On the Context-Freeness Problem for Vector Addition Systems

Jérôme Leroux, Vincent Penelle, Grégoire Sutre

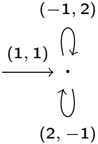
LICS, 25 juin 2013

Table of Contents

Table of Contents

VAS

$$\begin{aligned} \boldsymbol{A} &= \{(-1,2),(2,-1)\} \\ \boldsymbol{c}_{\text{init}} &= (1,1) \end{aligned}$$

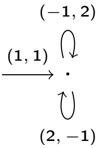


A trace is a sequence $\boldsymbol{a}_1, \ldots, \boldsymbol{a}_n$ such that:

$$\forall i \leq n, \boldsymbol{c}_{\text{init}} + \boldsymbol{a}_1 + \cdots + \boldsymbol{a}_i \geq 0$$

VAS

$$\begin{aligned} \boldsymbol{A} &= \{(-1,2),(2,-1)\} \\ \boldsymbol{c}_{\mathrm{init}} &= (1,1) \end{aligned}$$



A trace is a sequence $\boldsymbol{a}_1, \ldots, \boldsymbol{a}_n$ such that:

$$\forall i \leq n, \boldsymbol{c}_{\mathrm{init}} + \boldsymbol{a}_1 + \cdots + \boldsymbol{a}_i \geq 0$$

$(-1,2)$ $(2,-1)$ $(2,-1)$ is a trace

(1,1) (0,3) (2,2) (4,1)

$(-1,2)$ $(2,-1)$ $(-1,2)$ $(-1,2)$ $(-1,2)$ is not a trace

(1,1) (2,2) (0,6) (−1,8)

Overview of VAS

VAS $\sim$ VAS with states $\sim$ Petri nets

- Prove theorems on VAS
- Model systems with Petri nets

Many decidable problems:

- Coverability [Karp and Miller 69]
- Reachability [Mayr 81, Kosaraju 82]
- Regularity of trace languages [Ginzburg and Yoeli 80, Valk and Vidal-Naquet 81]
- Context-freeness of trace languages [Schwer 92]

Overview of VAS

VAS $\sim$ VAS with states $\sim$ Petri nets

- Prove theorems on VAS
- Model systems with Petri nets

Many decidable problems:

- Coverability [Karp and Miller 69] ExpSpace-complete
- Reachability [Mayr 81, Kosaraju 82] ExpSpace-hard & Decidable
- Regularity of trace languages [Ginzburg and Yoeli 80, Valk and Vidal-Naquet 81] ExpSpace-complete
- Context-freeness of trace languages [Schwer 92] Decidable

Overview of VAS

VAS $\sim$ VAS with states $\sim$ Petri nets

- Prove theorems on VAS
- Model systems with Petri nets

Many decidable problems:

- Coverability [Karp and Miller 69] ExpSpace-complete
- Reachability [Mayr 81, Kosaraju 82] ExpSpace-hard & Decidable
- Regularity of trace languages [Ginzburg and Yoeli 80, Valk and Vidal-Naquet 81] ExpSpace-complete
- Context-freeness of trace languages [Schwer 92] Decidable
  - intricate proof with flaws

Table of Contents

Idea

- Stack: store the value of the counters
- Positive vector $\Rightarrow$ Push it onto the stack
- Non-positive vector $\Rightarrow$ Match it with the stack
- Problem: some components may be lost

Example

$$
\begin{tikzcd}
{} \arrow[r, "{(0,0)}"] & \cdot \arrow[loop above, "{\boldsymbol{a}=(1,1)}"] \arrow[r, "{(0,0)}"] & \cdot \arrow[loop above, "{\boldsymbol{b}=(1,0)}"] \arrow[r, "{(0,0)}"] & \cdot \arrow[loop above, "{\boldsymbol{c}=(-1,-1)}"]
\end{tikzcd}
$$

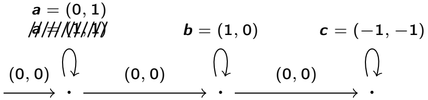
Example

$$\begin{tikzcd} {} \arrow[r, "{(0,0)}"] & \cdot \arrow[loop above, "{\boldsymbol{a}=(1,1)}"] \arrow[r, "{(0,0)}"] & \cdot \arrow[loop above, "{\boldsymbol{b}=(1,0)}"] \arrow[r, "{(0,0)}"] & \cdot \arrow[loop above, "{\boldsymbol{c}=(-1,-1)}"] \end{tikzcd}$$

It is context-free.

$$L = \{\boldsymbol{a}^n \boldsymbol{b}^m \boldsymbol{c}^p \mid n \geq p \wedge m \geq 0\}$$

Example



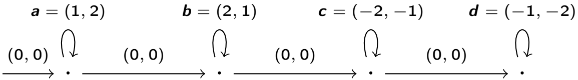
Example

$$\begin{tikzcd} {} \arrow[r, "{(0,0)}"] & \cdot \arrow[loop above, "{\substack{\boldsymbol{a} = (0,1) \\ \cancel{\boldsymbol{a} = (1,1)}}}"] \arrow[r, "{(0,0)}"] & \cdot \arrow[loop above, "{\boldsymbol{b} = (1,0)}"] \arrow[r, "{(0,0)}"] & \cdot \arrow[loop above, "{\boldsymbol{c} = (-1,-1)}"] \end{tikzcd}$$

