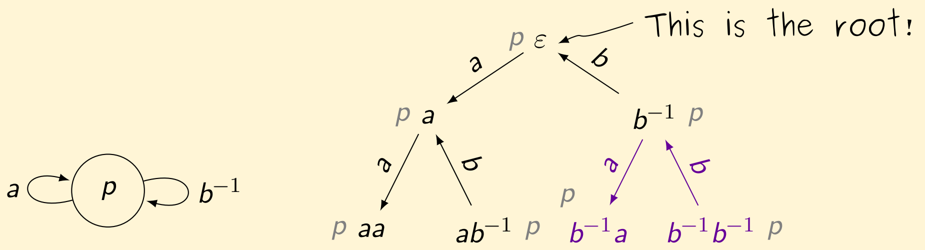


pDFAs Generating Trees

pDFA: partial deterministic finite automaton over $A^{\pm 1}$

Every state p of a pDFA generates an involutive tree $\Gamma(p)$ via its tree of runs/path space.

Example

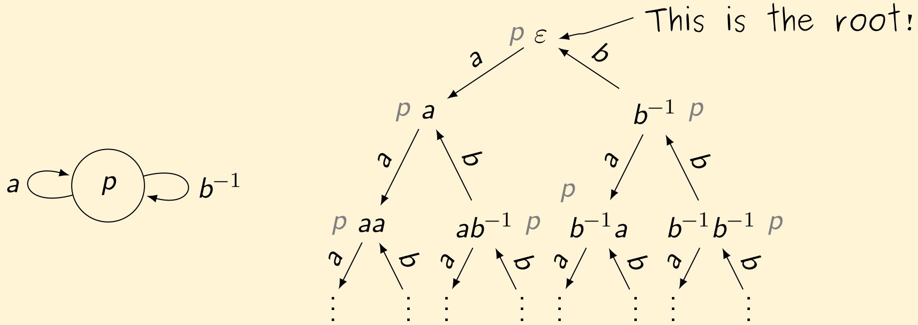


pDFAs Generating Trees

pDFA: partial deterministic finite automaton over $A^{\pm 1}$

Every state p of a pDFA generates an involutive tree $\Gamma(p)$ via its tree of runs/path space.

Example

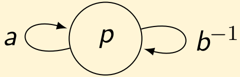


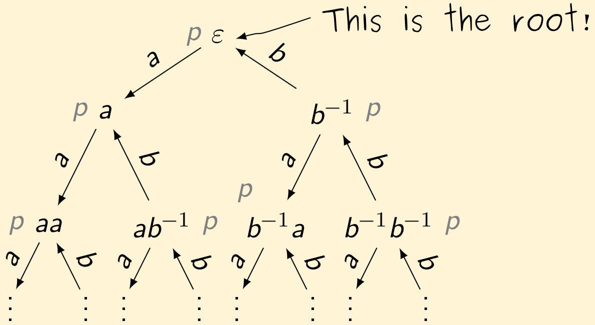
pDFAs Generating Trees

pDFA: partial deterministic finite automaton over $A^{\pm 1}$

Every state p of a pDFA generates an involutive tree $\Gamma(p)$ via its tree of runs/path space.

Example





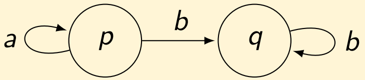
$\Gamma(p)$

pDFAs Generating Trees

pDFA: partial deterministic finite automaton over $A^{\pm 1}$

Every state p of a pDFA generates an involutive tree $\Gamma(p)$ via its tree of runs/path space.

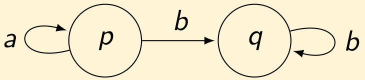
Example



pDFA: partial deterministic finite automaton over $A^{\pm 1}$

Every state p of a pDFA generates an involutive tree $\Gamma(p)$ via its tree of runs/path space.

Example



$\Gamma(p)$

pDFAs Generating Trees

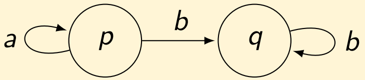
pDFA: partial deterministic finite automaton over $A^{\pm 1}$

Every state p of a pDFA generates an involutive tree $\Gamma(p)$ via its tree of runs/path space.

Example

p

ε



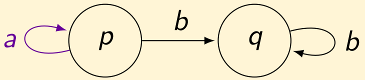
$\Gamma(p)$

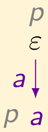
pDFAs Generating Trees

pDFA: partial deterministic finite automaton over $A^{\pm 1}$

Every state p of a pDFA generates an involutive tree $\Gamma(p)$ via its tree of runs/path space.

Example





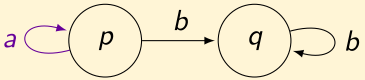
$\Gamma(p)$

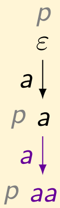
pDFAs Generating Trees

pDFA: partial deterministic finite automaton over $A^{\pm 1}$

Every state p of a pDFA generates an involutive tree $\Gamma(p)$ via its tree of runs/path space.

Example





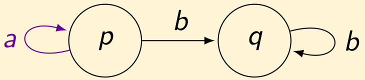
$\Gamma(p)$

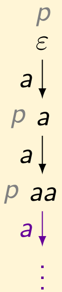
pDFAs Generating Trees

pDFA: partial deterministic finite automaton over $A^{\pm 1}$

Every state p of a pDFA generates an involutive tree $\Gamma(p)$ via its tree of runs/path space.

Example





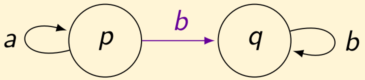
$\Gamma(p)$

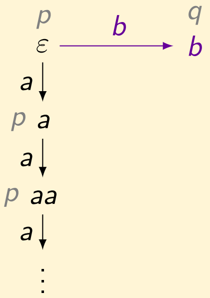
pDFAs Generating Trees

pDFA: partial deterministic finite automaton over $A^{\pm 1}$

Every state p of a pDFA generates an involutive tree $\Gamma(p)$ via its tree of runs/path space.

Example





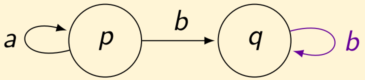
$\Gamma(p)$

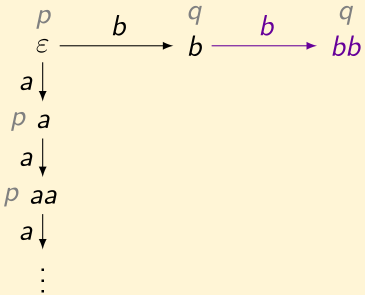
pDFAs Generating Trees

pDFA: partial deterministic finite automaton over $A^{\pm 1}$

Every state p of a pDFA generates an involutive tree $\Gamma(p)$ via its tree of runs/path space.

Example





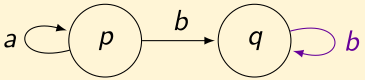
$\Gamma(p)$

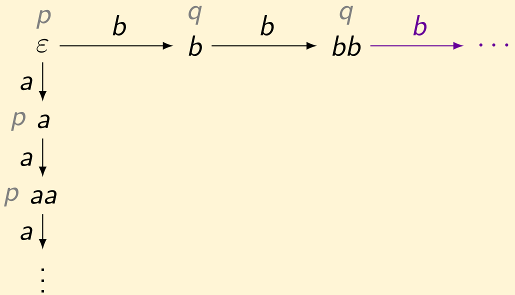
pDFAs Generating Trees

pDFA: partial deterministic finite automaton over $A^{\pm 1}$

Every state p of a pDFA generates an involutive tree $\Gamma(p)$ via its tree of runs/path space.

Example





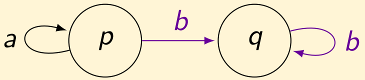
$\Gamma(p)$

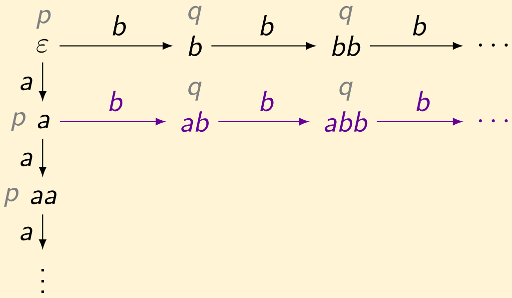
pDFAs Generating Trees

pDFA: partial deterministic finite automaton over $A^{\pm 1}$

Every state p of a pDFA generates an involutive tree $\Gamma(p)$ via its tree of runs/path space.

Example





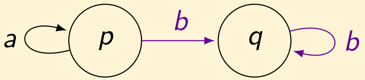
$\Gamma(p)$

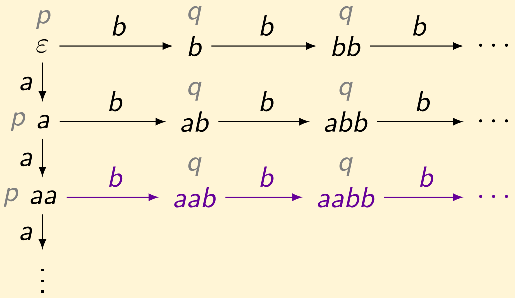
pDFAs Generating Trees

pDFA: partial deterministic finite automaton over $A^{\pm 1}$

Every state p of a pDFA generates an involutive tree $\Gamma(p)$ via its tree of runs/path space.

