

Hyper-Ackermannian Bounds for Pushdown Vector Addition Systems

Jérôme Leroux¹ M. Praveen² Grégoire Sutre¹

¹Univ. Bordeaux & CNRS, LaBRI, Talence, France

²Chennai Mathematical Institute, Chennai, India

ACTS, Chennai, India, Feb 2015

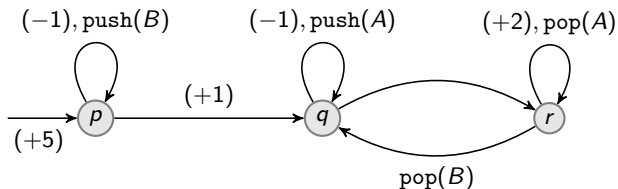
Table of Contents

- 1 Pushdown Vector Addition Systems
- 2 Reduced Reachability Tree for Pushdown VAS
- 3 Worst-Case Size of the Reduced Reachability Tree
- 4 Conclusion

Table of Contents

- 1 Pushdown Vector Addition Systems
- 2 Reduced Reachability Tree for Pushdown VAS
- 3 Worst-Case Size of the Reduced Reachability Tree
- 4 Conclusion

Pushdown Vector Addition Systems — Model



VAS

d (implicit) counters over \mathbb{N}
counter actions

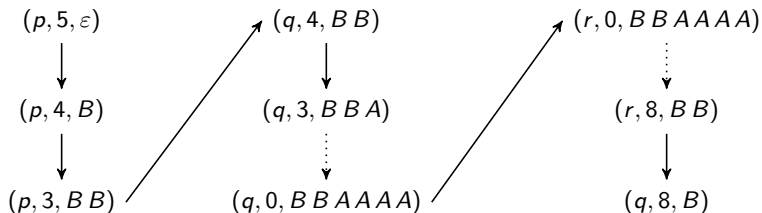
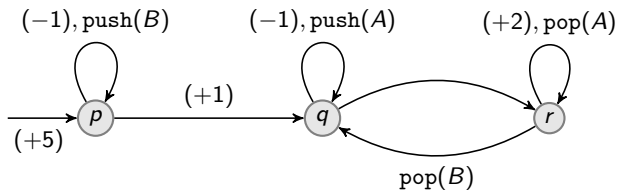
- syntax: $\mathbf{a} \in \mathbb{Z}^d$
- semantics: $\underbrace{\mathbf{v}}_{\in \mathbb{N}^d} \rightarrow \underbrace{\mathbf{v} + \mathbf{a}}_{\in \mathbb{N}^d}$

+

Stack

finite stack alphabet
push and pop

Pushdown Vector Addition Systems — Model

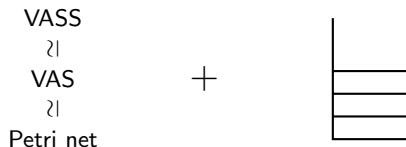


Pushdown Vector Addition Systems — Motivations



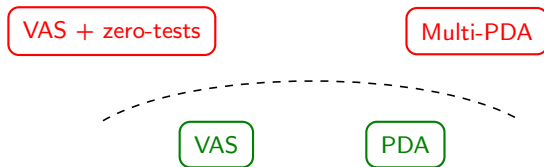
- ⇒ Richer model for the verification of concurrent systems
- Multi-threaded recursive programs
 - One recursive server + unboundedly many finite-state clients

Pushdown Vector Addition Systems — Motivations



- ⇒ Richer model for the verification of concurrent systems
- Multi-threaded recursive programs
 - One recursive server + unboundedly many finite-state clients

⇒ Is the model too powerful?