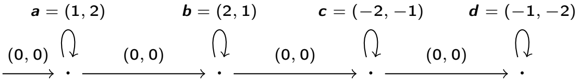
It is not context-free.

$$L \;=\; \{\boldsymbol{a}^n \boldsymbol{b}^m \boldsymbol{c}^p \mid n \geq p \quad \wedge \quad \substack{m \geq p \\ \cancel{m \geq 0}}\}$$

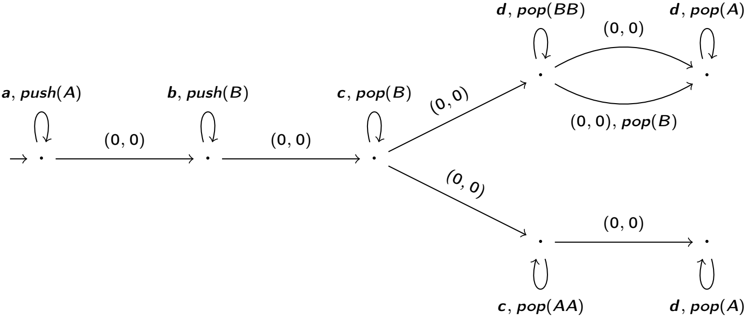
Example



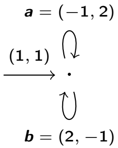
Example



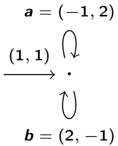
It is context-free.



Example



Example



It is not context-free.

$$L \cap (\boldsymbol{ab})^\star \boldsymbol{a}^\star \boldsymbol{b}^\star$$

$$=$$

$$\{(\boldsymbol{ab})^n \boldsymbol{a}^m \boldsymbol{b}^p \mid n + 1 \geq m \wedge n + 1 + 2m \geq p\}$$

Table of Contents

Vector Pushdown Automata

A vector pushdown automaton (VPDA) contains:

- A set of states Q
- A counter vector $\mathtt{r} \in \mathbb{Q}^d_{\geq 0}$
- A $\mathtt{stack}$ containing vectors in $\mathbb{Q}^d_{\geq 0}$
- A set of transitions T:
  - actions on $\mathtt{stack}$: push and pop
  - addition of positive vectors to $\mathtt{r}$
  - tests: boolean combinations of

$$\mathtt{r}(i) + \Delta(\mathtt{stack})(i) \# z \qquad\qquad \mathtt{r}(i) \# z$$

where $\# \in \{\leq, \geq\}$

Vector Pushdown Automata and context-freeness

Proposition

The language of a VPDA with finitely many states and transitions is context-free

Idea:
Finite set of transitions
$\Rightarrow$ Finite set of tests, finite stack alphabet
$\Rightarrow$ Above a given value of $\mathtt{r}$ and $\Delta(\mathtt{stack})$, a test is always satisfied (or not)
$\Rightarrow$ We can abstract their values and encode $\mathtt{r}$ is the states and $\Delta(\mathtt{stack})$ in $\mathtt{stack}$

Match and Remainder

$\boldsymbol{u} \geq 0$, $\boldsymbol{v} \not\geq 0$, and λ the greatest rational such that $\boldsymbol{u} + \lambda\boldsymbol{v} \geq 0$,

$$\mathrm{mat}(\boldsymbol{u}, \boldsymbol{v}) \;=\; \begin{cases} (1-\lambda)\cdot\boldsymbol{v} & \text{if } \lambda < 1 \\ (1-\frac{1}{\lambda})\cdot\boldsymbol{u} & \text{if } \lambda \geq 1 \end{cases}$$

$$\mathrm{rem}(\boldsymbol{u}, \boldsymbol{v}) \;=\; \boldsymbol{u} + \boldsymbol{v} - \mathrm{mat}(\boldsymbol{u}, \boldsymbol{v}) \geq 0$$

$$\begin{tikzcd} \arrow[r, "read(\boldsymbol{u})"] & \arrow[r, "push(\boldsymbol{u})"] & \arrow[r, dotted] & \arrow[r, "read(\boldsymbol{v})"] & \arrow[r, "\boldsymbol{p}\leftarrow\boldsymbol{v}"] & \arrow[r, "pop(\boldsymbol{u})"] & {} \arrow[ur, "\lambda\geq1"] \arrow[dr, "\lambda<1"'] & push(mat(\boldsymbol{u},\boldsymbol{p})) \\ & & & & & & & p\leftarrow mat(\boldsymbol{u},\boldsymbol{p}) \end{tikzcd}$$

Match and Remainder: Example with $\lambda < 1$

Take $\boldsymbol{u} = (1, 2, 1)$ and $\boldsymbol{v} = (-1, -3, 2)$

The greatest λ such that $\boldsymbol{u} + \lambda \boldsymbol{v} \geq 0$ is $\lambda = \dfrac{2}{3}$

