| | |
|---|---|
| Title | Well-Structured Pushdown Systems, Part 1: Decidable Classes for Coverability |
| Author(s) | Cai, Xiaojuan; Ogawa, Mizuhito |
| Citation | Research report (School of Information Science, Japan Advanced Institute of Science and Technology), IS-RR-2013-001: 1-19 |
| Issue Date | 2013-01-08 |
| Type | Technical Report |
| Text version | publisher |
| URL | http://hdl.handle.net/10119/11347 |
| Rights | |
| Description | リサーチレポート（北陸先端科学技術大学院大学情報科学研究科） |

# Well-Structured Pushdown Systems, Part 1: Decidable Classes for Coverability

Xiaojuan Cai
BASICS Lab, Shanghai Jiotong University

Mizuhito Ogawa
*School of Information Science*
*Japan Advanced Institute of Science and Technology*

January 8, 2013
IS-RR-2013-001

# Well-Structured Pushdown Systems, Part I: Decidable classes for Coverability*

Xiaojuan Cai[1] and Mizuhito Ogawa[2]

[1] BASICS Lab, Shanghai Jiao Tong University, China
cxj@sjtu.edu.cn
[2] Japan Advanced Institute of Science and Technology, Japan
mizuhito@jaist.ac.jp

**Abstract.** *Pushdown systems* (PDSs) nicely model single-thread recursive programs, and *well-structured transition systems* (WSTS), such as *vector addition systems*, are useful to represent non-recursive multi-thread programs. Our goal is to investigate *well-structured pushdown systems* (WSPDSs), pushdown systems with well-quasi-ordered control states and stack alphabet, to combine these ideas.
This paper focuses on decidable classes of coverability and extends P-automata techniques for configuration reachability of PDSs to those for coverability of WSPDSs, in forward and backward ways. A *Post*$^*$-automata (resp. *Pre*$^*$-automata) construction is combined with Karp-Miller acceleration (resp. ideal representation) to characterize the set of successors (resp. predecessors) of given configurations. We show decidability results of coverability, which include *recursive vector addition system with states* [1], *multi-set pushdown systems* [2, 3], and a WSPDS with finite control states and well-quasi-ordered stack alphabet.

## 1 Introduction

There are two directions of infinite (discrete) state systems. A *pushdown system* (PDS) consists of finite control states and finite stack alphabet, where a stack stores the context. It nicely models single-thread recursive programs. *Well-structured transition systems* (WSTS) [4, 5] consists of a well-quasi-ordered set of states, in which *Vector addition system* (VAS, or Petri Net) is a typical example. It often works for modeling dynamic thread creation of multi-thread program [6]. Our naive motivation comes from what happens when we combine them as a general framework for modeling recursive multi-thread programs.

Ramalingam [7] showed that a 3-thread recursive program with synchronization mechanism can solve *Post-correspondence-problem.* This is a natural result since a 2-stack PDS is Turing complete. Roughly speaking, there are two sources to be Turing complete in a 2-stack PDS. i) the depth of both stacks is unbounded. ii) the interleaving between two stacks can be arbitrarily many. By restricting i),

* This work is supported by the NSFC-JSPS joint project (61011140074) and NSFC projects (61003013,61100052)

Qadeer an Rehof proposed context-bounded pushdown model [8], in which the number of context switching is bounded. The idea is after a bounded number of context switching, only one stack can work, so that it is simulated by a single stack. Atig, *et.al.* further extended with dynamic thread creation [6].

By restricting ii), Sen *et. al.* [2] proposed Multi-set pushdown systems (Multi-set PDSs) to model multi-thread asynchronous programs, and Bouajjani and Emmi [1] proposed a Recursive Vector Addition System with States (RVASS) to model multi-thread programs with fork/join synchronizations. They showed decidability of the coverability and the state reachability, respectively. Note that the coverability lies between the configuration reachability and the state reachability. They are single stack PDSs with infinite control states and stack alphabet, which are beyond ordinary PDSs with finite control states and stack alphabet.

The *configuration reachability*, i.e., to determine whether a target configuration is reachable from an initial configuration, is decidable for ordinary PDSs. In implementation, P-automata construction is a popular technique, which can be tracked back to Büchi's seminal work [9], and has been clarified in [10–12]. There are two kinds of P-automata constructions. A $Post^*$ automaton computes the set of successor configurations from an initial configuration, and a $Pre^*$ automaton computes the set of predecessor configurations from a target configuration.

Different from PDSs, a popular property of WSTSs is *coverability*, which is reachability from an initial configuration to a certain configuration that covers the target configuration. There are also forward and backward proof techniques. For instance, in case of VASs, Karp-Miller acceleration [13] is a typical instance of the former, which was generalized in [14, 15]. For the latter, an ideal (i.e., an upward closed set) representation is a typical technique [4, 5]. Note that the reachability is hard for WSTSs. For instance, the reachability of VASs stays decidable, but its proof requires deep insight on Presburger arithmetic [16, 17].

