Hyper-Ackermannian Bounds for Pushdown Vector Addition Systems

Jérôme Leroux[1] M. Praveen[2] Grégoire Sutre[1]

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

[2]Chennai Mathematical Institute, Chennai, India

ACTS, Chennai, India, Feb 2015

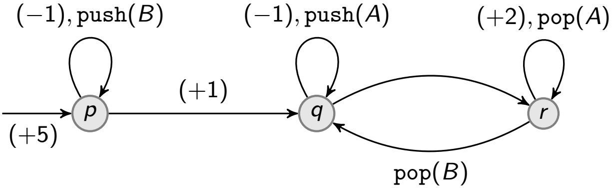
Table of Contents

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: $\boldsymbol{a} \in \mathbb{Z}^d$
- semantics: $\underbrace{\boldsymbol{v}}_{\in\mathbb{N}^d} \to \underbrace{\boldsymbol{v} + \boldsymbol{a}}_{\in\mathbb{N}^d}$

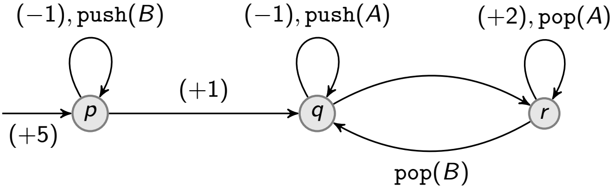
$+$

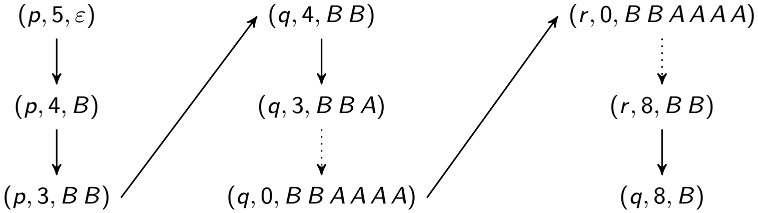
Stack

finite stack alphabet

push and pop

Pushdown Vector Addition Systems — Model

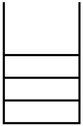




Pushdown Vector Addition Systems — Motivations

VASS
$\wr|$
VAS
$\wr|$
Petri net

$+$

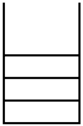


➪ Richer model for the verification of concurrent systems

- Multi-threaded recursive programs
- One recursive server + unboundedly many finite-state clients

VASS
$\wr|$
VAS
$\wr|$
Petri net

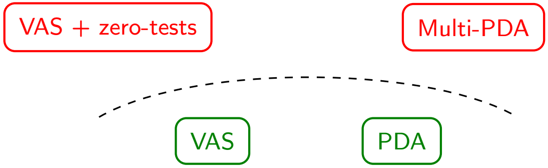
$+$



➪ 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	ExpSpace-c[1,2]	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

[4] Kosaraju 1982

[5] Finkel, Sangnier 2010

[6] Reinhardt 2008

[7] Lazić 2012

Brief State of the Art

	Boundedness	Coverability	Reachability
VAS	ExpSpace-c[1,2]	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]
- VAS ∩ CFL of finite index [Atig, Ganty 2011]

Brief State of the Art

	Boundedness	Coverability	Reachability
VAS	ExpSpace-c[1,2]	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]
- VAS $\cap$ CFL of finite index [Atig, Ganty 2011]

Our Contribution

➪ Boundedness is decidable for pushdown VAS

- Reduced reachability tree: adaptation of the VAS case

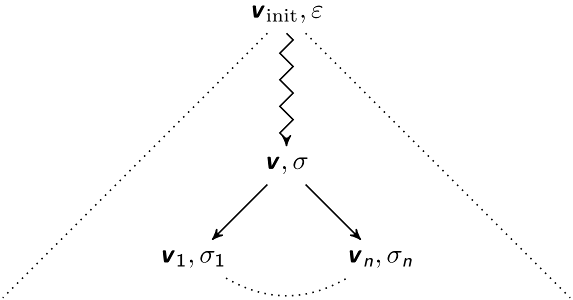
➪ Worst-case complexity of the algorithm: hyper-Ackermannian