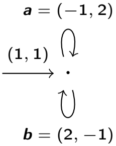
After matching $\boldsymbol{u}$ and $\boldsymbol{v}$, we are left with

$$\mathrm{mat}(\boldsymbol{u}, \boldsymbol{v}) \; = \; (1 - \lambda) \cdot \boldsymbol{v} \; = \; (-\frac{1}{3}, -1, \frac{2}{3})$$

The remainder is

$$\mathrm{rem}(\boldsymbol{u}, \boldsymbol{v}) \; = \; \boldsymbol{u} + \lambda \boldsymbol{v} \; = \; (\frac{1}{3}, 0, \frac{7}{3})$$

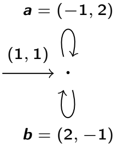
From VAS to VPDA: Example



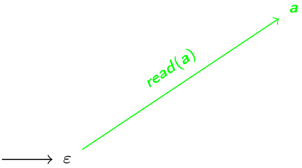
ε
$(0, 0)$
$[]$

$\longrightarrow \varepsilon$

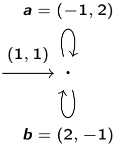
From VAS to VPDA: Example



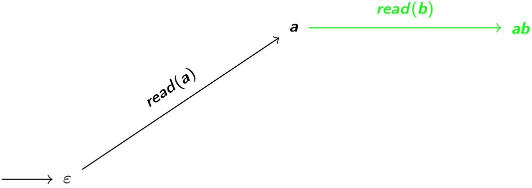
$$\begin{matrix}\varepsilon\\(0,0)\\[]\end{matrix} \xrightarrow{\textbf{\textit{read}}(\boldsymbol{a})} \begin{matrix}\boldsymbol{a}\\(0,0)\\[]\end{matrix}$$



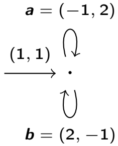
From VAS to VPDA: Example



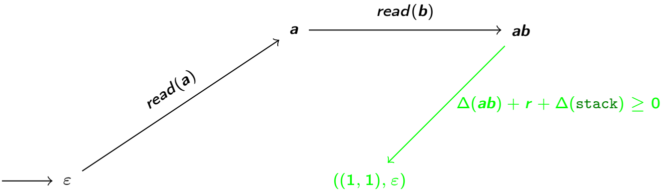
$$\begin{matrix}\varepsilon \\ (0,0) \\ [\,]\end{matrix} \xrightarrow{\textit{read}(a)} \begin{matrix}a \\ (0,0) \\ [\,]\end{matrix} \xrightarrow{\textit{read}(b)} \begin{matrix}ab \\ (0,0) \\ [\,]\end{matrix}$$



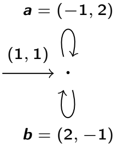
From VAS to VPDA: Example



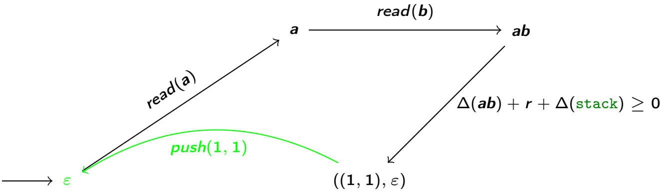
$$\begin{array}{c} \varepsilon \\ (0,0) \\ [] \end{array} \xrightarrow{read(a)} \begin{array}{c} a \\ (0,0) \\ [] \end{array} \xrightarrow{read(b)} \begin{array}{c} ab \\ (0,0) \\ [] \end{array}$$



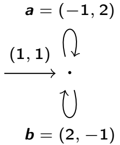
From VAS to VPDA: Example

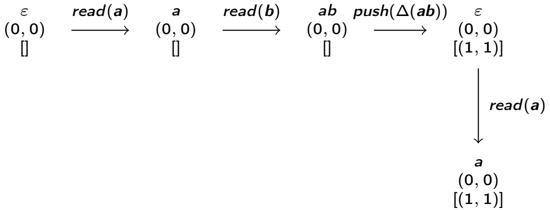


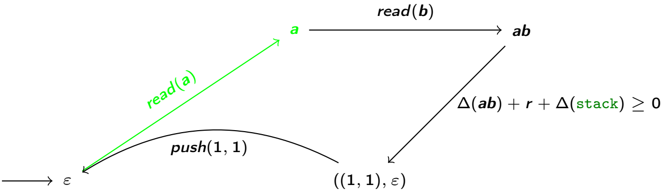
$$
\begin{array}{ccccccc}
\varepsilon & & a & & ab & & \varepsilon \\
(0,0) & \xrightarrow{read(a)} & (0,0) & \xrightarrow{read(b)} & (0,0) & \xrightarrow{push(\Delta(ab))} & (0,0) \\
[] & & [] & & [] & & [(1,1)]
\end{array}
$$