Our ultimate goal is to investigate *well-structured pushdown systems* (WSPDSs), pushdown systems with well-quasi-ordered control states and stack alphabet, to combine PDSs and WSTSs. This paper focuses on decidable classes of coverability and extends P-automata techniques for configuration reachability of PDSs to those for coverability of WSPDSs, in forward and backward ways. $Post^*$-automata (resp. $Pre^*$-automata) construction is combined with Karp-Miller acceleration (resp. ideal representation) to characterize the set of successors (resp. predecessors) of given configurations. We show decidability results of coverability, which include RVASSs [1], Multi-set PDSs [2, 3], and a WSPDS with finite control states and WQO stack alphabet. The first one extends the decidability of the state reachability of RVASSs [1] to that of the coverability.

## Related Work

Combining PDSs and VASs is not new. Process rewrite system (PRS) [18] is a pioneer work on such combination. A PRS is a(n AC) ground term rewriting system, consisting of the sequential composition ".", the parallel composition "|| ", and finitely many constants, which can be regarded as a PDS with finite control states and vector stack alphabet. The decidability of the reachability between ground terms was shown based on the reachability of a VAS. However, a PRS is rather weak to model multi-thread programs, since it cannot describe vector additions between adjacent stack frames during push/pop operations.

An RVASS [1], in which we are inspired, allows vector additions during pop rules. The state reachability was shown by reduction of an RVASS into a Branching VASS [19]. Our WSPDS framework extends the decidability result to the coverability. A more general framework is a WQO automaton [20], which is a WSTS with auxiliary storage (e.g., stacks and queues). Although in general undecidable, its coverability becomes decidable under the compatibility of *rank* functions with a WQO. An Multi-set PDS [3, 2] is such a instance.

To sum up, our contribution is a simplified framework, which has more focus on well-quasi-ordered stack alphabet, and a unified proof methodology based on extensions of P-automata techniques.

## 2 Preliminaries

### 2.1 Well-structured transition system

A *quasi-order* $(D, \leq)$ is a reflexive transitive binary relation on $D$. An upward closure of $X \subseteq D$, denoted by $X^{\uparrow}$, includes all elements in $D$ larger than elements in $X$, i.e., $X^{\uparrow} = \{d \in D \mid \exists x \in X.x \leq d\})$. A subset $I$ is an *ideal* if $I = I^{\uparrow}$. Similarly, a downward closure of $X \subseteq D$ is denoted by $X^{\downarrow} = \{d \in D \mid \exists x \in X.x \geq d\}$. We denote the set of all ideals by $\mathcal{I}(D)$. A quasi-order $(D, \leq)$ is a *well-quasi-order* (WQO) if, for each infinite sequence $a_1, a_2, a_3, \cdots$ in $D$, there exist $i, j$ with $i < j$ and $a_i \leq a_j$.

**Definition 1.** *A* well-structured transition system *(WSTS) is a triplet $M = \langle (P, \preceq), \Delta \rangle$ where $(P, \preceq)$ is a WQO, and $\Delta \subseteq P \times P$ is the set of transitions. We write $p \to q$ if $(p, q) \in \Delta$.*

*$M$ is* monotonic *if, for each $p_1, q_1, p_2 \in P$, $p_1 \to q_1 \wedge p_1 \preceq p_2$ implies $\exists q_2 \,.\, p_2 \to q_2 \ \wedge q_1 \preceq q_2$.*

Given two states $p, q \in P$, the *coverability* problem is to determine whether there exists some $q' \succeq q$ and $p \to^* q'$.

*Vector addition systems* (VAS) (equivalently, Petri net) are WSTSs, with vectors as states and additions as transition rules. The reachability problem of VAS is decidable [16, 17]. It is elegant, but too difficult to implement. The coverability also attracts attentions and is implemented, such as in **Pep**. *Karp-Miller acceleration* is an efficient technique for the coverability. If there is a descendant vector (wrt transitions) strictly larger than one of its ancestors on some coordinates, values at these coordinates are accelerated to $\omega$.

There is an alternative backward method to decide coverability for a WSTS, beyond VASs. Starting from an ideal $\{q\}^{\uparrow}$, where $q$ is the target state to be covered, its predecessors are repeatedly computed. Note that, for a monotonic WSTS and an ideal $I(\subseteq P)$, the predecessor set $pre(I) = \{p \in P \mid \exists q \in I.p \to q\}$ is also an ideal. Its termination is obtained by the following lemma.

**Lemma 1.** *[5] $(D, \leq)$ is a WQO, if, and only if, any infinite sequence $I_0 \subseteq I_1 \subseteq I_2 \subseteq \cdots$ in $\mathcal{I}(D)$ eventually stabilize.*

From now on, we denote $\mathbb{N}$ (resp. $\mathbb{Z}$) for the set of natural numbers (resp. integers), and $\mathbb{N}^k$ (resp. $\mathbb{Z}^k$) is the set of $k$-dimensional vectors over $\mathbb{N}$ (resp. $\mathbb{Z}$). As notational convention, $\boldsymbol{n}, \boldsymbol{m}$ are for vectors in $\mathbb{N}^k$, $\boldsymbol{z}, \boldsymbol{z}'$ are for vectors in $\mathbb{Z}^k$, $\widetilde{\boldsymbol{n}}, \widetilde{\boldsymbol{m}}$ are for sequences of vectors.