Example





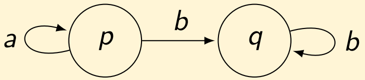
$\Gamma(p)$

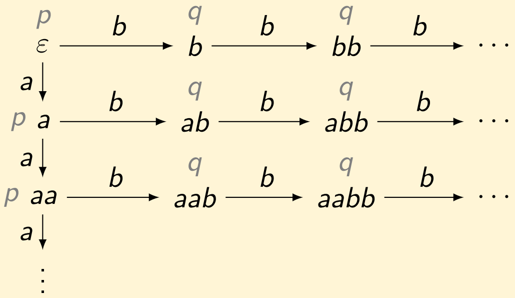
pDFAs Generating Trees

pDFA: partial deterministic finite automaton over $A^{\pm 1}$

Every state p of a pDFA generates an involutive tree $\Gamma(p)$ via its tree of runs/path space.

Example





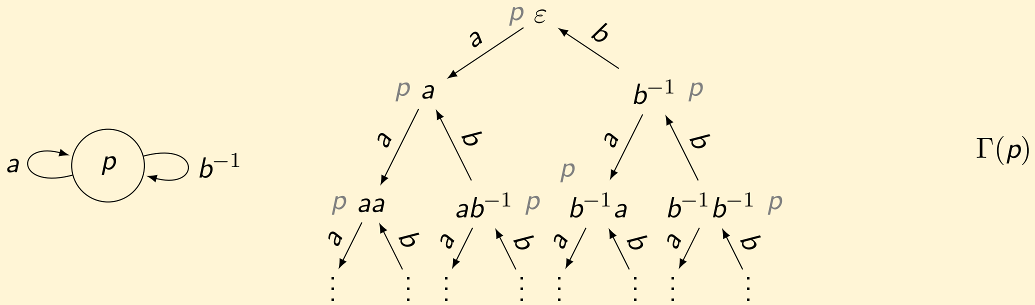
$\Gamma(p)$

pDFAs Generating Trees

pDFA: partial deterministic finite automaton over $A^{\pm 1}$

Every state p of a pDFA generates an involutive tree $\Gamma(p)$ via its tree of runs/path space.

Example

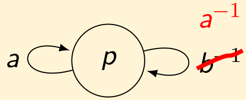


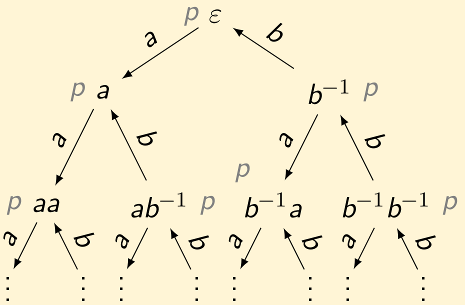
pDFAs Generating Trees

pDFA: partial deterministic finite automaton over $A^{\pm 1}$

Every state p of a pDFA generates an involutive tree $\Gamma(p)$ via its tree of runs/path space.

Example



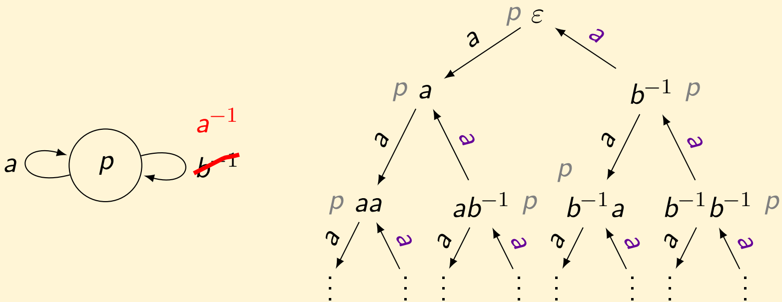


$\Gamma(p)$

pDFA: partial deterministic finite automaton over $A^{\pm 1}$

Every state p of a pDFA generates an involutive tree $\Gamma(p)$ via its tree of runs/path space.

Example



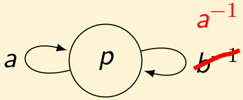
$\Gamma(p)$

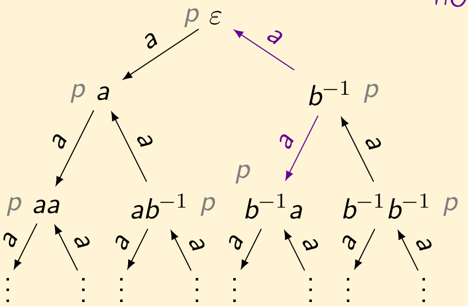
pDFAs Generating Trees

pDFA: partial deterministic finite automaton over $A^{\pm 1}$

Every state p of a pDFA generates an involutive tree $\Gamma(p)$ via its tree of runs/path space.

Example





not deterministic!

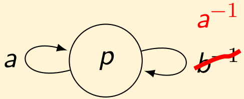
$\Gamma(p)$

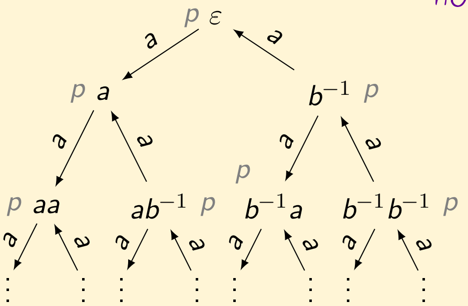
pDFA: partial deterministic finite automaton over $A^{\pm 1}$

Every state p of a pDFA generates an involutive tree $\Gamma(p)$ via its tree of runs/path space.

Example

not deterministic!





$\Gamma(p)$

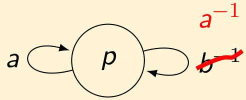
$\Gamma(p)$ is inverse and deterministic $\iff$ aa^{-1} cannot be read for any $a \in A^{\pm 1}$

pDFAs Generating Trees

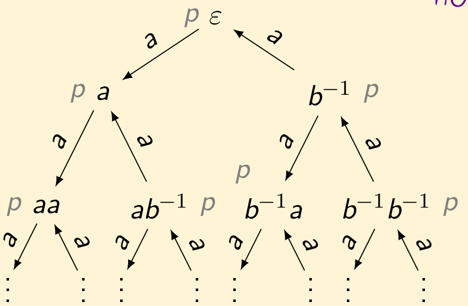
pDFA: partial deterministic finite automaton over $A^{\pm 1}$

Every state p of a pDFA generates an involutive tree $\Gamma(p)$ via its tree of runs/path space.

Example



not deterministic!



$\Gamma(p)$ is inverse and deterministic $\iff$ aa^{-1} cannot be read for any $a \in A^{\pm 1}$

$\iff$: the pDFA is reduced

Context-Free Trees

Γ: inverse $A^{\pm 1}$-tree, i. e. a tree as an unlabeled, undirected graph

Context-Free Trees

Γ: inverse $A^{\pm 1}$-tree, i. e. a tree as an unlabeled, undirected graph

Theorem (W., arXiv 2026)

Γ *is context-free*

Context-Free Trees

Γ: inverse $A^{\pm 1}$-tree, i. e. a tree as an unlabeled, undirected graph

Theorem (W., arXiv 2026)

Γ *is context-free* $\iff$ Γ *arises as* $\Gamma(p)$ *for a state* p *of a reduced pDFA*

Context-Free Trees

Γ: inverse $A^{\pm 1}$-tree, i. e. a tree as an unlabeled, undirected graph

Theorem (W., arXiv 2026)

Γ is context-free $\iff$ Γ arises as $\Gamma(p)$ for a state p of a reduced pDFA

Context-free trees

- arise naturally as Schützenberger graphs of

Context-Free Trees

Γ: inverse $A^{\pm 1}$-tree, i. e. a tree as an unlabeled, undirected graph

Theorem (W., arXiv 2026)

Γ is context-free $\iff$ Γ arises as $\Gamma(p)$ for a state p of a reduced pDFA

Context-free trees

- arise naturally as Schützenberger graphs of
  - free inverse monoids (Munn trees) .

Context-Free Trees

Γ: inverse $A^{\pm 1}$-tree, i. e. a tree as an unlabeled, undirected graph

Theorem (W., arXiv 2026)

Γ is context-free $\iff$ Γ arises as $\Gamma(p)$ for a state p of a reduced pDFA

Context-free trees

- arise naturally as Schützenberger graphs of
  - free inverse monoids (Munn trees) and as Caley graphs of free groups.

Γ: inverse $A^{\pm 1}$-tree, i. e. a tree as an unlabeled, undirected graph

Theorem (W., arXiv 2026)

Γ is context-free $\iff$ Γ arises as $\Gamma(p)$ for a state p of a reduced pDFA

Context-free trees

- arise naturally as Schützenberger graphs of
  - free inverse monoids (Munn trees) and as Caley graphs of free groups.
  - inverse monoids with idempotent relators.