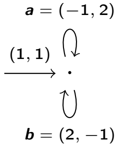
From VAS to VPDA: Example

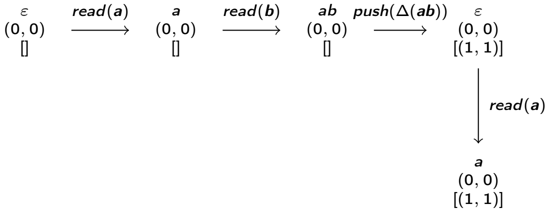


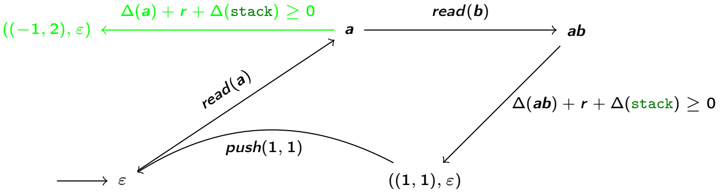




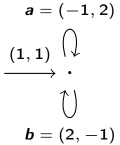
From VAS to VPDA: Example

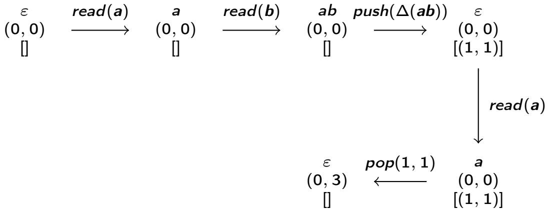


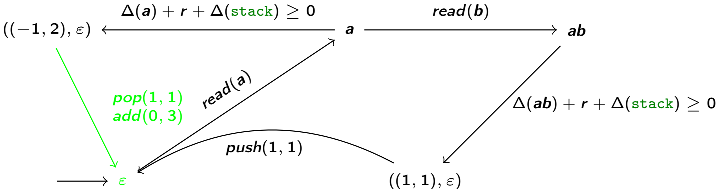




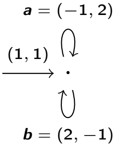
From VAS to VPDA: Example

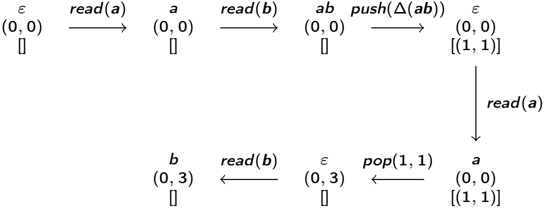


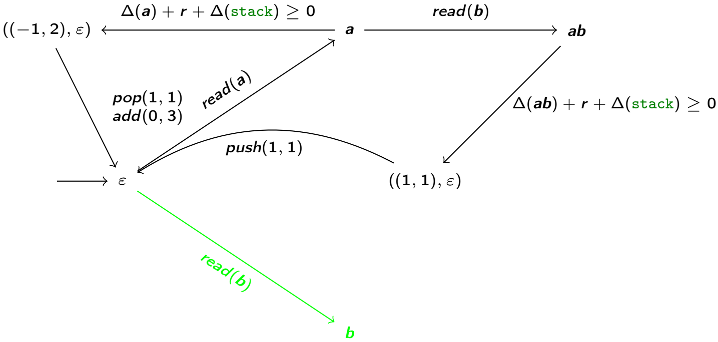




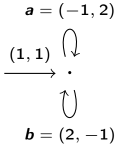
From VAS to VPDA: Example

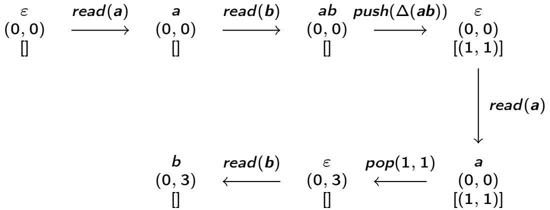


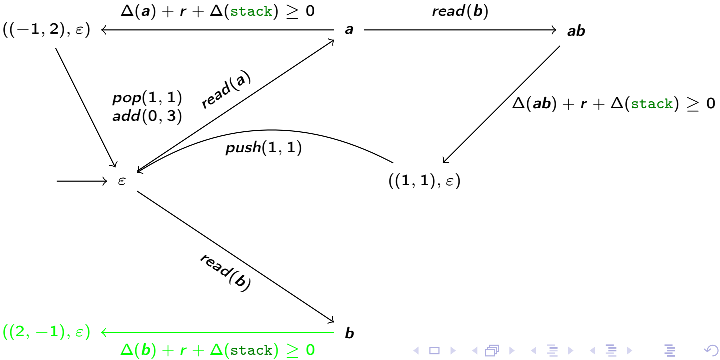




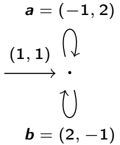
From VAS to VPDA: Example

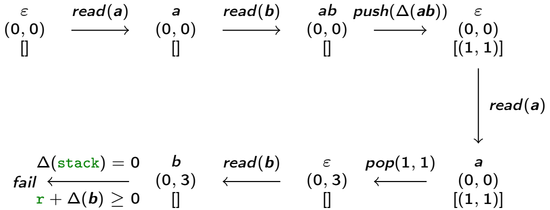


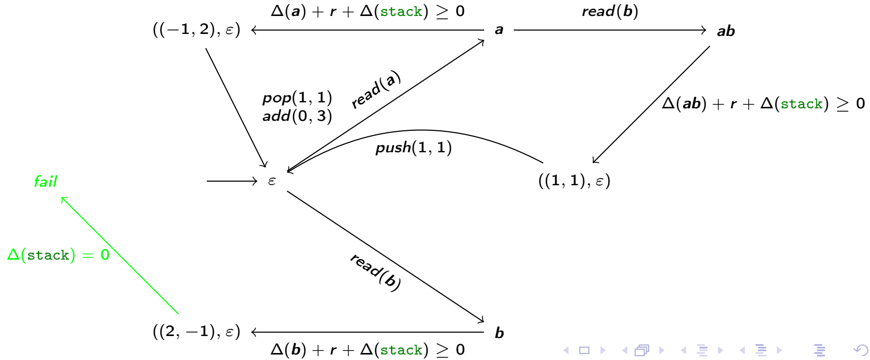




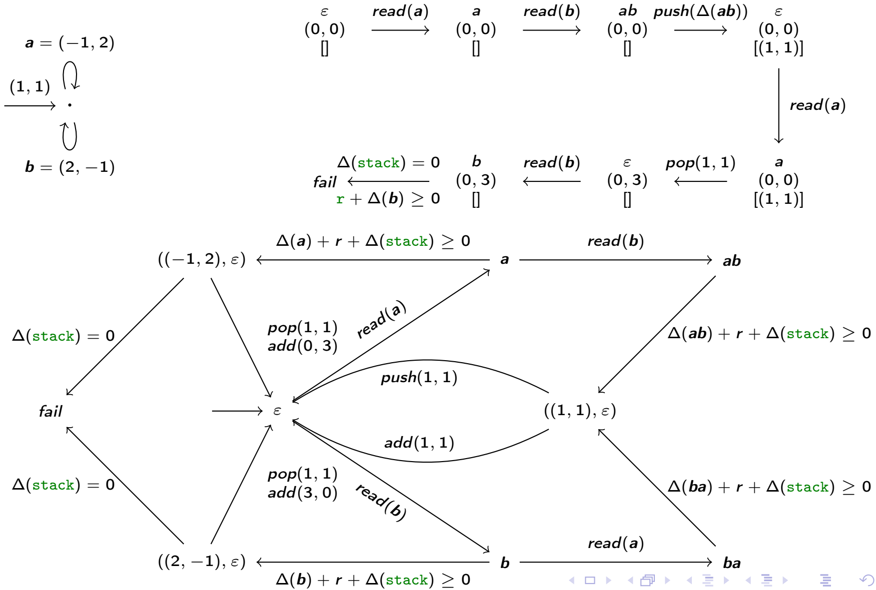
From VAS to VPDA: Example







From VAS to VPDA: Example



From VAS to VPDA: Properties

Proposition

The constructed VPDA has a finite set of states and transitions

Proposition

If fail is not reachable, the VPDA recognises the same language as the VAS

Corollary

If fail is not reachable, the VAS is context-free

Table of Contents

Witnesses of Non-Context-Freeness

$(2, 0) \quad (2, 2) \quad (0, 1) \quad (-2, -1) \quad (-1, -1)$

Definition

A witness of non-context-freeness is a tuple $(\boldsymbol{\sigma}_1, \ldots, \boldsymbol{\sigma}_k, U)$ such that:

- $\boldsymbol{\sigma}_1 \cdots \boldsymbol{\sigma}_k$ is a trace