Brief State of the Art

	Boundedness	Coverability	Reachability
VAS	$\text{EXPSPACE-c}^{1,2}$	$\text{EXPSPACE-c}^{1,2}$	Decidable ^{3,4}
+ full counter	Decidable ⁵	Decidable	Decidable ⁶
+ stack		TOWER-h^7	

[1] Lipton 1976

[2] Rackoff 1978

[3] Mayr 1981

[5] Finkel, Sangnier 2010

[7] Lazić 2012

[4] Kosaraju 1982

[6] Reinhardt 2008

Brief State of the Art

	Boundedness	Coverability	Reachability
VAS	$\text{EXPSPACE-c}^{1,2}$	$\text{EXPSPACE-c}^{1,2}$	Decidable ^{3,4}
+ full counter	Decidable ⁵	Decidable	Decidable ⁶
+ stack		TOWER-h ⁷	

[1] Lipton 1976

[2] Rackoff 1978

[3] Mayr 1981

[5] Finkel, Sangnier 2010

[7] Lazić 2012

[4] Kosaraju 1982

[6] Reinhardt 2008

Subclasses of pushdown VAS with decidable reachability

- Multiset pushdown systems [Sen, Viswanathan 2006]
- $\text{VAS} \cap \text{CFL}$ of finite index [Atig, Ganty 2011]

Brief State of the Art

	Boundedness	Coverability	Reachability
VAS	$\text{EXPSPACE-c}^{1,2}$	$\text{EXPSPACE-c}^{1,2}$	Decidable ^{3,4}
+ full counter	Decidable ⁵	Decidable	Decidable ⁶
+ stack	?	TOWER-h ⁷	

[1] Lipton 1976

[2] Rackoff 1978

[3] Mayr 1981

[5] Finkel, Sangnier 2010

[7] Lazić 2012

[4] Kosaraju 1982

[6] Reinhardt 2008

Subclasses of pushdown VAS with decidable reachability

- Multiset pushdown systems [Sen, Viswanathan 2006]
- $\text{VAS} \cap \text{CFL}$ of finite index [Atig, Ganty 2011]

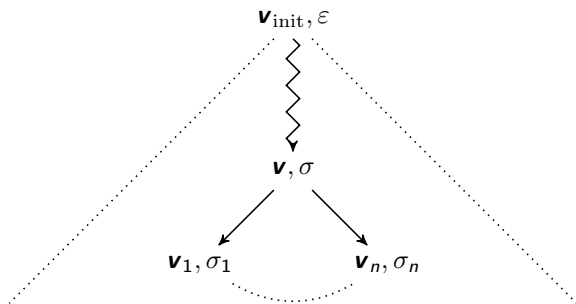
Our Contribution

- ⇒ Boundedness is decidable for pushdown VAS
 - Reduced reachability tree: adaptation of the VAS case
- ⇒ Worst-case complexity of the algorithm: hyper-Ackermannian
 - Bound the length of bad **nested** sequences over (\mathbb{N}^d, \leq)
 - Weak computation of an hyper-Ackermannian function
 - Inspired from recent results on bad sequences for various wqo's
 - ▶ [Figueira, Figueira, Schmitz, Schnoebelen 2011]
 - ▶ [Schmitz, Schnoebelen 2011]
 - ▶ ...

Table of Contents

- 1 Pushdown Vector Addition Systems
- 2 Reduced Reachability Tree for Pushdown VAS**
- 3 Worst-Case Size of the Reduced Reachability Tree
- 4 Conclusion

Reachability Tree of a Pushdown VAS

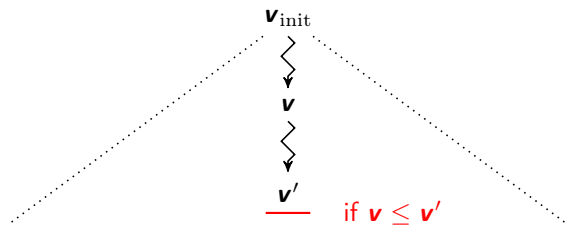


⇒ Exhaustive and enumerative forward exploration from $(\mathbf{v}_{\text{init}}, \varepsilon)$

⇒ Potentially **infinite**, need to **truncate**

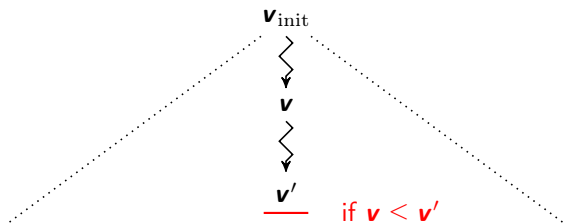
Reduced Reachability Tree for VAS [Karp, Miller 1969]

Truncation rule:



Reduced Reachability Tree for VAS [Karp, Miller 1969]

Truncation rule:



For every VAS, \leq and $<$ are simulation relations

Truncation entails that

- $v_{\text{init}} \xrightarrow{*} v \xrightarrow{*} v' \xrightarrow{*} v'' \xrightarrow{*} v''' \dots$
- If $v < v'$ then $v < v' < v'' < v''' < \dots$

Theorem ([Karp, Miller 1969])

The reduced reachability tree of a VAS \mathcal{A} is finite.