- 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

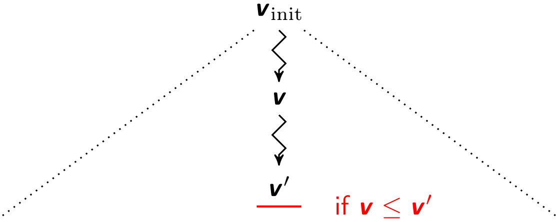


➪ Exhaustive and enumerative forward exploration from $(\boldsymbol{v}_{\text{init}}, \varepsilon)$

➪ Potentially infinite, need to truncate

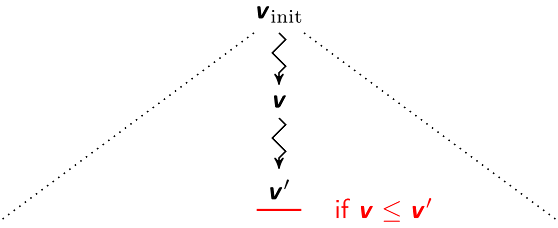
Reduced Reachability Tree for VAS [Karp, Miller 1969]

Truncation rule:



Reduced Reachability Tree for VAS [Karp, Miller 1969]

Truncation rule:



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

Truncation entails that

- $\boldsymbol{v}_{\text{init}} \xrightarrow{*} \boldsymbol{v} \xrightarrow{*} \boldsymbol{v}' \xrightarrow{*} \boldsymbol{v}'' \xrightarrow{*} \boldsymbol{v}''' \cdots$
- If $\boldsymbol{v} < \boldsymbol{v}'$ then $\boldsymbol{v} < \boldsymbol{v}' < \boldsymbol{v}'' < \boldsymbol{v}''' < \cdots$

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 $\square$

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

$$(\boldsymbol{v}_{\mathrm{init}}, \varepsilon) \xrightarrow{*} (\boldsymbol{v}, \sigma) \xrightarrow{*} (\boldsymbol{v}', \sigma')$$

such that

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

Then $(\boldsymbol{v}_{\mathrm{init}}, \varepsilon) \xrightarrow{*} (\boldsymbol{v}, \sigma) \xrightarrow{*} (\boldsymbol{v}', \sigma') \xrightarrow{*} (\boldsymbol{v}'', \sigma'') \xrightarrow{*} (\boldsymbol{v}''', \sigma''') \cdots$

Tentative Simulation-Based Truncation for Pushdown VAS

Consider a run

$$(\boldsymbol{v}_{\mathrm{init}}, \varepsilon) \xrightarrow{*} (\boldsymbol{v}, \sigma) \xrightarrow{*} (\boldsymbol{v}', \sigma')$$

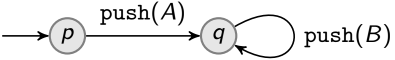
such that

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

Then $(\boldsymbol{v}_{\mathrm{init}}, \varepsilon) \xrightarrow{*} (\boldsymbol{v}, \sigma) \xrightarrow{*} (\boldsymbol{v}', \sigma') \xrightarrow{*} (\boldsymbol{v}'', \sigma'') \xrightarrow{*} (\boldsymbol{v}''', \sigma''') \cdots$

But:

$$\begin{array}{c} (p, \varepsilon) \\ \downarrow \\ (q, A) \\ \downarrow \\ (q, A\,B) \\ \downarrow \\ (q, A\,B\,B) \\ \vdots \\ \downarrow \end{array}$$



No truncation, infinite branch!

Truncation rule:

$$\boldsymbol{v}_{\mathrm{init}}, \varepsilon \rightsquigarrow \boldsymbol{v}, \sigma \overset{(\boldsymbol{u}, \rho)}{\rightsquigarrow} \underline{\boldsymbol{v}', \sigma'} \quad \text{if } \begin{cases} \boldsymbol{v} \leq \boldsymbol{v}' \\ \sigma \leq_{\text{prefix}} \rho \text{ for all } (\boldsymbol{u}, \rho) \end{cases}$$

Truncation entails that

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

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

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

•

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

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

$$\begin{array}{ccccc} \bullet & \bullet & \bullet & \bullet & \cdots \\ & \boldsymbol{v} & & \boldsymbol{v}' \geq \boldsymbol{v} & \end{array}$$