### 2.2 Pushdown system

We define a pushdown system (PDS) with extra rules, *simple-push* and *nonstandard-pop*. These extra rules do not appear in the standard definition, but they can be encoded into standard rules. For example, a non-standard pop rule $(p, \alpha\beta \to q, \gamma)$ can be split into $(p, \alpha \to p_\alpha, \epsilon)$ and $(p_\alpha, \beta \to q, \gamma)$ by adding an intermediate state $p_\alpha$. However, later we will consider a PDS with infinite stack alphabet, and this encoding may change the context. For instance, when a PDS has finite states and infinite stack alphabet, the encoding of nonstandard pop rules make a PDS with both infinite states and stack alphabet.

**Definition 2.** *A* pushdown system (PDS) *is a triplet $\langle P, \Gamma, \Delta \rangle$ where*

- *$P$ is a finite set of states,*
- *$\Gamma$ is finite stack alphabet, and*
- *$\Delta \subseteq P \times \Gamma^{\leq 2} \times P \times \Gamma^{\leq 2}$ is a finite set of transitions, where $(p, v, q, w) \in \Delta$ is denoted by $(p, v \to q, w)$.*

We use $\alpha, \beta, \gamma, \cdots$ to range over $\Gamma$, and $w, v, \cdots$ over words in $\Gamma^*$. A *configuration* $\langle p, w \rangle$ is a pair of a state $p$ and a stack content (word) $w$. As convention, we denote configurations by $c_1, c_2, \cdots$. One step transition $\hookrightarrow$ between configurations is defined as follows. $\hookrightarrow^*$ is the reflexive transitive closure of $\hookrightarrow$.

$$\text{inter}\frac{(p, \gamma \to p', \gamma') \in \Delta}{\langle p, \gamma w \rangle \hookrightarrow \langle p', \gamma' w \rangle} \quad \text{push}\frac{(p, \gamma \to p', \alpha\beta) \in \Delta}{\langle p, \gamma w \rangle \hookrightarrow \langle p', \alpha\beta w \rangle} \quad \text{pop}\frac{(p, \gamma \to p', \epsilon) \in \Delta}{\langle p, \gamma w \rangle \hookrightarrow \langle p', w \rangle}$$

$$\text{simple-push}\frac{(p, \epsilon \to p', \alpha) \in \Delta}{\langle p, w \rangle \hookrightarrow \langle p', \alpha w \rangle} \quad \text{nonstandard-pop}\frac{(p, \alpha\beta \to p', \gamma) \in \Delta}{\langle p, \alpha\beta w \rangle \hookrightarrow \langle p', \gamma w \rangle}$$

A PDS enjoys decidable *reachability*, i.e., given configurations $\langle p, w \rangle$, $\langle q, v \rangle$ with $p, q \in P$ and $w, v \in \Gamma^*$, decide whether $\langle p, w \rangle \hookrightarrow^* \langle q, v \rangle$.

## 3 WSPDS and P-automata technique

### 3.1 P-automaton

P-automaton is an automaton which exactly accepts the reachable configurations of some PDS. Distinguished by the forward and backward of transitions, P-automata are classified into $Post^*$-automata and $Pre^*$-automata.

**Definition 3.** *Given a PDS $M = \langle P, \Gamma, \Delta \rangle$, a P-automaton $\mathcal{A}$ is a quadruplet $(S, \Gamma, \nabla, F)$ where*

- *$F$ is the set of final states, and $P \subseteq S \setminus F$, and*
- *$\nabla \subseteq S \times (\Gamma \cup \{\epsilon\}) \times S$.*

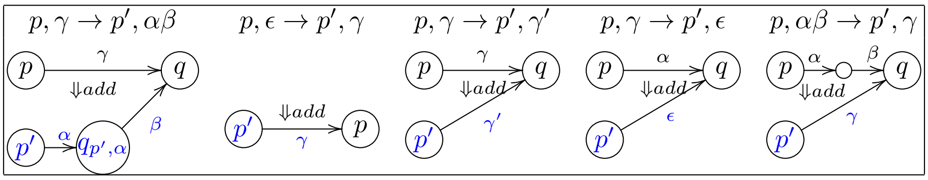
*We write $s \overset{\gamma}{\mapsto} s'$ for $(s, \gamma, s') \in \nabla$ and $\Mapsto$ for the reflexive transitive closure of $\mapsto$; It* accepts *$\langle p, w \rangle$ for $p \in P$ and $w \in \Gamma^*$ if $p \overset{w}{\Longmapsto} f \in F$. We use $L(\mathcal{A})$ to denote the set of configurations $\mathcal{A}$ accepts. In an* initial P-automaton*, we assume there is no transitions $s \overset{\gamma}{\mapsto} s'$ such that $s' \in P$.*

Let $C_0$ be a regular set of configurations of a PDS, and let $\mathcal{A}_0$ be an initial P-automaton that accepts $C_0$. The procedure for computing $post^*(C_0)$ starts from $\mathcal{A}_0$, and repeatedly adds edges according to the rules of a PDS until convergence. We call this procedure *saturation*. The $Post^*$-saturation rules are given in Definition 4 and illustrated in followed diagram.