Proof. König's Lemma + Dickson's Lemma □

Theorem ([Karp, Miller 1969])

A VAS \mathcal{A} is unbounded if, and only if, its reduced reachability tree contains a leaf that is strictly larger than one of its ancestors.

RRT-based Algorithm for VAS Boundedness

Theorem ([Karp, Miller 1969])

The reduced reachability tree of a VAS \mathcal{A} is finite.

Proof. König's Lemma + Dickson's Lemma □

Theorem ([Karp, Miller 1969])

A VAS \mathcal{A} is unbounded if, and only if, its reduced reachability tree contains a leaf that is strictly larger than one of its ancestors.

Theorem ([McAloon 1984], [Figueira et al. 2011])

The size of the reduced reachability tree of a VAS \mathcal{A} is at most

- *primitive-recursive in $|\mathcal{A}|$ when the dimension d is fixed,*
- *Ackermannian in $|\mathcal{A}|$ when the dimension is part of the input.*

Tentative Simulation-Based Truncation for Pushdown VAS

Consider a run

$$(\mathbf{v}_{\text{init}}, \varepsilon) \xrightarrow{*} (\mathbf{v}, \sigma) \xrightarrow{*} (\mathbf{v}', \sigma')$$

such that

$$\mathbf{v} \leq \mathbf{v}' \quad \text{and} \quad \sigma \leq_{\text{suffix}} \sigma'$$

Then $(\mathbf{v}_{\text{init}}, \varepsilon) \xrightarrow{*} (\mathbf{v}, \sigma) \xrightarrow{*} (\mathbf{v}', \sigma') \xrightarrow{*} (\mathbf{v}'', \sigma'') \xrightarrow{*} (\mathbf{v}''', \sigma''') \dots$

Tentative Simulation-Based Truncation for Pushdown VAS

Consider a run

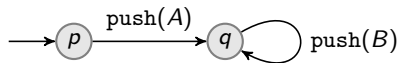
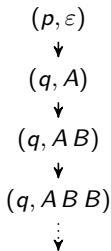
$$(\mathbf{v}_{\text{init}}, \varepsilon) \xrightarrow{*} (\mathbf{v}, \sigma) \xrightarrow{*} (\mathbf{v}', \sigma')$$

such that

$$\mathbf{v} \leq \mathbf{v}' \quad \text{and} \quad \sigma \leq_{\text{suffix}} \sigma'$$

Then $(\mathbf{v}_{\text{init}}, \varepsilon) \xrightarrow{*} (\mathbf{v}, \sigma) \xrightarrow{*} (\mathbf{v}', \sigma') \xrightarrow{*} (\mathbf{v}'', \sigma'') \xrightarrow{*} (\mathbf{v}''', \sigma''') \dots$

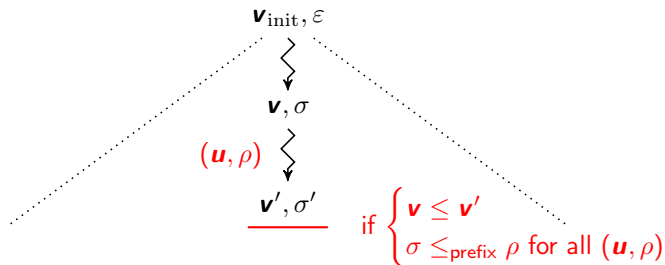
But:



No truncation, infinite branch!