Finiteness of the Reduced Reachability Tree

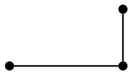
Theorem

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

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



Finiteness of the Reduced Reachability Tree

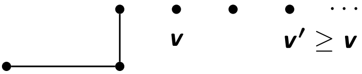
Theorem

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

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



Finiteness of the Reduced Reachability Tree

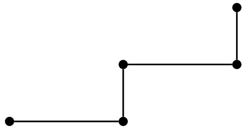
Theorem

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

$$(\boldsymbol{v}_{\mathrm{init}}, \varepsilon) \to (\boldsymbol{v}_1, \sigma_1) \to (\boldsymbol{v}_2, \sigma_2) \cdots$$



Finiteness of the Reduced Reachability Tree

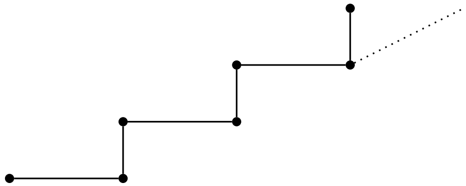
Theorem

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

$$(\boldsymbol{v}_{\mathrm{init}}, \varepsilon) \to (\boldsymbol{v}_1, \sigma_1) \to (\boldsymbol{v}_2, \sigma_2) \cdots$$



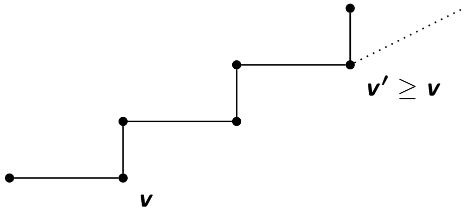
Theorem

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

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



□

RRT-based Algorithm for Pushdown VAS Boundedness

Theorem

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

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

such that $\boldsymbol{v} \leq \boldsymbol{v}'$ and $\sigma \leq_{\mathit{prefix}} \sigma'$, 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 $F_\alpha : \mathbb{N} \to \mathbb{N}$, indexed by ordinals $\alpha \leq \omega^\omega$

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

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 $\mathtt{r}$ and $\overline{\mathtt{r}}$ such that $\mathtt{r} + \overline{\mathtt{r}} = n + 1$
- ➪ Maintain $\alpha = \omega^d.\, c_d + \cdots + \omega^0.\, c_0$ 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 $\mathtt{r}$ and $\overline{\mathtt{r}}$ such that $\mathtt{r} + \overline{\mathtt{r}} = n + 1$

- Maintain $\alpha = \omega^d . c_d + \cdots + \omega^0 . c_0$ 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 $\mathtt{r}$ and $\overline{\mathtt{r}}$ such that $\mathtt{r} + \overline{\mathtt{r}} = n + 1$

➪ Maintain $\alpha = \omega^d . c_d + \cdots + \omega^0 . c_0$ 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 $\mathrm{c}_0, \ldots, \mathrm{c}_d$
- Revert them when popping

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

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

$$\bullet \quad \bullet \quad \underset{\boldsymbol{v}}{\bullet} \quad \underbrace{\bullet \quad \bullet \quad \bullet \quad \cdots\cdots \quad \bullet}_{\not\geq \boldsymbol{v}}$$

Bad sequences are finite, but can be arbitrarily long

A sequence $v_0, v_1, \ldots$ is n-controlled if $\|\boldsymbol{v}_i\|_\infty \leq n + i$ 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 $(\mathbb{N}^d, \leq)$

$$\underset{\boldsymbol{v}}{\bullet} \qquad \underbrace{\bullet \qquad \bullet \qquad \bullet \qquad \cdots\cdots \qquad \bullet}_{\not\geq\, \boldsymbol{v}}$$

Bad sequences are finite, but can be arbitrarily long

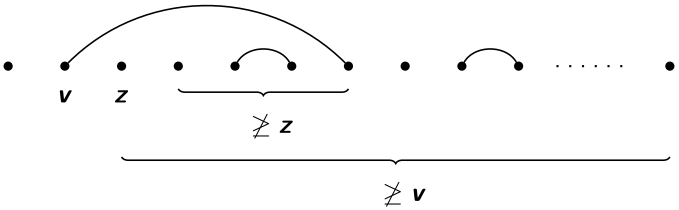
A sequence $v_0, v_1, \ldots$ is *n-controlled* if $\|\boldsymbol{v}_i\|_\infty \leq n + i$ for all $i \geq 0$