**Definition 4.** *For a PDS $\langle P, \Gamma, \Delta \rangle$, let $\mathcal{A}_0$ be an initial P-automaton accepting $C_0$. $Post^*(\mathcal{A}_0)$ is the result of repeated applications of the following $Post^*$-saturation rules.*

$$\frac{(S, \Gamma, \nabla, F), \quad (p \overset{w}{\Longmapsto} q) \in \nabla}{(S \cup \{p'\}, \Gamma, \nabla \cup \{p' \overset{\gamma}{\mapsto} q\}, F)} \; (p, w \to p', \gamma) \in \Delta, |w| \leq 2$$

$$\frac{(S, \Gamma, \nabla, F), \quad (p \overset{\gamma}{\mapsto} q) \in \nabla}{(S \cup \{p', q_{p',\alpha}\}, \Gamma, \nabla \cup \{p' \overset{\alpha}{\mapsto} q_{p',\alpha} \overset{\beta}{\mapsto} q\}, F)} \; (p, \gamma \to p', \alpha\beta) \in \Delta$$




For instance, consider a push rule $(p, \gamma \to p', \alpha\beta)$. If $p \overset{\gamma}{\mapsto} q$ is in $\nabla$, then $p' \overset{\alpha}{\mapsto} q_{p',\alpha} \overset{\beta}{\mapsto} q$ is added to $\nabla$. The intuition is if there exists $v \in \Gamma^*$ such that $\langle p, \gamma v \rangle$ is in $post^*(C_0)$, then $\langle p', \alpha\beta v \rangle$ is also in $post^*(C_0)$ by applying rule $(p, \gamma \to p', \alpha\beta)$. The $Pre^*$-saturation rules to compute $pre^*(C_0)$ are similar, but in a backward way.

*Remark 1.* $Post^*$ and $Pre^*$-saturations introduce $\epsilon$-transitions when applying standard pop rules and simple push rules, respectively. $\epsilon$-transitions make arguments complicated, and we preprocess the PDS for saturations.

1. The set $\Gamma$ is extended with $\perp$ to denote the bottom of the stack.
2. For $Post^*$-saturation, every standard pop rule $p, \alpha \to q, \epsilon$ is replaced with $(p, \alpha\gamma \to q, \gamma)$ for each $\gamma \in \Gamma$.

3. For $Pre^*$-saturation, every *simple push* rule $p, \epsilon \to q, \alpha$ is replaced with $(p, \gamma \to q, \alpha\gamma)$ for each $\gamma \in \Gamma$.

**Lemma 2.** *Let $\langle P, \Gamma, \Delta \rangle$ be a PDS, and let $\mathcal{A}_0$ be an initial P-automaton accepting $C_0$. Assume that $p \overset{w}{\Longmapsto} q$ in $Post^*(\mathcal{A}_0)$ and $p \in P$.*

1. *If $q \in P$, $\langle q, \epsilon \rangle \hookrightarrow^* \langle p, w \rangle$;*
2. *If $q \in S(\mathcal{A}_0) \backslash P$, there exists $q' \overset{v}{\Longmapsto} q$ in $\mathcal{A}_0$ with $q' \in P$ and $\langle q', v \rangle \hookrightarrow^* \langle p, w \rangle$.*

Lemma 2 shows the set of accepted configurations by the P-automata always falls into $post^*(C_0)$ during the saturation process (*completeness*). We give an inductive proof in Appendix A. On the other hand, the $Post^*$ saturation rules compute one-step successor configurations, thus all the configurations in $post^*(C_0)$ will finally be accepted by $Post^*(\mathcal{A}_0)$ (*soundness*). Therefore, Lemma 2 leads the correctness of P-automata construction.

**Theorem 1.** *$post^*(C_0) = L(Post^*(\mathcal{A}_0))$, and $pre^*(C_0) = L(Pre^*(\mathcal{A}_0))$.*

### 3.2 P-automata for Coverability

We combine PDSs with WSTSs as *well-structured pushdown systems* (WSPDS). $\mathcal{P}Fun(X, Y)$ denotes the set of partial functions from $X$ to $Y$.

**Definition 5.** *A* well-structured pushdown system *(WSPDS) is a triplet $M = \langle (P, \preceq), (\Gamma, \leq), \Delta \rangle$ where*

- *$(P, \preceq)$ and $(\Gamma, \leq)$ are WQOs, and*
- *$\Delta \subseteq \mathcal{P}Fun(P, P) \times \mathcal{P}Fun(\Gamma^{\leq 2}, \Gamma^{\leq 2})$ is the finite set of transitions rules. We write $(p, w \to \phi(p), \psi(w))$ if $(\phi, \psi) \in \Delta$ and each has definition on $p, w$ respectively.*

A PDS is a WSPDS with finite $P$ and finite $\Gamma$, and WSTS is a WSPDS without operations on stack, i.e., $\Delta \subseteq \mathcal{P}Fun(P, P)$. Let $\lll$, the quasi-order[3] on $\Gamma^*$, be the element-wise extension of $\leq$ on $\Gamma$, i.e., $\alpha_1 \cdots \alpha_n \lll \beta_1 \cdots \beta_m$ if and only if $m = n$ and $\forall i. \alpha_i \leq \beta_i$. $M$ is *monotonic* if, $(\phi, \psi) \in \Delta$ implies $\psi$ and $\phi$ are monotonic functions wrt $\preceq$ and $\lll$. In a monotonic WSPDS, instead of reachability, we consider the *coverability*, i.e., the reachability to an ideal of configurations.