Reduced Reachability Tree for Pushdown VAS

Truncation rule:



Truncation entails that

- $(\mathbf{v}_{\text{init}}, \varepsilon) \xrightarrow{*} (\mathbf{v}, \sigma) \xrightarrow{*} (\mathbf{v}', \sigma') \xrightarrow{*} (\mathbf{v}'', \sigma'') \xrightarrow{*} (\mathbf{v}''', \sigma''') \dots$
- If $\mathbf{v} < \mathbf{v}'$ then $\mathbf{v} < \mathbf{v}' < \mathbf{v}'' < \mathbf{v}''' < \dots$
- If $\sigma <_{\text{prefix}} \sigma'$ then $\sigma <_{\text{prefix}} \sigma' <_{\text{prefix}} \sigma'' <_{\text{prefix}} \sigma''' <_{\text{prefix}} \dots$

Finiteness of the Reduced Reachability Tree

Theorem

The reduced reachability tree of a pushdown VAS \mathcal{A} is finite.

Proof. By contradiction, assume that it is infinite.

The tree is finitely branching. So, by König's Lemma, there is an infinite branch

$$(\mathbf{v}_{\text{init}}, \varepsilon) \rightarrow (\mathbf{v}_1, \sigma_1) \rightarrow (\mathbf{v}_2, \sigma_2) \cdots$$

•

Finiteness of the Reduced Reachability Tree

Theorem

The reduced reachability tree of a pushdown VAS \mathcal{A} is finite.

Proof. By contradiction, assume that it is infinite.

The tree is finitely branching. So, by König's Lemma, there is an infinite branch

$$(\mathbf{v}_{\text{init}}, \varepsilon) \rightarrow (\mathbf{v}_1, \sigma_1) \rightarrow (\mathbf{v}_2, \sigma_2) \cdots$$

$$\begin{array}{ccccccc} \bullet & \bullet & \bullet & \bullet & \cdots & & \\ & \mathbf{v} & & \mathbf{v}' \geq \mathbf{v} & & & \end{array}$$

Finiteness of the Reduced Reachability Tree

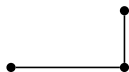
Theorem

The reduced reachability tree of a pushdown VAS \mathcal{A} is finite.

Proof. By contradiction, assume that it is infinite.

The tree is finitely branching. So, by König's Lemma, there is an infinite branch

$$(\mathbf{v}_{\text{init}}, \varepsilon) \rightarrow (\mathbf{v}_1, \sigma_1) \rightarrow (\mathbf{v}_2, \sigma_2) \cdots$$



Finiteness of the Reduced Reachability Tree

Theorem

The reduced reachability tree of a pushdown VAS \mathcal{A} is finite.

Proof. By contradiction, assume that it is infinite.

The tree is finitely branching. So, by König's Lemma, there is an infinite branch

$$(\mathbf{v}_{\text{init}}, \varepsilon) \rightarrow (\mathbf{v}_1, \sigma_1) \rightarrow (\mathbf{v}_2, \sigma_2) \cdots$$



Finiteness of the Reduced Reachability Tree

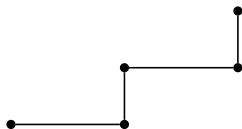
Theorem

The reduced reachability tree of a pushdown VAS \mathcal{A} is finite.

Proof. By contradiction, assume that it is infinite.

The tree is finitely branching. So, by König's Lemma, there is an infinite branch

$$(\mathbf{v}_{\text{init}}, \varepsilon) \rightarrow (\mathbf{v}_1, \sigma_1) \rightarrow (\mathbf{v}_2, \sigma_2) \cdots$$



Finiteness of the Reduced Reachability Tree

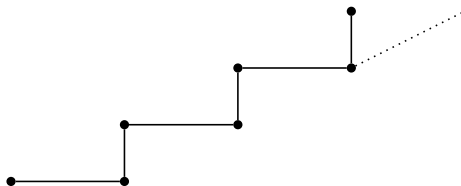
Theorem

The reduced reachability tree of a pushdown VAS \mathcal{A} is finite.

Proof. By contradiction, assume that it is infinite.

The tree is finitely branching. So, by König's Lemma, there is an infinite branch

$$(\mathbf{v}_{\text{init}}, \varepsilon) \rightarrow (\mathbf{v}_1, \sigma_1) \rightarrow (\mathbf{v}_2, \sigma_2) \cdots$$



Finiteness of the Reduced Reachability Tree

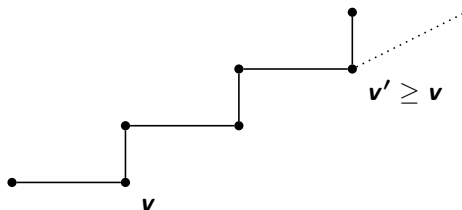
Theorem

The reduced reachability tree of a pushdown VAS \mathcal{A} is finite.