Context-Free Trees

Γ: inverse $A^{\pm 1}$-tree, i. e. a tree as an unlabeled, undirected graph

Theorem (W., arXiv 2026)

Γ is context-free $\iff$ Γ arises as $\Gamma(p)$ for a state p of a reduced pDFA

Context-free trees

- arise naturally as Schützenberger graphs of
  - free inverse monoids (Munn trees) and as Caley graphs of free groups.
  - inverse monoids with idempotent relators.
- hopefully serve as a stepping stone for general context-free graphs.

Context-Free Trees

Γ: inverse $A^{\pm 1}$-tree, i. e. a tree as an unlabeled, undirected graph

Theorem (W., arXiv 2026)

Γ is context-free $\iff$ Γ arises as $\Gamma(p)$ for a state p of a reduced pDFA

Context-free trees

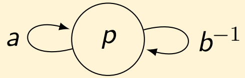
- arise naturally as Schützenberger graphs of
  - free inverse monoids (Munn trees) and as Caley graphs of free groups.
  - inverse monoids with idempotent relators.
- hopefully serve as a stepping stone for general context-free graphs.

Stephen (1990): Deciding Green's relations $(\mathcal{R}, \mathcal{L}, \mathcal{D}, \mathcal{J})$ is closely related to the (rooted) isomorphism (homomorphism) problem for the Schützenberger graphs.

The Rooted Isomorphism Problem

The Rooted Isomorphism Problem

Example



The Rooted Isomorphism Problem

$\mathscr{L}(p)$: words readable from state p in its pDFA

Example

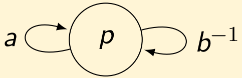
$$p \xrightarrow{a} p, \quad p \xrightarrow{b^{-1}} p$$

$\mathscr{L}(p) = \{a, b^{-1}\}^*$

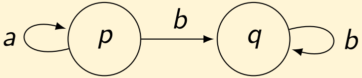
The Rooted Isomorphism Problem

$\mathscr{L}(p)$: words readable from state p in its pDFA

Example



$\mathscr{L}(p) = \{a, b^{-1}\}^*$

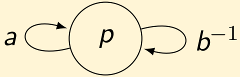


$\mathscr{L}(p) = a^* b^*$

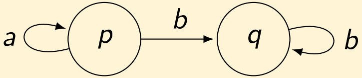
The Rooted Isomorphism Problem

$\mathscr{L}(p)$: words readable from state p in its pDFA

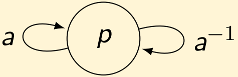
Example



$\mathscr{L}(p) = \{a, b^{-1}\}^*$



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

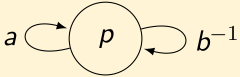


$\mathscr{L}(p) = \{a, a^{-1}\}^*$

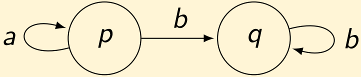
The Rooted Isomorphism Problem

$\mathscr{L}(p)$: words readable from state p in its pDFA

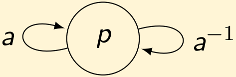
Example



$\mathscr{L}(p) = \{a, b^{-1}\}^*$



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



$\mathscr{L}(p) = \{a, a^{-1}\}^*$

Fact

Let p and q be states of reduced pDFAs. Then:

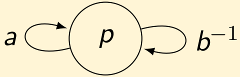
$$\Gamma(p) \doteq \Gamma(q) \iff \mathscr{L}(p) = \mathscr{L}(q)$$

isomorphic as **rooted** graphs

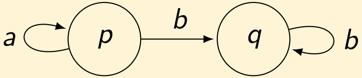
The Rooted Isomorphism Problem

$\mathscr{L}(p)$: words readable from state p in its pDFA

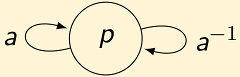
Example



$\mathscr{L}(p) = \{a, b^{-1}\}^*$



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



$\mathscr{L}(p) = \{a, a^{-1}\}^*$

Fact

Let p and q be states of reduced pDFAs. Then:

$$\Gamma(p) \doteqdot \Gamma(q) \iff \mathscr{L}(p) = \mathscr{L}(q)$$

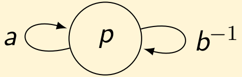
*isomorphic as **rooted** graphs*

It is well known that the latter is NL-complete.

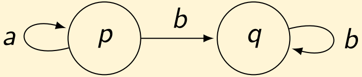
The Rooted Isomorphism Problem

$\mathscr{L}(p)$: words readable from state p in its pDFA

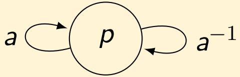
Example



$\mathscr{L}(p) = \{a, b^{-1}\}^*$



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



$\mathscr{L}(p) = \{a, a^{-1}\}^*$

Fact

Let p and q be states of reduced pDFAs. Then:

$$\Gamma(p) \stackrel{\vee}{=} \Gamma(q) \iff \mathscr{L}(p) = \mathscr{L}(q)$$

isomorphic as **rooted** graphs

It is well known that the latter is NL-complete.

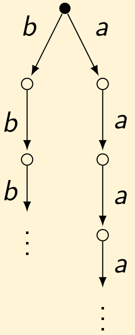
Adapting a construction by Cho & Huynh, we may assume every state to be accessible from p and q, respectively (for the lower bound).

The Non-Rooted Case: The Problem

The Non-Rooted Case: The Problem

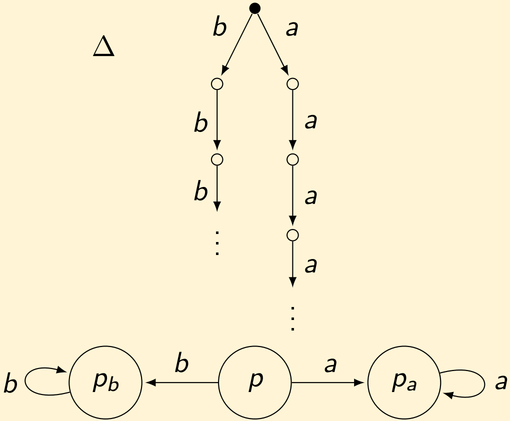
Example

Δ



The Non-Rooted Case: The Problem

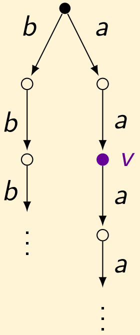
Example

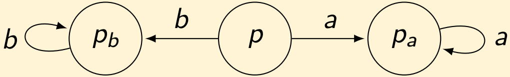


The Non-Rooted Case: The Problem

Example

Δ

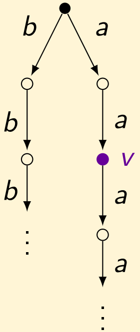




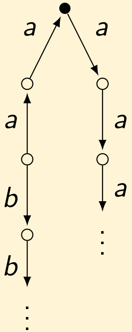
The Non-Rooted Case: The Problem

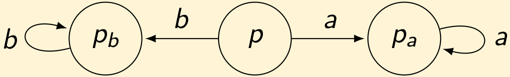
Example

Δ



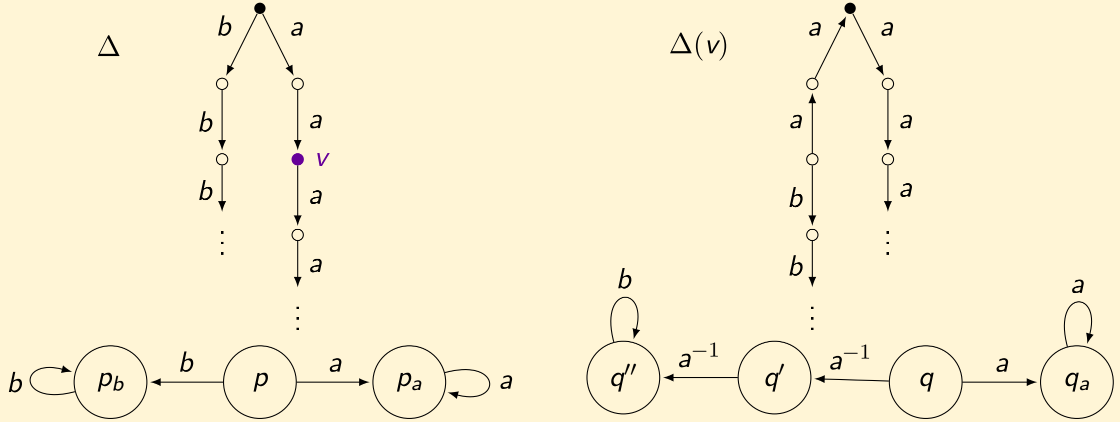
$\Delta(v)$



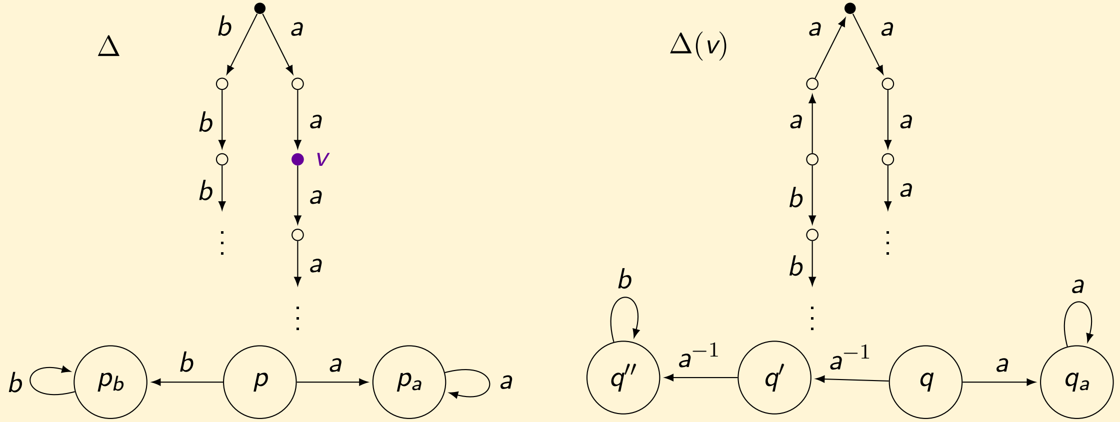


The Non-Rooted Case: The Problem

Example



Example



The languages don't seem connected!