Witnesses of Non-Context-Freeness

$(2,0)$ $(2,2)$ $(0,1)$ $(-2,-1)$ $(-1,-1)$

Definition

A witness of non-context-freeness is a tuple $(\boldsymbol{\sigma}_1, \ldots, \boldsymbol{\sigma}_k, U)$ such that:

- $\boldsymbol{\sigma}_1 \cdots \boldsymbol{\sigma}_k$ is a trace
- $\forall (s,t) \in U$, $s < t$, $\boldsymbol{\sigma}_s \geq 0$, $\boldsymbol{\sigma}_t \not\geq 0$

Witnesses of Non-Context-Freeness

$(2,0) \qquad (2,2) \qquad (0,1) \qquad (-2,-1) \qquad (-1,-1)$

Definition

A witness of non-context-freeness is a tuple $(\boldsymbol{\sigma}_1, \ldots, \boldsymbol{\sigma}_k, U)$ such that:

- $\boldsymbol{\sigma}_1 \cdots \boldsymbol{\sigma}_k$ is a trace
- $\forall (s,t) \in U$, $s < t$, $\boldsymbol{\sigma}_s \geq 0$, $\boldsymbol{\sigma}_t \not\geq 0$

Witnesses of Non-Context-Freeness

$(2,0) \quad (2,2) \quad (0,1) \quad (-2,-1) \quad (-1,-1)$

Definition

A witness of non-context-freeness is a tuple $(\boldsymbol{\sigma}_1, \ldots, \boldsymbol{\sigma}_k, U)$ such that:

- $\boldsymbol{\sigma}_1 \cdots \boldsymbol{\sigma}_k$ is a trace
- $\forall (s,t) \in U,\ s < t,\ \boldsymbol{\sigma}_s \geq 0,\ \boldsymbol{\sigma}_t \not\geq 0$
- $\forall (s,t),(u,v) \in U,\ \neg(s < u < t < v)$

Witnesses of Non-Context-Freeness

$(2, 0) \qquad (2, 2) \qquad (0, 1) \qquad (-2, -1) \qquad (-1, -1)$

Definition

A witness of non-context-freeness is a tuple $(\boldsymbol{\sigma}_1, \ldots, \boldsymbol{\sigma}_k, U)$ such that:

- $\boldsymbol{\sigma}_1 \cdots \boldsymbol{\sigma}_k$ is a trace
- $\forall (s, t) \in U,\ s < t,\ \boldsymbol{\sigma}_s \geq 0,\ \boldsymbol{\sigma}_t \ngeq 0$
- $\forall (s, t), (u, v) \in U,\ \neg(s < u < t < v)$

Witnesses of Non-Context-Freeness

$(2,0) \quad (2,2) \quad (0,1) \quad (-2,-1) \quad (-1,-1)$

Definition

A witness of non-context-freeness is a tuple $(\boldsymbol{\sigma}_1, \ldots, \boldsymbol{\sigma}_k, U)$ such that:

- $\boldsymbol{\sigma}_1 \cdots \boldsymbol{\sigma}_k$ is a trace
- $\forall (s,t) \in U,\ s < t,\ \boldsymbol{\sigma}_s \geq 0,\ \boldsymbol{\sigma}_t \not\geq 0$
- $\forall (s,t), (u,v) \in U,\ \neg(s < u < t < v)$
- $\forall (s,t) \in U$, with $t < k$,
  $\exists s' \leq s, (s',t) \in U \wedge \forall i, \boldsymbol{\sigma}_t(i) < 0 \rightarrow \boldsymbol{\sigma}_{s'}(i) > 0$

Witnesses of Non-Context-Freeness

$(2,0) \qquad (2,2) \qquad (0,1) \qquad (-2,-1) \qquad (-1,-1)$

Definition

A witness of non-context-freeness is a tuple $(\boldsymbol{\sigma}_1, \ldots, \boldsymbol{\sigma}_k, U)$ such that:

- $\boldsymbol{\sigma}_1 \cdots \boldsymbol{\sigma}_k$ is a trace
- $\forall (s,t) \in U,\ s < t,\ \boldsymbol{\sigma}_s \geq 0,\ \boldsymbol{\sigma}_t \not\geq 0$
- $\forall (s,t),(u,v) \in U,\ \neg(s < u < t < v)$
- $\forall (s,t) \in U$, with $t < k$,
  $\exists s' \leq s, (s',t) \in U \land \forall i, \boldsymbol{\sigma}_t(i) < 0 \rightarrow \boldsymbol{\sigma}_{s'}(i) > 0$

$$(2,0) \qquad (2,2) \qquad (0,1) \qquad (-2,-1) \qquad (-1,-1)$$

Definition

A witness of non-context-freeness is a tuple $(\boldsymbol{\sigma}_1, \ldots, \boldsymbol{\sigma}_k, U)$ such that:

- $\boldsymbol{\sigma}_1 \cdots \boldsymbol{\sigma}_k$ is a trace
- $\forall (s,t) \in U,\ s < t,\ \boldsymbol{\sigma}_s \geq 0,\ \boldsymbol{\sigma}_t \not\geq 0$
- $\forall (s,t),(u,v) \in U,\ \neg(s < u < t < v)$
- $\forall (s,t) \in U$, with $t < k$,
  $\exists s' \leq s, (s',t) \in U \wedge \forall i, \boldsymbol{\sigma}_t(i) < 0 \rightarrow \boldsymbol{\sigma}_{s'}(i) > 0$
- $\forall i, \boldsymbol{\sigma}_k(i) > 0 \rightarrow \exists (s,t) \in U, rem(\boldsymbol{\sigma}_s, \boldsymbol{\sigma}_t)(i) > 0$

Witnesses of Non-Context-Freeness

$(2, 0) \qquad (2, 2) \qquad (0, 1) \qquad (-2, -1) \qquad (-1, -1)$

Definition

A witness of non-context-freeness is a tuple $(\boldsymbol{\sigma}_1, \ldots, \boldsymbol{\sigma}_k, U)$ such that:

- $\boldsymbol{\sigma}_1 \cdots \boldsymbol{\sigma}_k$ is a trace
- $\forall (s, t) \in U,\ s < t,\ \boldsymbol{\sigma}_s \geq 0,\ \boldsymbol{\sigma}_t \not\geq 0$
- $\forall (s, t), (u, v) \in U,\ \neg(s < u < t < v)$
- $\forall (s, t) \in U$, with $t < k$,
  $\exists s' \leq s, (s', t) \in U \wedge \forall i, \boldsymbol{\sigma}_t(i) < 0 \rightarrow \boldsymbol{\sigma}_{s'}(i) > 0$
- $\forall i, \boldsymbol{\sigma}_k(i) > 0 \rightarrow \exists (s, t) \in U, \mathit{rem}(\boldsymbol{\sigma}_s, \boldsymbol{\sigma}_t)(i) > 0$

Witnesses and Vector Pushdown Automata

Proposition

If a VAS admits a witness of non-context-freeness, it is not context free

Proof based on Ginsburg characterisation of bounded context-free languages

Witnesses and Vector Pushdown Automata

Proposition

If a VAS admits a witness of non-context-freeness, it is not context free

Proof based on Ginsburg characterisation of bounded context-free languages

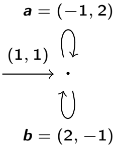
Proposition

If fail is reachable in the vector pushdown automaton of a VAS, it admits a witness of non-context-freeness