Proof. By contradiction, assume that it is infinite.

The tree is finitely branching. So, by König's Lemma, there is an infinite branch

$$(\mathbf{v}_{\text{init}}, \varepsilon) \rightarrow (\mathbf{v}_1, \sigma_1) \rightarrow (\mathbf{v}_2, \sigma_2) \cdots$$



□

Theorem

A pushdown VAS \mathcal{A} is unbounded if, and only if, its reduced reachability tree contains a path

$$\underbrace{(\mathbf{v}, \sigma) \rightsquigarrow \dots \rightsquigarrow (\mathbf{v}', \sigma')}_{\sigma \text{ remains a prefix}}$$

such that $\mathbf{v} \leq \mathbf{v}'$ and $\sigma \leq_{\text{prefix}} \sigma'$, and at least one of these inequalities is strict.

How big can the reduced reachability tree be?

Table of Contents

- 1 Pushdown Vector Addition Systems
- 2 Reduced Reachability Tree for Pushdown VAS
- 3 Worst-Case Size of the Reduced Reachability Tree
- 4 Conclusion

Fast Growing Functions

Functions $F_\alpha : \mathbb{N} \rightarrow \mathbb{N}$, indexed by ordinals $\alpha \leq \omega^\omega$

$$\begin{aligned}F_0(n) &= n + 1, \\F_{\alpha+1}(n) &= F_\alpha^{n+1}(n) \\F_\lambda(n) &= F_{\lambda_n}(n) \quad \text{if } \lambda \text{ is a limit ordinal}\end{aligned}$$

F_1 : linear, F_2 exponential, F_3 tower of exponentials

F_ω is an Ackermannian function

F_{ω^ω} is an **hyper-Ackermannian** function

Example

$$\begin{aligned}F_{\omega^\omega}(2) &= F_{\omega^3}(2) = F_{\omega^{2.3}}(2) \\&= F_{\omega^{2.2+\omega.3}}(2) \\&= F_{\omega^{2.2+\omega.2+3}}(2) \\&= F_{\omega^{2.2+\omega.2+2}}(F_{\omega^{2.2+\omega.2+2}}(F_{\omega^{2.2+\omega.2+2}}(2)))\end{aligned}$$

Hyper-Ackermannian Bounds

Theorem

The height of the reduced reachability tree of a pushdown VAS \mathcal{A} is at most $F_{\omega^{(d+1)}}(|\mathcal{A}|)$ where d is the dimension of \mathcal{A} .

Corollary

The size of the reduced reachability tree of a pushdown VAS \mathcal{A} is at most

- *multiply-recursive in $|\mathcal{A}|$ when the dimension d is fixed,*
- *hyper-Ackermannian in $|\mathcal{A}|$ when the dimension is part of the input.*

Theorem

For all $n \in \mathbb{N}$, there exists a pushdown VAS \mathcal{A}_n , of size quadratic in n , such that the reduced reachability tree of \mathcal{A}_n has at least $F_{\omega^{\omega}}(n)$ nodes.

Lower Bound

Weak computation of $F_{\omega^d}(n)$ by a **bounded** pushdown VASS $\mathcal{A}_d(n)$

- ⇒ Use the stack to implement recursive calls
- ⇒ Maintain n in 2 counters r and \bar{r} such that $r + \bar{r} = n + 1$
- ⇒ Maintain $\alpha = \omega^d \cdot c_d + \dots + \omega^0 \cdot c_0$ in $d + 1$ counters
- ⇒ Implement the inductive definition of F_α by pushdown VAS rules

Lower Bound

Weak computation of $F_{\omega^d}(n)$ by a **bounded** pushdown VASS $\mathcal{A}_d(n)$

⇒ Use the stack to implement recursive calls

But we cannot store the calling context α !

⇒ Maintain n in 2 counters r and \bar{r} such that $r + \bar{r} = n + 1$

⇒ Maintain $\alpha = \omega^d \cdot c_d + \dots + \omega^0 \cdot c_0$ in $d + 1$ counters

⇒ Implement the inductive definition of F_α by pushdown VAS rules

Lower Bound

Weak computation of $F_{\omega^d}(n)$ by a **bounded** pushdown VASS $\mathcal{A}_d(n)$

⇒ Use the stack to implement recursive calls

But we cannot store the calling context α !