The Non-Rooted Isomorphism Problem

Theorem (W., arXiv 2026)

The non-rooted isomorphism problem for inverse context-free trees

Input:	*states* $\check{p}$ *and* $\check{q}$ *of reduced pDFAs*
Question:	*are* $\Gamma(\check{p})$ *and* $\Gamma(\check{q})$ *isomorphic as non-rooted graphs*

is in NL*-complete.*

The Non-Rooted Isomorphism Problem

Theorem (W., arXiv 2026)

The non-rooted isomorphism problem for inverse context-free trees

Input:	*states* $\check{p}$ *and* $\check{q}$ *of reduced pDFAs*
Question:	*are* $\Gamma(\check{p})$ *and* $\Gamma(\check{q})$ *isomorphic as non-rooted graphs*

is in NL*-complete.*

The tricky part is "$\in \mathrm{NL}$".

Proof Idea

Proof Idea

p: state of reduced pDFA $\mathcal{A}$,

Proof Idea

p: state of reduced pDFA $\mathcal{A}$, $\check{q}$: state of a reduced pDFA $\mathcal{B}$,

Proof Idea

p: state of reduced pDFA $\mathcal{A}$, $\check{q}$: state of a reduced pDFA $\mathcal{B}$, $\Delta = \Gamma(\check{q})$

Proof Idea

p: state of reduced pDFA $\mathcal{A}$, $\check{q}$: state of a reduced pDFA $\mathcal{B}$, $\Delta = \Gamma(\check{q})$

Predicate: $U(p, q)$

Proof Idea

p: state of reduced pDFA $\mathcal{A}$, $\check{q}$: state of a reduced pDFA $\mathcal{B}$, $\Delta = \Gamma(\check{q})$

Predicate: $U(p, q) \iff \exists$ node $v \in \Delta$

Proof Idea

p: state of reduced pDFA $\mathcal{A}$, $\check{q}$: state of a reduced pDFA $\mathcal{B}$, $\Delta = \Gamma(\check{q})$

Predicate: $U(p, q) \iff \exists$ node $v \in \Delta$ labeled by q

Proof Idea

p: state of reduced pDFA $\mathcal{A}$, $\check{q}$: state of a reduced pDFA $\mathcal{B}$, $\Delta = \Gamma(\check{q})$

Predicate: $U(p, q) \iff \exists\,\text{node } v \in \Delta \text{ labeled by } q : \Gamma(p) \doteqdot \Delta(v)$

Proof Idea

p: state of reduced pDFA $\mathcal{A}$, $\check{q}$: state of a reduced pDFA $\mathcal{B}$, $\Delta = \Gamma(\check{q})$

Predicate: $U(p, q) \iff \exists$ node $v \in \Delta$ labeled by $q : \Gamma(p) \doteq \Delta(v)$

Case: v is the root

Proof Idea

p: state of reduced pDFA $\mathcal{A}$, $\check{q}$: state of a reduced pDFA $\mathcal{B}$, $\Delta = \Gamma(\check{q})$

Predicate: $U(p, q) \iff \exists\,\text{node } v \in \Delta \text{ labeled by } q : \Gamma(p) \doteq \Delta(v)$

Case: v is the root

$\implies q = \check{q}$,

Proof Idea

p: state of reduced pDFA $\mathcal{A}$, $\check{q}$: state of a reduced pDFA $\mathcal{B}$, $\Delta = \Gamma(\check{q})$

Predicate: $U(p, q) \iff \exists\,\text{node } v \in \Delta \text{ labeled by } q : \Gamma(p) \doteq \Delta(v)$

Case: v is the root

$$\implies q = \check{q},\ \Delta(v) = \Delta$$

Proof Idea

p: state of reduced pDFA $\mathcal{A}$, $\check{q}$: state of a reduced pDFA $\mathcal{B}$, $\Delta = \Gamma(\check{q})$

Predicate: $U(p, q) \iff \exists$ node $v \in \Delta$ labeled by $q : \Gamma(p) \doteq \Delta(v)$

Case: v is the root

$\implies q = \check{q}, \ \Delta(v) = \Delta$, i.e. $U(p, \check{q})$

Proof Idea

p: state of reduced pDFA $\mathcal{A}$, $\check{q}$: state of a reduced pDFA $\mathcal{B}$, $\Delta = \Gamma(\check{q})$

Predicate: $U(p, q) \iff \exists\,\text{node } v \in \Delta \text{ labeled by } q : \Gamma(p) \doteq \Delta(v)$

Case: v is the root

$$\implies q = \check{q},\ \Delta(v) = \Delta, \text{ i.e. } U(p, \check{q})$$

$$\Gamma(p) \doteq \Delta(v) = \Delta = \Gamma(\check{q})$$

Proof Idea

p: state of reduced pDFA $\mathcal{A}$, $\check{q}$: state of a reduced pDFA $\mathcal{B}$, $\Delta = \Gamma(\check{q})$

Predicate: $U(p, q) \iff \exists\,$node $v \in \Delta$ labeled by $q : \Gamma(p) \doteqdot \Delta(v)$

Case: v is the root

$\implies q = \check{q},\ \Delta(v) = \Delta$, i.e. $U(p, \check{q})$

$\Gamma(p) \doteqdot \Delta(v) = \Delta = \Gamma(\check{q}) \iff \mathscr{L}(p) = \mathscr{L}(\check{q})$

Proof Idea

p: state of reduced pDFA $\mathcal{A}$, $\check{q}$: state of a reduced pDFA $\mathcal{B}$, $\Delta = \Gamma(\check{q})$

Predicate: $U(p, q) \iff \exists$ node $v \in \Delta$ labeled by $q : \Gamma(p) \doteq \Delta(v)$

Case: v is the root

$$\implies q = \check{q},\ \Delta(v) = \Delta, \text{ i.e. } U(p, \check{q})$$

$$\Gamma(p) \doteq \Delta(v) = \Delta = \Gamma(\check{q}) \iff \mathscr{L}(p) = \mathscr{L}(\check{q})$$

Thus: $U(p, q) \iff \big(q = \check{q} \text{ and } \mathscr{L}(p) = \mathscr{L}(\check{q})\big)$

Proof Idea

p: state of reduced pDFA $\mathcal{A}$, $\check{q}$: state of a reduced pDFA $\mathcal{B}$, $\Delta = \Gamma(\check{q})$

Predicate: $U(p, q) \iff \exists$ node $v \in \Delta$ labeled by $q : \Gamma(p) \doteq \Delta(v)$

Case: v is the root

$\implies q = \check{q}$, $\Delta(v) = \Delta$, i.e. $U(p, \check{q})$

$\Gamma(p) \doteq \Delta(v) = \Delta = \Gamma(\check{q}) \iff \mathscr{L}(p) = \mathscr{L}(\check{q})$

Thus: $U(p, q) \iff \left(q = \check{q} \text{ and } \mathscr{L}(p) = \mathscr{L}(\check{q})\right)$ or "v is not the root"

Proof Idea

p: state of reduced pDFA $\mathcal{A}$, $\check{q}$: state of a reduced pDFA $\mathcal{B}$, $\Delta = \Gamma(\check{q})$

Predicate: $U(p, q) \iff \exists$ node $v \in \Delta$ labeled by $q : \Gamma(p) \doteq \Delta(v)$

Case: v is not the root

Proof Idea

p: state of reduced pDFA $\mathcal{A}$, $\check{q}$: state of a reduced pDFA $\mathcal{B}$, $\Delta = \Gamma(\check{q})$

Predicate: $U(p, q) \iff \exists$ node $v \in \Delta$ labeled by $q : \Gamma(p) \doteq \Delta(v)$