- **Coverability**: Given configurations $\langle p, w \rangle$, $\langle q, v \rangle$ with $p, q \in P$ and $w, v \in \Gamma^*$, we say $\langle p, w \rangle$ *covers* $\langle q, v \rangle$ if there exist $q' \succeq q$ and $v' \ggg v$ s.t. $\langle p, w \rangle \hookrightarrow^* \langle q', v' \rangle$. Coverability problem is to decide whether $\langle p, w \rangle$ *covers* $\langle q, v \rangle$.

There are two ways for the coverability, either forward or backward. The forward method starts from an initial configuration $\langle p, w \rangle$, and computes coverable configurations. The backward method starts from a target configuration $\langle q, v \rangle$, and computes predecessor configurations that cover $\langle q, v \rangle$. For a set of configurations $C_0$, they construct a $Post^*$- and a $Pre^*$-automaton $\mathcal{A}$, respectively.

[3] Note $\lll$ might not be a well-quasi-ordering.

– (Post) $\mathcal{A}$ accepts the downward-closed set of successors of $C_0$, i.e., $L(\mathcal{A}) = \bigcup_{i\geq 0}(post^i(C_0)^{\downarrow}) = (\bigcup_{i\geq 0} post^i(C_0))^{\downarrow} = (post^*(C_0))^{\downarrow}$.
– (Pre) $\mathcal{A}$ accepts predecessors of the upward-closed set $C_0^{\uparrow}$ of $C_0$, i.e., $L(\mathcal{A}) = \bigcup_{i\geq 0} pre^i(C_0^{\uparrow}) = pre^*(C_0^{\uparrow})$.

*Remark 2.* As in Remark 1, we preprocess WSPDSs to eliminate standard pop rules for $Post^*$-saturation and simple push rules for $Pre^*$-saturation. In later decidability results on WSPDSs, the finiteness of the number of transition rules is crucial. We replace partial functions in transition rules with the following ones, which keeps the finiteness.
- In $Post^*$-saturation, a standard pop rule $\psi(\gamma) = \epsilon$ is replaced with $\psi'(\gamma\gamma') = \gamma'$.
- In $Pre^*$-saturation, a simple push rule $\psi(\epsilon) = \gamma$ is replaced with $\psi'(\gamma') = \gamma\gamma'$.

If a PDS is finite (i.e., with finite control states and stack alphabet) and $\mathcal{A}_0$ is a finite automaton, $Post^*(\mathcal{A}_0)$ and $Pre^*(\mathcal{A}_0)$ have bounded numbers of states. (Recall that each newly added state $q_{p,\gamma}$ other than that from $P$ has an index of a pair of a state and a stack symbol, which are finitely many.) Thus, the saturation procedure finitely converges. For a WSPDS with infinite control states and stack alphabet, although $Post^*(\mathcal{A}_0)$ and $Pre^*(\mathcal{A}_0)$ may not finitely converge, they converge as a limit and satisfy Theorem 1. In later sections (Section 4 and 5), we will discuss the finite convergence.

## 4 $Post^*$-automata for coverability

Coverability becomes decidable if either $Post^*$ or $Pre^*$-saturation finitely converges. Karp-Miller acceleration for vectors is proposed for showing decidability of the coverability of a VAS, and generalized in [14, 15] with certain assumptions on a WSTS. In this section we only consider those strictly monotonic WSPDS with *vectors* as stack symbols and without *standard push* rules (only *simple* push rules). Such a PDS is called a *Pushdown Vector Addition Systems* (PDVAS). The reason to exclude *standard push* rules is that $Post^*$-saturation rules for standard push will generate new states, which violates finite convergence.

We write $\mathbb{N}_+$ for $\mathbb{N} \cup \{\omega\}$. Let us fix the dimension $k > 0$ and let $j(\boldsymbol{n})$ be the $j$-th element of a vector $\boldsymbol{n} \in \mathbb{N}_+^k$. $\mathbf{0}$ is the zero-vector with $j(\mathbf{0}) = 0$ for each $j \leq k$. For $J \subseteq [1..k]$, we define the following orderings on vectors:

– $\boldsymbol{n} <_J \boldsymbol{n}'$ if $j(\boldsymbol{n}) < j(\boldsymbol{n}')$ for $j \in J$ and $j(\boldsymbol{n}) = j(\boldsymbol{n}')$ for $j \notin J$.
– $\boldsymbol{n} \leq_J \boldsymbol{n}'$ if $j(\boldsymbol{n}) \leq j(\boldsymbol{n}')$ for $j \in J$ and $j(\boldsymbol{n}) = j(\boldsymbol{n}')$ for $j \notin J$.
– $\boldsymbol{n}_1 \cdots \boldsymbol{n}_l \leqq_J \boldsymbol{n}'_1 \cdots \boldsymbol{n}'_{l'}$ if $l = l'$ and $\forall i. \boldsymbol{n}_i \leq_J \boldsymbol{n}'_i$.
– $\boldsymbol{n}_1 \cdots \boldsymbol{n}_l \ll_J \boldsymbol{n}'_1 \cdots \boldsymbol{n}'_{l'}$ if $\boldsymbol{n}_1 \cdots \boldsymbol{n}_l \leqq_J \boldsymbol{n}'_1 \cdots \boldsymbol{n}'_{l'}$ and $\exists i. \boldsymbol{n}_i <_J \boldsymbol{n}'_i$.

For example, $(1,2) <_{\{2\}} (1,3)$, and $(1,2) \leq_{\{1,2\}} (1,3)$. $(1,2)(1,1) \leqq_{\{1,2\}} (1,3)(1,1)$, but $(1,2)(1,1) \not\ll_{\{1,2\}} (1,3)(1,1)$. We may notice that $<_J, \leq_J, \leqq_J, \ll_J$ are $<, \leq, \leqq$ and $\ll$ on vectors if $J = \{1..k\}$, and are $=$ if $J = \emptyset$.

When $\boldsymbol{n} <_J \boldsymbol{n}'$ , an *acceleration* $\boldsymbol{n} \upharpoonright \boldsymbol{n}' = \boldsymbol{n}_J^{\uparrow}$ where $j(\boldsymbol{n}_J^{\uparrow}) = \omega$ if $j \in J$, and $j(\boldsymbol{n}_J^{\uparrow}) = j(\boldsymbol{n})$ otherwise. For example, $(1,2) \upharpoonright (2,2) = (1,2)_{\{1\}}^{\uparrow} = (\omega, 2)$.

**Definition 6.** *Fix $k \in \mathbb{N}$. A* Pushdown Vector Addition Systems *(PDVAS) is a monotonic WSPDS $\langle P, (\mathbb{N}^k, \leq), \Delta \rangle$ where*

- *$P$ is finite.*
- *$\Delta \in P \times P \times \mathcal{P}Fun((\mathbb{N}^k)^{\leq 2}, \mathbb{N}^k)$ is finite and without standard push rules.*
- *$\psi$ is* strictly monotonic *wrt $\ll_J$ for each rule $(p, q, \psi) \in \Delta$ and $J \subseteq [1..k]$.*

Strict monotonicity wrt $\ll_J$ is crucial for acceleration, which naturally holds in VASs. A VAS transition $\boldsymbol{n} \hookrightarrow \boldsymbol{n} + \boldsymbol{z}$ holds $\boldsymbol{n}' \hookrightarrow \boldsymbol{n}' + \boldsymbol{z} >_J \boldsymbol{n} + \boldsymbol{z}$ for any $\boldsymbol{n}' >_J \boldsymbol{n}$. A WSPDS may have a non-standard pop rule $(p, \boldsymbol{n}_1\boldsymbol{n}_2 \to q, \boldsymbol{m})$, and we require that the growth of either $\boldsymbol{n}_1$ or $\boldsymbol{n}_2$ leads the growth of $\boldsymbol{m}$.

### 4.1 Dependency

The acceleration for VAS occurs when a descendant is strictly larger than some of its ancestors. However, for a WSPDS, such descendant-ancestor relation is not obvious in a P-automaton. We solve this by introducing a relation on P-automata transitions called *dependency* $\Rrightarrow$, which is generated during *Post*$^*$-saturation.

**Definition 7.** *For a PDS $\langle P, \Gamma, \Delta \rangle$, a* dependency *$\Rrightarrow$ over transitions of a Post$^*$-automaton is generated during the saturation procedure, starting from $\emptyset$.*

1. *If a transition $p' \overset{\beta}{\mapsto} q$ is added from a rule $(p, \alpha \to p', \beta)$ and transition $p \overset{\alpha}{\mapsto} q$, then $(p \overset{\alpha}{\mapsto} q) \Rrightarrow (p' \overset{\beta}{\mapsto} q)$.*
2. *If a transition $p' \overset{\gamma}{\mapsto} q$ is added from a rule $(p, \alpha\beta \to p', \gamma)$ and transitions $p \overset{\alpha}{\mapsto} q' \overset{\beta}{\mapsto} q$, then $(p \overset{\alpha}{\mapsto} q') \Rrightarrow (p' \overset{\gamma}{\mapsto} q)$ and $(q' \overset{\beta}{\mapsto} q) \Rrightarrow (p' \overset{\gamma}{\mapsto} q)$.*
3. *Otherwise, we do not update $\Rrightarrow$.*

We denote the reflexive transitive closure of $\Rrightarrow$ by $\Rrightarrow^*$. Strict monotonicity leads to the following lemma which guarantees the soundness of accelerations.