⇒ Maintain n in 2 counters r and \bar{r} such that $r + \bar{r} = n + 1$

⇒ Maintain $\alpha = \omega^d \cdot c_d + \dots + \omega^0 \cdot c_0$ in $d + 1$ counters

⇒ Implement the inductive definition of F_α by pushdown VAS rules

Trick to restore the calling context α of pending recursive calls

- Push the operations (increments and decrements) that are being performed on c_0, \dots, c_d
- Revert them when popping

Upper Bound for VAS — Following [Figueira et al. 2011]

Each branch of the RRT is a **bad sequence** over (\mathbb{N}^d, \leq)



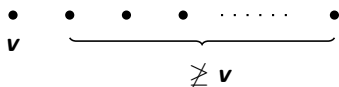
Bad sequences are finite, but can be arbitrarily long

A sequence v_0, v_1, \dots is **n -controlled** if $\|v_i\|_\infty \leq n + i$ for all $i \geq 0$

Given $S \subseteq \mathbb{N}^d$, define $L_S(n)$ to be the maximum length of n -controlled bad sequences over S

Upper Bound for VAS — Following [Figueira et al. 2011]

Each branch of the RRT is a **bad sequence** over (\mathbb{N}^d, \leq)



Bad sequences are finite, but can be arbitrarily long

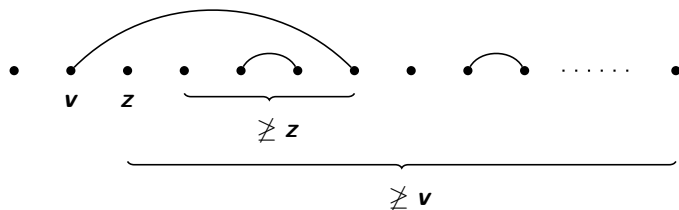
A sequence v_0, v_1, \dots is **n -controlled** if $\|v_i\|_\infty \leq n + i$ for all $i \geq 0$

Given $S \subseteq \mathbb{N}^d$, define $L_S(n)$ to be the maximum length of n -controlled bad sequences over S

$$L_S(n) = \max_{\mathbf{v} \in S, \|\mathbf{v}\|_\infty \leq n} 1 + L_{S/\mathbf{v}}(n+1)$$

Upper Bound for Pushdown VAS

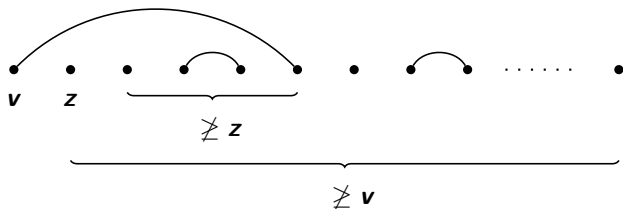
Each branch of the RRT is a **bad nested sequence** over (\mathbb{N}^d, \leq)



Define the maximum length of n -controlled bad nested sequences in the same way as non-nested ones

Upper Bound for Pushdown VAS

Each branch of the RRT is a **bad nested sequence** over (\mathbb{N}^d, \leq)



Define the maximum length of n -controlled bad nested sequences in the same way as non-nested ones

$$L_S(n) = \max_{v \in S, \|v\|_\infty \leq n} 1 + L_{S/v}(n+1) + L_{S/v}(n+1 + L_{S/v}(n+1))$$

Table of Contents

- 1 Pushdown Vector Addition Systems
- 2 Reduced Reachability Tree for Pushdown VAS
- 3 Worst-Case Size of the Reduced Reachability Tree
- 4 Conclusion

Summary

- ⇒ Extension of the reduced reachability tree from VAS to pushdown VAS
 - In the paper: extension to well-structured pushdown systems
- ⇒ Boundedness and termination are decidable for pushdown VAS
- ⇒ Hyper-Ackermannian ($F_{\omega\omega}$) worst-case running time
 - The reduced reachability tree of a pushdown VAS \mathcal{A} has at most $F_{\omega\omega}(|\mathcal{A}|)$ nodes
 - This bound is tight
- ⇒ Bounds on the reachability set when it is finite