Case: v is not the root

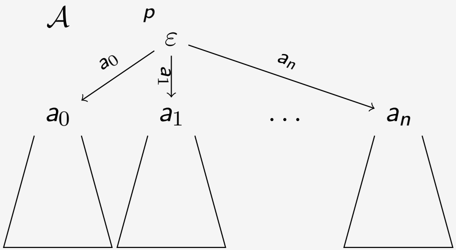
$\mathcal{A}$ p ε

Proof Idea

p: state of reduced pDFA $\mathcal{A}$, $\check{q}$: state of a reduced pDFA $\mathcal{B}$, $\Delta = \Gamma(\check{q})$

Predicate: $U(p, q) \iff \exists$ node $v \in \Delta$ labeled by $q : \Gamma(p) \doteq \Delta(v)$

Case: v is not the root

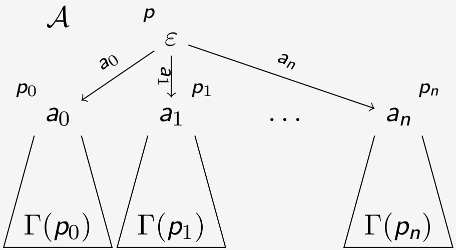


Proof Idea

p: state of reduced pDFA $\mathcal{A}$, $\check{q}$: state of a reduced pDFA $\mathcal{B}$, $\Delta = \Gamma(\check{q})$

Predicate: $U(p, q) \iff \exists$ node $v \in \Delta$ labeled by $q : \Gamma(p) \doteq \Delta(v)$

Case: v is not the root

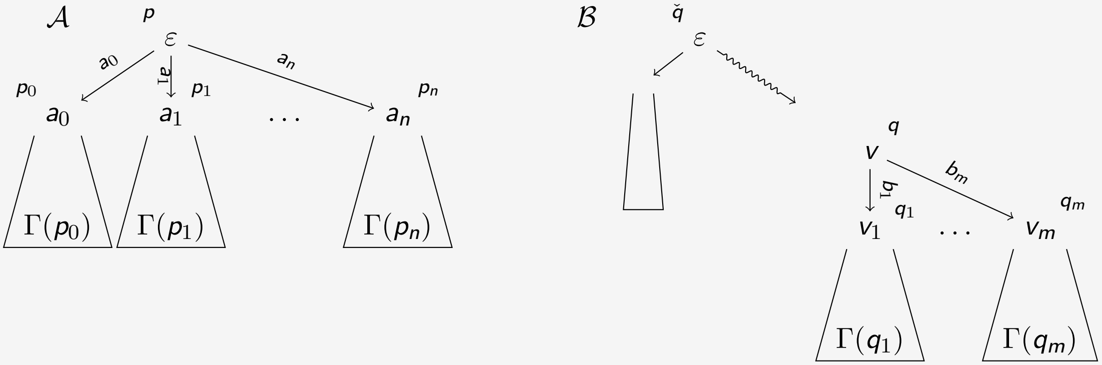


Proof Idea

p: state of reduced pDFA $\mathcal{A}$, $\check{q}$: state of a reduced pDFA $\mathcal{B}$, $\Delta = \Gamma(\check{q})$

Predicate: $U(p, q) \iff \exists$ node $v \in \Delta$ labeled by $q : \Gamma(p) \doteq \Delta(v)$

Case: v is not the root

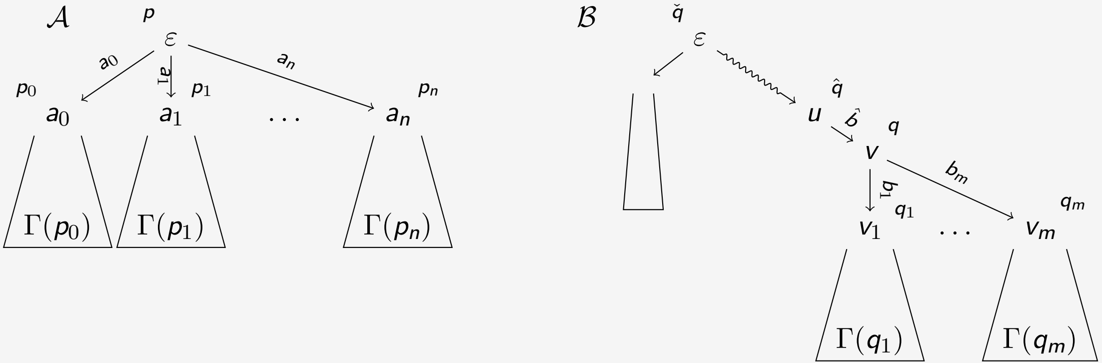


Proof Idea

p: state of reduced pDFA $\mathcal{A}$, $\check{q}$: state of a reduced pDFA $\mathcal{B}$, $\Delta = \Gamma(\check{q})$

Predicate: $U(p, q) \iff \exists\,\text{node } v \in \Delta \text{ labeled by } q : \Gamma(p) \doteq \Delta(v)$

Case: v is not the root

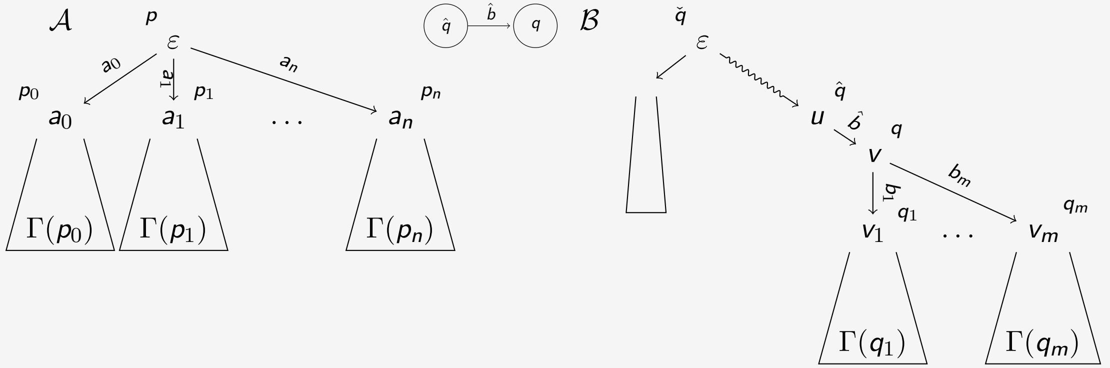


Proof Idea

p: state of reduced pDFA $\mathcal{A}$, $\check{q}$: state of a reduced pDFA $\mathcal{B}$, $\Delta = \Gamma(\check{q})$

Predicate: $U(p, q) \iff \exists\,\text{node } v \in \Delta \text{ labeled by } q : \Gamma(p) \doteq \Delta(v)$

Case: v is not the root

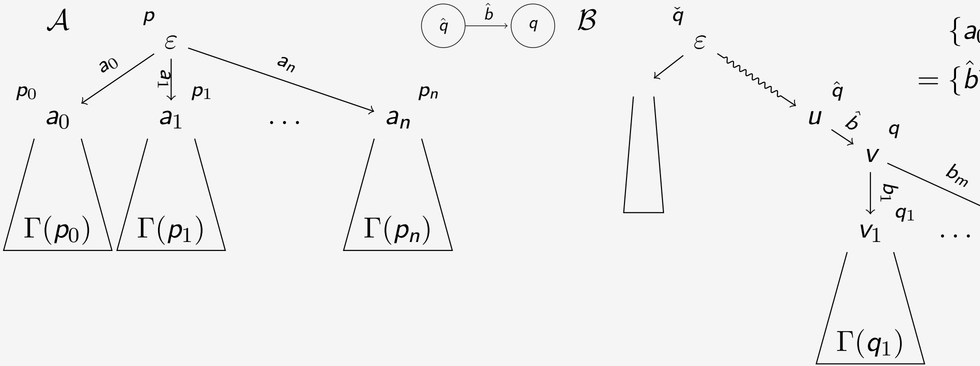


Proof Idea

p: state of reduced pDFA $\mathcal{A}$, $\quad \check{q}$: state of a reduced pDFA $\mathcal{B}$, $\quad \Delta = \Gamma(\check{q})$

Predicate: $\quad U(p,q) \quad \Longleftrightarrow \exists$ node $v \in \Delta$ labeled by $q : \Gamma(p) \doteq \Delta(v)$

Case: v is not the root



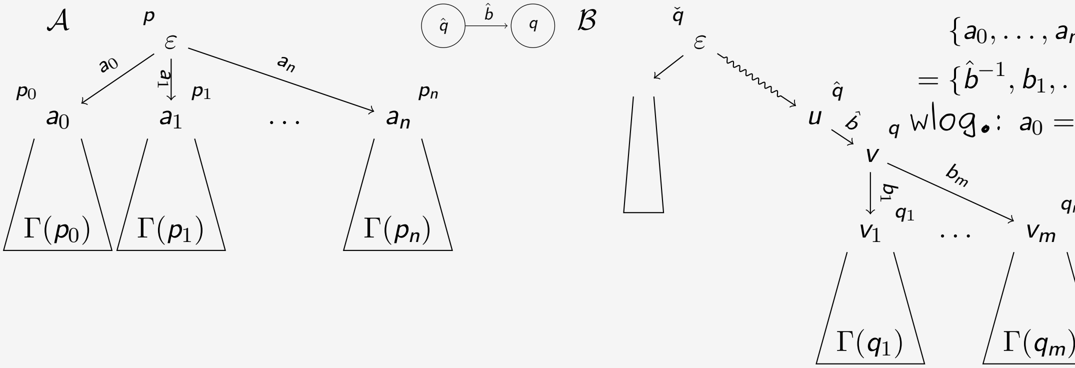
$$\{a_0, \dots, a_n\}$$
$$= \{\hat{b}^{-1}, b_1, \dots, b_m\}$$