**Lemma 3.** *For a Post$^*$-automaton $\mathcal{A}$ of a PDVAS, if $p \overset{\boldsymbol{n}}{\mapsto} q \Rrightarrow^* p' \overset{\boldsymbol{m}}{\mapsto} q'$ and $p \overset{\boldsymbol{n}'}{\mapsto} q \in \nabla(\mathcal{A})$ where $\boldsymbol{n}' >_J \boldsymbol{n}$, then there exists some $\boldsymbol{m}' >_J \boldsymbol{m}$ such that $p' \overset{\boldsymbol{m}'}{\mapsto} q' \in \nabla(\mathcal{A})$ and $p \overset{\boldsymbol{n}'}{\mapsto} q \Rrightarrow^* p' \overset{\boldsymbol{m}'}{\mapsto} q'$.*

If $(p \overset{\boldsymbol{n}}{\mapsto} q) \Rrightarrow^* (p \overset{\boldsymbol{n}_1}{\mapsto} q)$ and $\boldsymbol{n} <_J \boldsymbol{n}_1$, then applying Lemma 3 we get

$$(p \overset{\boldsymbol{n}}{\mapsto} q) \Rrightarrow^* (p \overset{\boldsymbol{n}_1}{\mapsto} q) \Rrightarrow^* (p \overset{\boldsymbol{n}_2}{\mapsto} q) \Rrightarrow^* \cdots \Rrightarrow^* (p \overset{\boldsymbol{n}_i}{\mapsto} q) \Rrightarrow^* \cdots$$

where $\boldsymbol{n}_i <_J \boldsymbol{n}_{i+1}$ for each $i$. Thus, we can safely apply the acceleration on $J$.

### 4.2 $Post^*_F$-saturation

As discussed in Section 4.1, accelerations will happen if $p \overset{\boldsymbol{n}}{\mapsto} q \Rrightarrow^* p \overset{\boldsymbol{n}'}{\mapsto} q$ and $\boldsymbol{n} <_J \boldsymbol{n}'$ for some $p, q$ and $J$ during the *Post*$^*$-saturation. We introduce dependency relation and accelerations into the post saturation rules for PDVAS. This new saturation procedure and constructed P-automaton are denoted by $Post^*_F$-saturation and $Post^*_F$-automaton, respectively.

We conservatively extend $\psi$ in a PDVAS, from $(\mathbb{N}^k)^{\leq 2} \to \mathbb{N}^k$ to $(\mathbb{N}^k_+)^{\leq 2} \to \mathbb{N}^k_+$. For any $\widetilde{\boldsymbol{n}} \in (\mathbb{N}^k_+)^{\leq 2}$, $\psi(\widetilde{\boldsymbol{n}}) = sup\{\psi(\widetilde{\boldsymbol{n}}') \mid \widetilde{\boldsymbol{n}}' \in (\mathbb{N}^k)^{\leq 2}, \widetilde{\boldsymbol{n}}' \lll \widetilde{\boldsymbol{n}}\}$.

**Definition 8.** *For a PDVAS $\langle P, (\mathbb{N}^k, \leq), \Delta \rangle$, let $\mathcal{A}_0 = (S_0, (\mathbb{N}^k_+, \leq), (\nabla_0, \emptyset), F)$ be an initial P-automaton accepting $C_0$. $Post^*_F(\mathcal{A}_0)$ is the result of repeated applications of the following $Post^*_F$ saturation rules.*

$$\frac{(S, \Gamma, (\nabla, \Rrightarrow), F), \quad p \overset{\widetilde{\boldsymbol{n}}}{\Longmapsto} q}{(S \cup \{p'\}, \Gamma, (\nabla, \Rrightarrow) \oplus (p' \overset{\boldsymbol{n}}{\mapsto} q, \Rrightarrow'), F)} \quad (p, p', \psi) \in \Delta,\ \psi(\widetilde{\boldsymbol{n}}) = \boldsymbol{n}$$

*where $\Rrightarrow'$ is the dependency newly added by Definition 7.*[4] *The operation $\oplus$ is defined as* $(\nabla, \Rrightarrow) \oplus (p' \overset{\boldsymbol{n}}{\mapsto} q, \Rrightarrow') =$

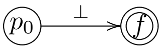
$$\begin{cases} (\nabla \cup \{p' \overset{\boldsymbol{n}' \upharpoonright \boldsymbol{n}}{\longmapsto} q\}, \Rrightarrow \cup \Rrightarrow'') & \textit{if } \exists p' \overset{\boldsymbol{n}'}{\mapsto} q \Rrightarrow^* \cdot \Rrightarrow' p' \overset{\boldsymbol{n}}{\mapsto} q \wedge \boldsymbol{n}' <_J \boldsymbol{n} \textit{ for some } J \\ (\nabla \cup \{p' \overset{\boldsymbol{n}}{\mapsto} q\}, \Rrightarrow \cup \Rrightarrow') & \text{otherwise} \end{cases}$$