Corollary

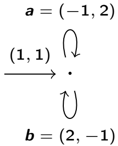
If fail is reachable in the vector pushdown automaton of a VAS, it is not context-free

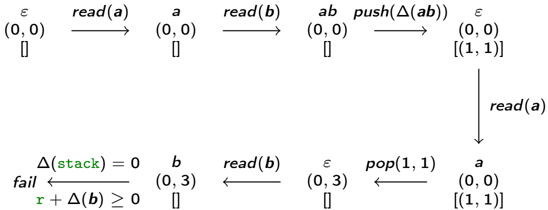
Witnesses and Vector Pushdown Automata: Example

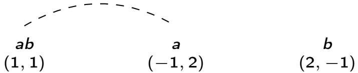


$$
\begin{tikzcd}[column sep=large]
\begin{matrix} \varepsilon \\ (0,0) \\ [] \end{matrix} \arrow[r, "read(a)"] & \begin{matrix} a \\ (0,0) \\ [] \end{matrix} \arrow[r, "read(b)"] & \begin{matrix} ab \\ (0,0) \\ [] \end{matrix} \arrow[r, "push(\Delta(ab))"] & \begin{matrix} \varepsilon \\ (0,0) \\ [(1,1)] \end{matrix} \arrow[d, "read(a)"] \\
fail & \begin{matrix} b \\ (0,3) \\ [] \end{matrix} \arrow[l, "\Delta(\texttt{stack}) = 0"', "\texttt{r} + \Delta(b) \geq 0"] & \begin{matrix} \varepsilon \\ (0,3) \\ [] \end{matrix} \arrow[l, "read(b)"'] & \begin{matrix} a \\ (0,0) \\ [(1,1)] \end{matrix} \arrow[l, "pop(1,1)"']
\end{tikzcd}
$$

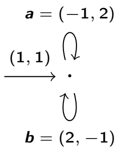
Witnesses and Vector Pushdown Automata: Example

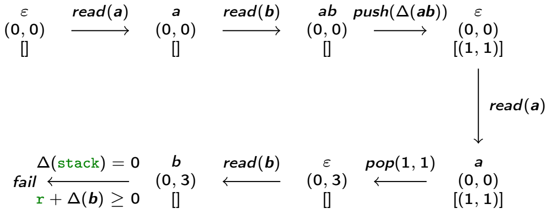


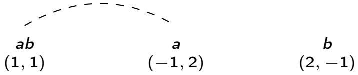




Witnesses and Vector Pushdown Automata: Example







$rem(ab, b) = (0, 3)$
$\Delta(b) = (2, -1)$
All negative components of b are lost.

It is a witness of non-context-freeness.

Conclusion

- New proof, simpler than the one of Schwer

Conclusion

- New proof, simpler than the one of Schwer
- Simpler characterisation of context-freeness:

Theorem 5.1. *Let $A=(T,\varphi,\boldsymbol{a})$ be a k-VAS and $L(A)$ be its associated language. Let $\mathscr{C}(A)=\langle T,Q,\delta,\boldsymbol{a}\rangle$ be its covering automaton. $L(A)$ is context-free if and only if the following conditions hold:*

(P1) *For every iterating system $\mathscr{I}=(\alpha_0,q_1,u_1,\alpha_1,q_2,\ldots,\alpha_{p-1},q_p,u_p,\alpha_p,q)$ related to an elementary loop (q,u), $\mathbb{C}1(\mathscr{I},(q,u))$ is satisfied.*

(P2) *For no (q,v), (q',v') in ε^-, (p,u), (p',u') in ε^+ $\mathbb{C}2((p,u),(p',u'),(q,v),(q',v'))$ is satisfied.*

(P3) *For all classes $[i]$ of states $\mathbb{C}3(label(\mathscr{W}_i))$ and $\mathbb{C}3(label(\mathscr{X}_i))$ are satisfied.*

(P4) *For all $j\in[1,m]$, the set $W=W_{I^*(j)}$ satisfies $\mathbb{C}4(W)$.*

(P5) *For all $j\in[1,m]$, if $W_{I^*(j)}$ has a dominating coordinate, then there is one in $ZERO(J)$.*

The language of traces of a VAS is context-free if and only if it has a context-free intersection with every bounded regular language

Definition 1.5. Let $A=(T,\varphi,\boldsymbol{a})$ be a k-VAS and $\mathscr{C}(A)=\langle T,Q,\delta,\boldsymbol{a}\rangle$ be its covering automaton. Let (q,w) be a loop, and let $\mathscr{I}=(\alpha_0,q_1,u_1,\alpha_1,q_2,\ldots,\alpha_{p-1},q_p,u_p,\alpha_p,q)$ be an iterating system of length p related to (q,w). We note $\mathbb{C}1(\mathscr{I},(q,w))$, the condition: the iterating system $\mathscr{I}$ can be transformed into an elementary minimal iterating system, of length at most 1, by operations of decomposition or reduction.