Proof Idea

p: state of reduced pDFA $\mathcal{A}$, $\check{q}$: state of a reduced pDFA $\mathcal{B}$, $\Delta = \Gamma(\check{q})$

Predicate: $U(p, q) \iff \exists$ node $v \in \Delta$ labeled by $q : \Gamma(p) \doteq \Delta(v)$

Case: v is not the root



$\{a_0, \dots, a_n\}$

$= \{\hat{b}^{-1}, b_1, \dots, b_m\}$

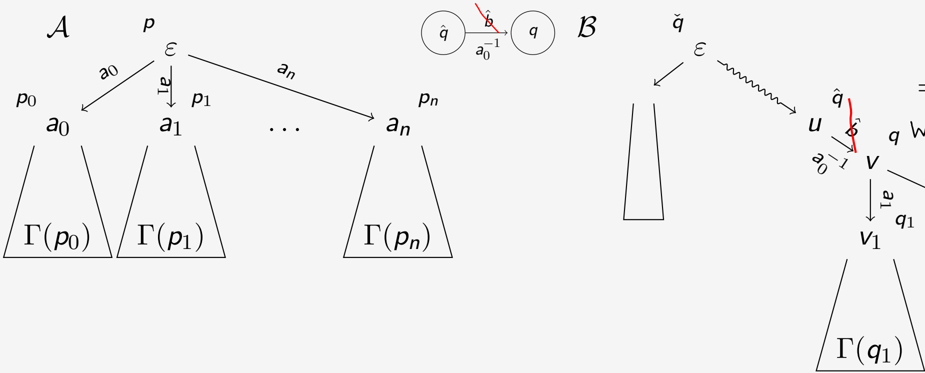
wlog.: $a_0 = \hat{b}^{-1}$

Proof Idea

p: state of reduced pDFA $\mathcal{A}$, $\check{q}$: state of a reduced pDFA $\mathcal{B}$, $\Delta = \Gamma(\check{q})$

Predicate: $U(p,q) \iff \exists$ node $v \in \Delta$ labeled by $q : \Gamma(p) \doteq \Delta(v)$

Case: v is not the root



$\{a_0, \dots, a_n\}$

$= \{\hat{b}^{-1}, b_1, \dots, b_m\}$

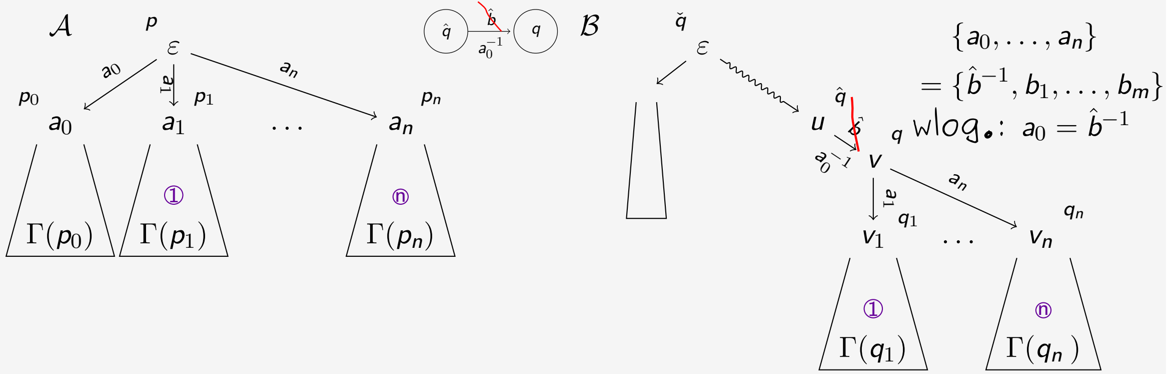
wlog.: $a_0 = \hat{b}^{-1}$

Proof Idea

p: state of reduced pDFA $\mathcal{A}$, $\check{q}$: state of a reduced pDFA $\mathcal{B}$, $\Delta = \Gamma(\check{q})$

Predicate: $U(p, q) \iff \exists$ node $v \in \Delta$ labeled by $q : \Gamma(p) \doteq \Delta(v)$

Case: v is not the root

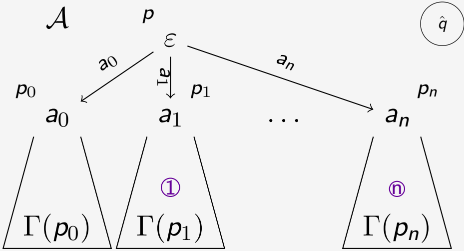


Proof Idea

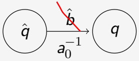
p: state of reduced pDFA $\mathcal{A}$, $\check{q}$: state of a reduced pDFA $\mathcal{B}$, $\Delta = \Gamma(\check{q})$

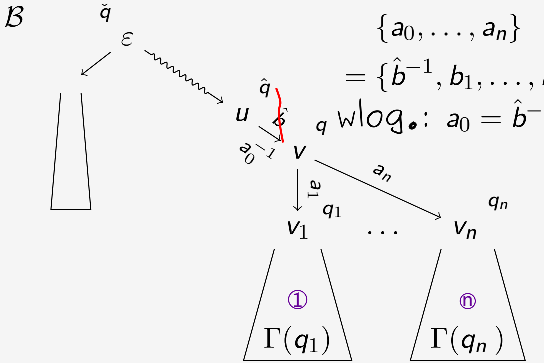
Predicate: $U(p,q) \iff \exists$ node $v \in \Delta$ labeled by $q : \Gamma(p) \doteq \Delta(v)$

Case: v is not the root



$\exists a_0 \in \mathscr{L}(p) : \mathscr{L}(p) \setminus a_0 A^{\pm *} = \mathscr{L}(q)$





$\{a_0, \dots, a_n\}$

$= \{\hat{b}^{-1}, b_1, \dots, b_m\}$

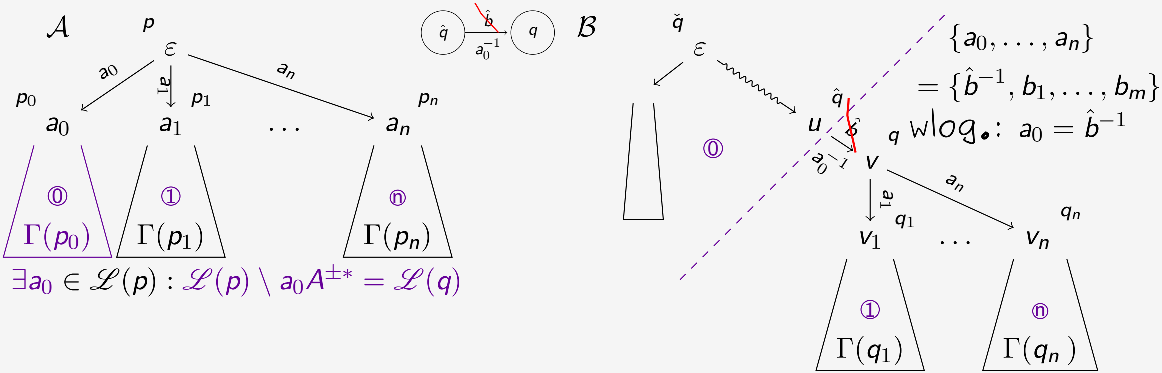
wlog.: $a_0 = \hat{b}^{-1}$

Proof Idea

p: state of reduced pDFA $\mathcal{A}$, $\check{q}$: state of a reduced pDFA $\mathcal{B}$, $\Delta = \Gamma(\check{q})$

Predicate: $U(p, q) \iff \exists$ node $v \in \Delta$ labeled by $q : \Gamma(p) \doteq \Delta(v)$

Case: v is not the root

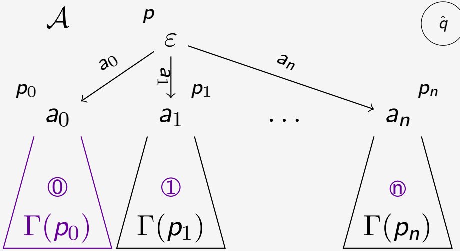


$\exists a_0 \in \mathscr{L}(p) : \mathscr{L}(p) \setminus a_0 A^{\pm *} = \mathscr{L}(q)$

p: state of reduced pDFA $\mathcal{A}$, $\check{q}$: state of a reduced pDFA $\mathcal{B}$, $\Delta = \Gamma(\check{q})$