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

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

Upper Bound for Pushdown VAS

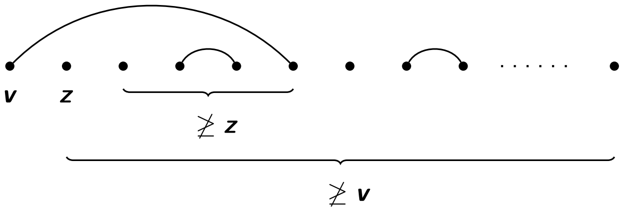
Each branch of the RRT is a bad nested sequence over $(\mathbb{N}^d, \leq)$



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

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

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_{ω^ω}) 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

Pushdown VAS — Syntax

Definition

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

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

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

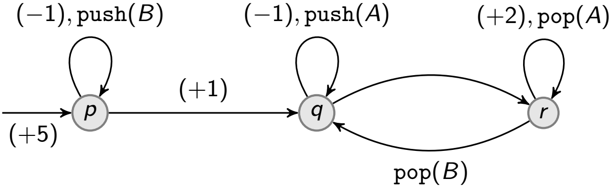
Pushdown VAS — Syntax

Definition

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

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

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



Pushdown VAS — Semantics

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

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

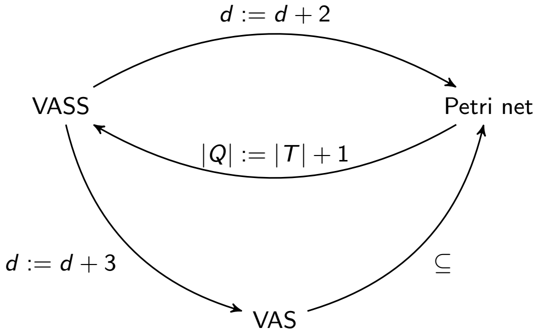
$$\frac{(\boldsymbol{a}, \mathrm{push}(\gamma)) \in \Delta \;\wedge\; \boldsymbol{v}' = \boldsymbol{v} + \boldsymbol{a} \geq \mathbf{0}}{(\boldsymbol{v}, \sigma) \rightarrow (\boldsymbol{v}', \sigma \cdot \gamma)}$$

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

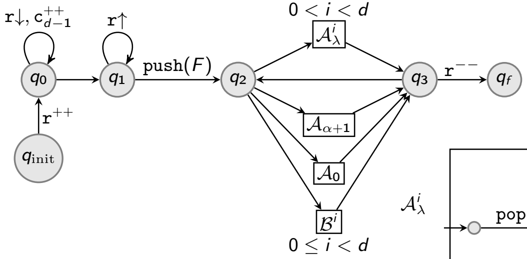
VASs $\simeq$ Petri nets $\simeq$ VASSs

Additional Feature of Petri nets

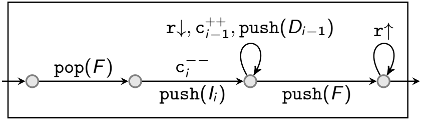
Test $\mathtt{x} \geq cst$ without modifying $\mathtt{x}$



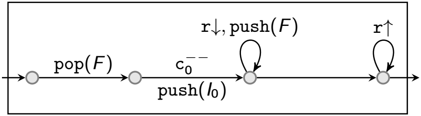
Pushdown VASS $\mathcal{A}_d(n)$ for the Lower Bound



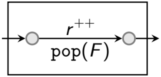
$\mathcal{A}_\lambda^i$



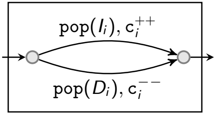
$\mathcal{A}_{\alpha+1}$



$$\mathtt{r}\!\downarrow \;=\; \mathtt{r}^{--}, \overline{\mathtt{r}}^{++}$$
$$\mathtt{r}\!\uparrow \;=\; \mathtt{r}^{++}, \overline{\mathtt{r}}^{--}$$



$\mathcal{A}_0$



$\mathcal{B}^i$