Corollary 1.6. *Let $A=(T,\varphi,\boldsymbol{a})$ be a k-VAS and $\mathscr{C}(A)=\langle T,Q,\delta,\boldsymbol{a}\rangle$ be its covering automaton. If $L(A)$ is context-free, then the following condition holds:*

(P1) *For every iterating system $\mathscr{I}=(\alpha_0,q_1,u_1,\alpha_1,q_2,\ldots,\alpha_{p-1},q_p,u_p,\alpha_p,q)$ related to an elementary loop (q,u), $\mathbb{C}1(\mathscr{I},(q,u))$ holds.*

Definition 4.8. Let $\boldsymbol{u}_1,\boldsymbol{u}_2,\ldots,\boldsymbol{u}_t$ be t k-tuples of $\mathbf{Z}^k$. We denote by $\mathbb{C}3(\boldsymbol{u}_1,\boldsymbol{u}_2,\ldots,\boldsymbol{u}_t)$, the condition: there exists a coordinate p which is a dominating coordinate with respect to $\{\boldsymbol{u}_1,\boldsymbol{u}_2,\ldots,\boldsymbol{u}_t\}$, and by $\mathbb{C}3(\boldsymbol{u}_1,\boldsymbol{u}_2,\ldots,\boldsymbol{u}_t;p,q)$, the condition: the coordinate p dominates the coordinate q with respect to $\{\boldsymbol{u}_1,\boldsymbol{u}_2,\ldots,\boldsymbol{u}_t\}$.

Definition 2.1. We note $\mathbb{C}2((p,u),(p',u'),(q,v),(q',v'))$, the condition: (q,v) and (q',v') are two loops in ε^-, and (p,u) and (p',u') are two loops in ε^+, such that there exist $\alpha_0,\alpha_1,\alpha_2$ and α_3 with $\delta(q_0,\alpha_0)=p$, $\delta(p,\alpha_1)=p'$, $\delta(p',\alpha_2)=q$ and $\delta(q,\alpha_3)=q'$ and such that $\|v\|^-\subseteq\|u\|^+$ and $\|v'\|^-\subseteq\|u'\|^+$, and such that either $\|v\|^-$ is not included in $\|u'\|^+$, or $\|v'\|^-$ is not included in $\|u\|^+$ (or both).

We can state our second necessary condition.

Proposition 2.2. *Let $A=(T,\varphi,\boldsymbol{a})$ be a k-VAS and $\mathscr{C}(A)=\langle T,Q,\delta,\boldsymbol{a}\rangle$ be its covering automaton such that* (P1) *holds. If $L(A)$ is context-free, than the following condition holds:*

(P2) *For no $(q,v),(q',v')$ in ε^-, $(p,u),(p',u')$ in ε^+, $\mathbb{C}2((p,u),(p'u'),(q,v),(q',v'))$ holds.*

Definition 4.1. For any set W of k-tuples, denote Δ_w the set: $\Delta_w=\{x\in H\mid\exists r,s\ \varphi_r$ and φ_s are not colinear and $x\in K_r\cap K_s\}$. We note $\mathbb{C}4(W)$, the condition: the set Δ_w is a stratified set.

Proposition 4.5. *Let $A=(T,\varphi,\boldsymbol{a})$ be a k-VAS and $\mathscr{C}(A)=\langle T,Q,\delta,\boldsymbol{a}\rangle$ be its covering automaton such that* (P1), (P2), (P3) *and* (P4) *hold. If $L(A)$ is context-free, then the following condition holds:*

(P5) *For all $j\in[1,m]$, if $W_{I^*(j)}$ has a dominating coordinate, then there is one in $ZERO(J)$.*

Conclusion

- New proof, simpler than the one of Schwer
- Simpler characterisation of context-freeness:

Theorem 5.1. *Let* $A=(T,\varphi,\boldsymbol{a})$ *be a k-VAS and* $L(A)$ *be its associated language. Let* $\mathscr{C}(A)=\langle T,Q,\delta,\boldsymbol{a}\rangle$ *be its covering automaton.* $L(A)$ *is context-free if and only if the following conditions hold:*

(P1) *For every iterating system* $\mathscr{I}=(\alpha_0,q_1,u_1,\alpha_1,q_2,\ldots,\alpha_{p-1},q_p,u_p,\alpha_p,q)$ *related to an elementary loop* (q,u), $\mathbb{C}1(\mathscr{I},(q,u))$ *is satisfied.*

(P2) *For no* (q,v), (q',v') *in* ε^-, (p,u), (p',u') *in* ε^+ $\mathbb{C}2((p,u),(p',u'),(q,v),(q',v'))$ *is satisfied.*

(P3) *For all classes* $[i]$ *of states* $\mathbb{C}3(label(\mathscr{W}_i))$ *and* $\mathbb{C}3(label(\mathscr{X}_i))$ *are satisfied.*

(P4) *For all* $j\in[1,m]$, *the set* $W=W_{I^*(j)}$ *satisfies* $\mathbb{C}4(W)$.

(P5) *For all* $j\in[1,m]$, *if* $W_{I^*(j)}$ *has a dominating coordinate, then there is one in* $ZERO(J)$.

The language of traces of a VAS is context-free if and only if it has a context-free intersection with every bounded regular language

Perspectives:

- The complexity is shown to be ExpSpace by J. Leroux, M. Praveen, and G. Sutre in an upcoming paper to CONCUR'13