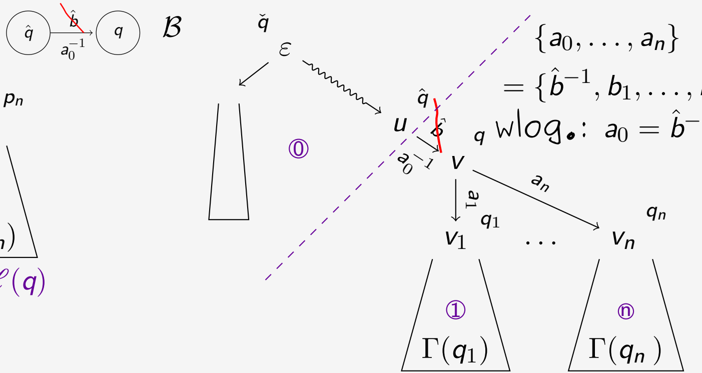
Predicate: $U(p, q) \iff \exists$ node $v \in \Delta$ labeled by $q : \Gamma(p) \doteq \Delta(v)$

Case: v is not the root



$\exists a_0 \in \mathscr{L}(p) : \mathscr{L}(p) \setminus a_0 A^{\pm *} = \mathscr{L}(q)$

$\& \; \exists \hat{q} : \hat{q} \xrightarrow{a_0^{-1}} q \in \mathcal{B}$



$\{a_0, \dots, a_n\}$

$= \{\hat{b}^{-1}, b_1, \dots, b_m\}$

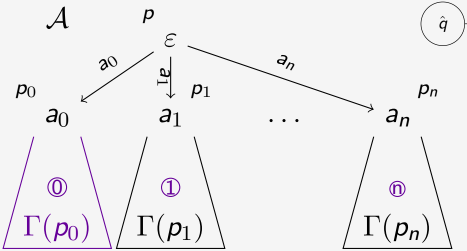
wlog.: $a_0 = \hat{b}^{-1}$

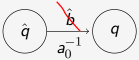
Proof Idea

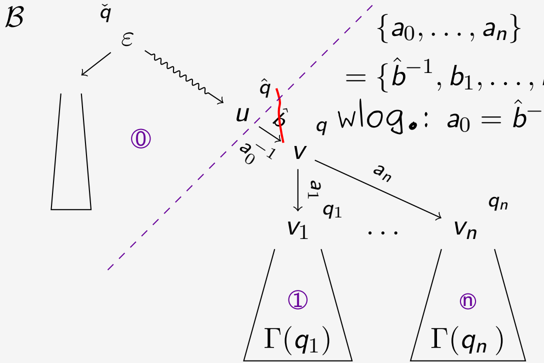
p: state of reduced pDFA $\mathcal{A}$, $\check{q}$: state of a reduced pDFA $\mathcal{B}$, $\Delta = \Gamma(\check{q})$

Predicate: $U(p,q) \iff \exists$ node $v \in \Delta$ labeled by $q : \Gamma(p) \doteqdot \Delta(v)$

Case: v is not the root







$\{a_0, \dots, a_n\}$

$= \{\hat{b}^{-1}, b_1, \dots, b_m\}$

wlog.: $a_0 = \hat{b}^{-1}$

$\exists a_0 \in \mathscr{L}(p) : \mathscr{L}(p) \setminus a_0 A^{\pm *} = \mathscr{L}(q)$

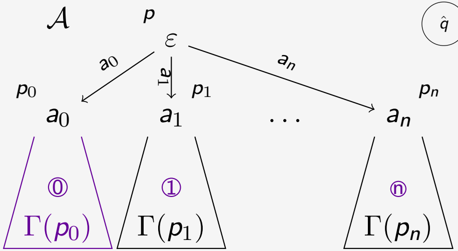
$\& \ \exists \hat{q} : \hat{q} \xrightarrow{a_0^{-1}} q \in \mathcal{B}$ and

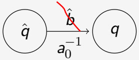
Proof Idea

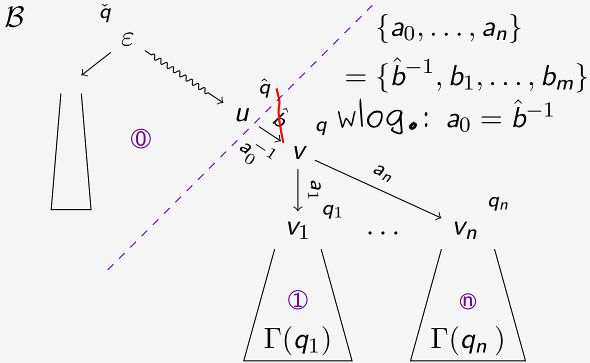
p: state of reduced pDFA $\mathcal{A}$, $\check{q}$: state of a reduced pDFA $\mathcal{B}$, $\Delta = \Gamma(\check{q})$

Predicate: $U(p, q, b_0) \iff \exists$ node $v \in \Delta$ labeled by $q : \Gamma(p) \doteq \Delta(v) \setminus v\, b_0 A^{\pm *}$

Case: v is not the root







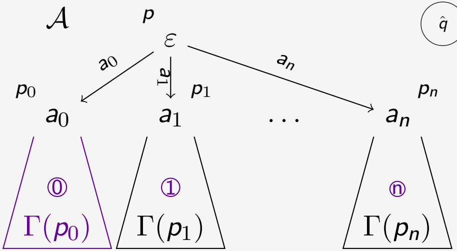
$\exists a_0 \in \mathscr{L}(p) : \mathscr{L}(p) \setminus a_0 A^{\pm *} = \mathscr{L}(q)$

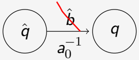
& $\exists \hat{q} : \hat{q} \xrightarrow{a_0^{-1}} q \in \mathcal{B}$ and

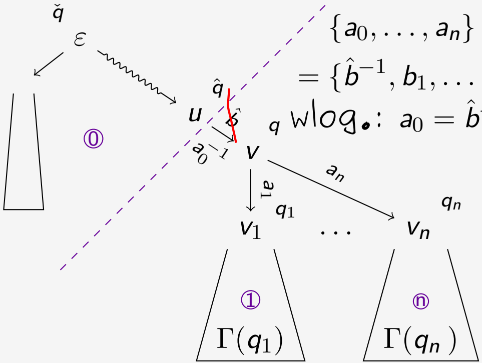
p: state of reduced pDFA $\mathcal{A}$, $\check{q}$: state of a reduced pDFA $\mathcal{B}$, $\Delta = \Gamma(\check{q})$

Predicate: $U(p, q, b_0) \iff \exists$ node $v \in \Delta$ labeled by $q : \Gamma(p) \doteq \Delta(v) \setminus v\, b_0 A^{\pm *}$

Case: v is not the root







$\{a_0, \dots, a_n\}$

$= \{\hat{b}^{-1}, b_1, \dots, b_m\}$

wlog.: $a_0 = \hat{b}^{-1}$

$\exists a_0 \in \mathscr{L}(p) : \mathscr{L}(p) \setminus a_0 A^{\pm *} = \mathscr{L}(q)$

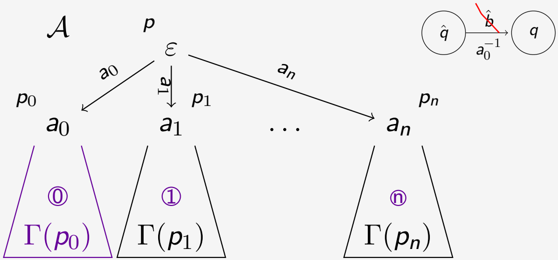
$\& \; \exists \hat{q} : \hat{q} \xrightarrow{a_0^{-1}} q \in \mathcal{B}$ and $U(\underbrace{p \cdot a_0}_{p_0}, \hat{q}, a_0^{-1})$

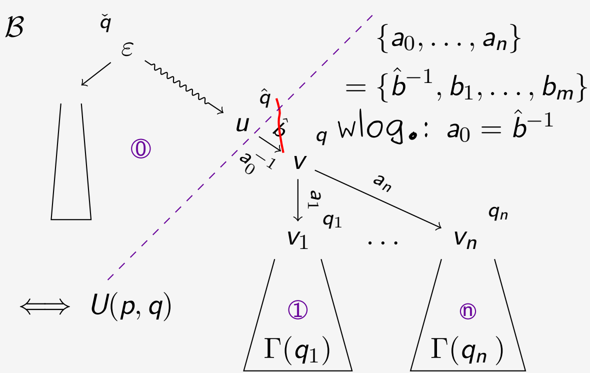
Proof Idea

p: state of reduced pDFA $\mathcal{A}$, $\check{q}$: state of a reduced pDFA $\mathcal{B}$, $\Delta = \Gamma(\check{q})$

Predicate: $U(p, q, b_0) \iff \exists$ node $v \in \Delta$ labeled by $q : \Gamma(p) \doteq \Delta(v) \setminus v\, b_0 A^{\pm *}$

Case: v is not the root





$\exists a_0 \in \mathscr{L}(p) : \mathscr{L}(p) \setminus a_0 A^{\pm *} = \mathscr{L}(q)$

$\&\ \exists \hat{q} : \hat{q} \xrightarrow{a_0^{-1}} q \in \mathcal{B}$ and $U(\underbrace{p \cdot a_0}_{p_0}, \hat{q}, a_0^{-1})$