Open Problems

- ⇒ Complexity of the boundedness problem for pushdown VAS
 - Lower bound: tower of exponentials (F_3) from [Lazić 2012]
 - Upper bound: hyper-Ackermann (F_{ω^ω})

- ⇒ Decidability of coverability / reachability for Pushdown VAS
 - Coverability decidable for 1-dim VASS + stack [Submitted]
 - Reachability open even for 1-dim VASS + stack

- ⇒ Complexity of these problems for VAS + full counter
 - Coverability for this model is harder than reachability for VAS

Thank You

Definition

A **pushdown vector addition system** is a triple $\langle \mathbf{v}_{\text{init}}, \Gamma, \Delta \rangle$ where

- $\mathbf{v}_{\text{init}} \in \mathbb{N}^d$: **initial vector**
- Γ : finite **stack alphabet**
- $\Delta \subseteq (\mathbb{Z}^d \times \text{Op})$: finite set of **actions**, with

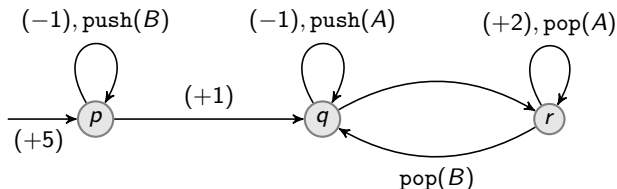
$$\text{Op} = \{\varepsilon\} \cup \{\text{push}(\gamma), \text{pop}(\gamma) \mid \gamma \in \Gamma\}$$

Definition

A **pushdown vector addition system** is a triple $\langle \mathbf{v}_{\text{init}}, \Gamma, \Delta \rangle$ where

- $\mathbf{v}_{\text{init}} \in \mathbb{N}^d$: **initial vector**
- Γ : finite **stack alphabet**
- $\Delta \subseteq (\mathbb{Z}^d \times \text{Op})$: finite set of **actions**, with

$$\text{Op} = \{\varepsilon\} \cup \{\text{push}(\gamma), \text{pop}(\gamma) \mid \gamma \in \Gamma\}$$



The **semantics** of a pushdown VAS $\langle \mathbf{v}_{\text{init}}, \Gamma, \Delta \rangle$ is the transition system $\langle \mathbb{N}^d \times \Gamma^*, (\mathbf{v}_{\text{init}}, \varepsilon), \rightarrow \rangle$ whose transition relation \rightarrow is given by

$$\frac{(\mathbf{a}, \varepsilon) \in \Delta \wedge \mathbf{v}' = \mathbf{v} + \mathbf{a} \geq \mathbf{0}}{(\mathbf{v}, \sigma) \rightarrow (\mathbf{v}', \sigma)}$$

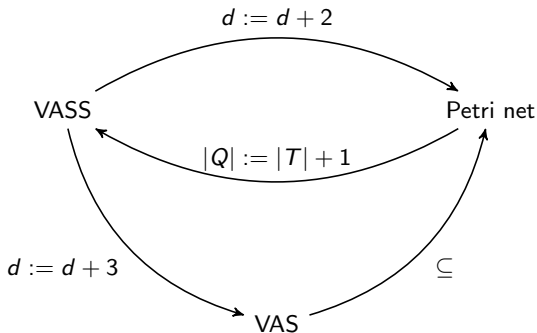
$$\frac{(\mathbf{a}, \text{push}(\gamma)) \in \Delta \wedge \mathbf{v}' = \mathbf{v} + \mathbf{a} \geq \mathbf{0}}{(\mathbf{v}, \sigma) \rightarrow (\mathbf{v}', \sigma \cdot \gamma)}$$

$$\frac{(\mathbf{a}, \text{pop}(\gamma)) \in \Delta \wedge \mathbf{v}' = \mathbf{v} + \mathbf{a} \geq \mathbf{0}}{(\mathbf{v}, \sigma \cdot \gamma) \rightarrow (\mathbf{v}', \sigma)}$$

VASs \simeq Petri nets \simeq VASSs

Additional Feature of Petri nets

Test $x \geq cst$ without modifying x



Pushdown VASS $\mathcal{A}_d(n)$ for the Lower Bound