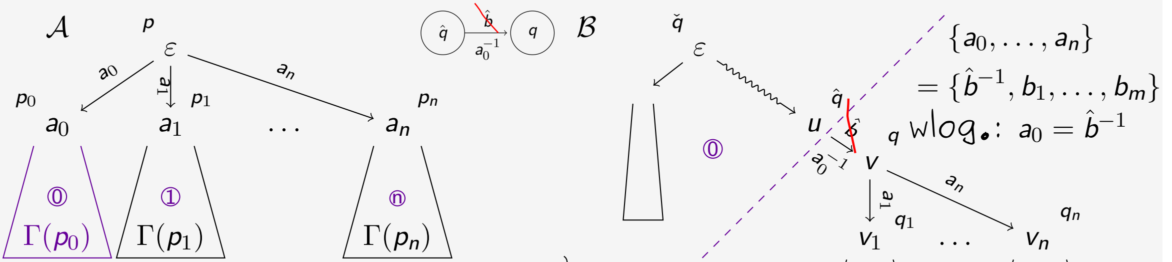
$\iff U(p, q)$

Proof Idea

p: state of reduced pDFA $\mathcal{A}$, $\check{q}$: state of a reduced pDFA $\mathcal{B}$, $\Delta = \Gamma(\check{q})$

Predicate: $U(p, q, b_0) \iff \exists$ node $v \in \Delta$ labeled by $q : \Gamma(p) \doteq \Delta(v) \backslash v\, b_0 A^{\pm *}$

Case: v is not the root



$\left.\begin{array}{l}\exists a_0 \in \mathscr{L}(p) : \mathscr{L}(p) \setminus a_0 A^{\pm *} = \mathscr{L}(q) \backslash b_0 A^{\pm *} \\ \& \ \exists \hat{q} : \hat{q} \xrightarrow{a_0^{-1}} q \in \mathcal{B} \text{ and } U(\underbrace{p \cdot a_0}_{p_0}, \hat{q}, a_0^{-1})\end{array}\right\} \iff U(p, q, b_0)$

Summary

p: state of reduced pDFA $\mathcal{A}$, $\check{q}$: state of a reduced pDFA $\mathcal{B}$, $\Delta = \Gamma(\check{q})$

Predicate: $U(p, q, b_0) \iff \exists$ node $v \in \Delta$ labeled by $q : \Gamma(p) \doteq \Delta(v) \backslash v\, b_0 A^{\pm *}$

Summary

p: state of reduced pDFA $\mathcal{A}$, $\check{q}$: state of a reduced pDFA $\mathcal{B}$, $\Delta = \Gamma(\check{q})$

Predicate: $U(p, q, b_0) \iff \exists\,$node $v \in \Delta$ labeled by $q : \Gamma(p) \doteq \Delta(v) \setminus v\, b_0 A^{\pm *}$

$$U(p, q, b_0) \iff$$

Summary

p: state of reduced pDFA $\mathcal{A}$, $\check{q}$: state of a reduced pDFA $\mathcal{B}$, $\Delta = \Gamma(\check{q})$

Predicate: $U(p, q, b_0) \iff \exists\,$node $v \in \Delta$ labeled by $q : \Gamma(p) \doteq \Delta(v) \setminus v\, b_0 A^{\pm *}$

$$U(p, q, b_0) \iff \quad q = \check{q} \;\&\; \mathscr{L}(p) = \mathscr{L}(\check{q}) \setminus b_0 A^{\pm *}$$

or

Summary

p: state of reduced pDFA $\mathcal{A}$, $\check{q}$: state of a reduced pDFA $\mathcal{B}$, $\Delta = \Gamma(\check{q})$

Predicate: $U(p, q, b_0) \iff \exists$ node $v \in \Delta$ labeled by $q : \Gamma(p) \doteq \Delta(v) \backslash v\, b_0 A^{\pm *}$

$$U(p, q, b_0) \iff q = \check{q} \,\&\, \mathscr{L}(p) = \mathscr{L}(\check{q}) \backslash b_0 A^{\pm *}$$

or $\exists a_0 \in \mathscr{L}(p)$:

Summary

p: state of reduced pDFA $\mathcal{A}$, $\check{q}$: state of a reduced pDFA $\mathcal{B}$, $\Delta = \Gamma(\check{q})$

Predicate: $U(p, q, b_0) \iff \exists$ node $v \in \Delta$ labeled by $q : \Gamma(p) \doteq \Delta(v) \backslash v\, b_0 A^{\pm *}$

$$U(p, q, b_0) \iff \quad q = \check{q}\ \&\ \mathscr{L}(p) = \mathscr{L}(\check{q}) \backslash b_0 A^{\pm *}$$

$$\text{or} \quad \exists a_0 \in \mathscr{L}(p) : \mathscr{L}(p) \setminus a_0 A^{\pm *} = \mathscr{L}(q) \backslash b_0 A^{\pm *}$$

and

Summary

p: state of reduced pDFA $\mathcal{A}$, $\check{q}$: state of a reduced pDFA $\mathcal{B}$, $\Delta = \Gamma(\check{q})$

Predicate: $U(p, q, b_0) \iff \exists\,\text{node } v \in \Delta \text{ labeled by } q : \Gamma(p) \doteq \Delta(v) \backslash v\, b_0 A^{\pm *}$

$$U(p, q, b_0) \iff \begin{array}{ll} & q = \check{q} \;\&\; \mathscr{L}(p) = \mathscr{L}(\check{q}) \backslash b_0 A^{\pm *} \\ \text{or} & \exists a_0 \in \mathscr{L}(p) : \mathscr{L}(p) \setminus a_0 A^{\pm *} = \mathscr{L}(q) \backslash b_0 A^{\pm *} \end{array}$$

and $\exists \hat{q}$:

Summary

p: state of reduced pDFA $\mathcal{A}$, $\check{q}$: state of a reduced pDFA $\mathcal{B}$, $\Delta = \Gamma(\check{q})$

Predicate: $U(p, q, b_0) \iff \exists$ node $v \in \Delta$ labeled by $q : \Gamma(p) \doteq \Delta(v) \backslash v\, b_0 A^{\pm *}$

$$U(p, q, b_0) \iff \quad q = \check{q} \;\&\; \mathscr{L}(p) = \mathscr{L}(\check{q}) \backslash b_0 A^{\pm *}$$

or $\exists a_0 \in \mathscr{L}(p) : \mathscr{L}(p) \setminus a_0 A^{\pm *} = \mathscr{L}(q) \backslash b_0 A^{\pm *}$

and $\exists \hat{q} : \hat{q} \xrightarrow{a_0^{-1}} q \in \mathcal{B} \;\&$

Summary

p: state of reduced pDFA $\mathcal{A}$, $\check{q}$: state of a reduced pDFA $\mathcal{B}$, $\Delta = \Gamma(\check{q})$

Predicate: $U(p, q, b_0) \iff \exists$ node $v \in \Delta$ labeled by $q : \Gamma(p) \doteq \Delta(v) \backslash v\, b_0 A^{\pm *}$

$$U(p, q, b_0) \iff \quad q = \check{q} \;\&\; \mathscr{L}(p) = \mathscr{L}(\check{q}) \backslash b_0 A^{\pm *}$$

or $\exists a_0 \in \mathscr{L}(p) : \mathscr{L}(p) \setminus a_0 A^{\pm *} = \mathscr{L}(q) \backslash b_0 A^{\pm *}$

and $\exists \hat{q} : \hat{q} \xrightarrow{a_0^{-1}} q \in \mathcal{B} \;\&\; U(p \cdot a_0, \hat{q}, a_0^{-1})$

Summary

p: state of reduced pDFA $\mathcal{A}$, $\check{q}$: state of a reduced pDFA $\mathcal{B}$, $\Delta = \Gamma(\check{q})$

Predicate: $U(p, q, b_0) \iff \exists$ node $v \in \Delta$ labeled by $q : \Gamma(p) \doteq \Delta(v) \setminus v\, b_0 A^{\pm *}$

$$U(p, q, b_0) \iff \quad q = \check{q} \;\&\; \mathscr{L}(p) = \mathscr{L}(\check{q}) \setminus b_0 A^{\pm *}$$

or $\exists a_0 \in \mathscr{L}(p) : \mathscr{L}(p) \setminus a_0 A^{\pm *} = \mathscr{L}(q) \setminus b_0 A^{\pm *}$

and $\exists \hat{q} : \hat{q} \xrightarrow{a_0^{-1}} q \in \mathcal{B} \;\&\; U(p \cdot a_0, \hat{q}, a_0^{-1})$

NL−Algorithm: Guess proof for $U(\check{p}, q)$ for some q nondeterministically.

Outlook

Outlook

1. Extend this to general inverse context-free graphs.

Outlook

1. Extend this to general inverse context-free graphs. PSPACE?

Outlook

1. Extend this to general inverse context-free graphs. PSPACE?
2. Compute the generating automaton from an inverse semigroup/group presentation.

Thank you!