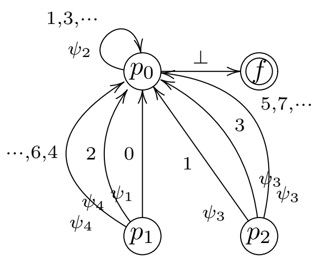
*where $\Rrightarrow''$ is obtained from $\Rrightarrow'$ by replacing the destination $(p' \overset{\boldsymbol{n}}{\mapsto} q)$ with $(p' \overset{\boldsymbol{n}' \upharpoonright \boldsymbol{n}}{\longmapsto} q)$. $A \Rrightarrow^* \cdot \Rrightarrow' B$ means there exists $C$ s.t. $A \Rrightarrow^* C \Rrightarrow' B$.*

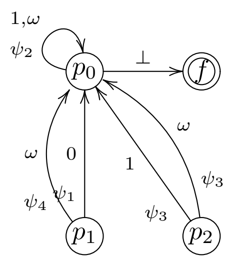
*Example 1.* The following figure shows the infinite $Post^*$-automaton $\mathcal{A}'$ and converged $Post^*_F$-automaton $\mathcal{A}$ of a PDVAS with transition rules $\psi_{1,2,3,4}$. We start from $C_0 = \{\langle p_0, \bot \rangle\}$. In $\mathcal{A}'$, $p_2 \overset{1}{\mapsto} p_0$ is generated from $p_1 \overset{0}{\mapsto} p_0 \overset{1}{\mapsto} p_0$ by transition rule $\psi_3$, and $p_1 \overset{2}{\mapsto} p_0$ is generated from $p_2 \overset{1}{\mapsto} p_0$ by transition $\psi_4$.

In $\mathcal{A}$, we have $(p_1 \overset{0}{\mapsto} p_0) \Rrightarrow (p_2 \overset{1}{\mapsto} p_0) \Rrightarrow (p_1 \overset{2}{\mapsto} p_0)$. Therefore we can apply acceleration and $(p_1 \overset{\omega}{\mapsto} p_0)$ is added instead of $(p_1 \overset{2}{\mapsto} p_0)$. Then $p_2 \overset{\omega}{\mapsto} p_0$ and $p_0 \overset{\omega}{\mapsto} p_0$ is added according to transition rule $\psi_3$ and $\psi_2$ respectively. This leads to the finitely converged $Post^*_F$ automaton $\mathcal{A}$, and $(post^*(C_0))^{\downarrow} = L(\mathcal{A})^{\downarrow} \cap (\mathbb{N}^k)^*$.

$\psi_1 : p_0, \epsilon \to p_1, 0$
$\psi_2 : p_1, \boldsymbol{n} \to p_0, \boldsymbol{n} + 1$
$\psi_3 : p_1, \boldsymbol{n}_1\boldsymbol{n}_2 \to p_2, \boldsymbol{n}_1 + \boldsymbol{n}_2$
$\psi_4 : p_2, \boldsymbol{n} \to p_1, \boldsymbol{n} + 1$

$\mathcal{A}_0$ :



$\mathcal{A}'$ :



$\mathcal{A}$ :



An immediate observation is that $Post^*_F$-saturation is sound, because configurations accepted by $Post^*(\mathcal{A}_0)$ are covered by $Post^*_F(\mathcal{A}_0)$ according to the monotonicity. The completeness follows from Lemma 4, which says the downward closure of any transition in $Post^*_F(\mathcal{A}_0)$ is included in the downward closure of transitions in $Post^*(\mathcal{A}_0)$. We leave the proof in Appendix B.

**Lemma 4.** *Let $\mathcal{A}_0$ be an initial P-automaton of some PDVAS. If $p \overset{\boldsymbol{n}}{\mapsto} q$ in $Post^*_F(\mathcal{A}_0)$, then for any $\boldsymbol{n}' \leq \boldsymbol{n}$ and $\boldsymbol{n}' \in \mathbb{N}^k$ there exists $p \overset{\boldsymbol{n}''}{\mapsto} q$ in $Post^*(\mathcal{A}_0)$ for some $\boldsymbol{n}''$ such that $\boldsymbol{n}' \leq \boldsymbol{n}'' \leq \boldsymbol{n}$.*

[4] $\Rrightarrow' = \emptyset$ if $(p, p', \psi)$ is a push rule; otherwise the destination of each pair in $\Rrightarrow'$ is $p' \overset{\boldsymbol{n}}{\mapsto} q$ with $\boldsymbol{n} = \psi(\widetilde{\boldsymbol{n}})$.

Since PDVASs do not have standard-push rules, the saturation procedure does not add new states. So the states in $Post_F^*(\mathcal{A}_0)$ and $Post^*(\mathcal{A}_0)$ are the same. From Lemma 4, we can obtain $L(Post_F^*(\mathcal{A}_0))^{\downarrow} \cap (\mathbb{N}^k)^* = (post^*(C_0))^{\downarrow}$.

The finite convergence of $Post_F^*$-saturation follows from the fact that $\{(p, \boldsymbol{n}, q) \mid p, q \in S, \boldsymbol{n} \in \mathbb{N}_+^k\}$ is well-quasi-ordered. Each path of the dependency $\Rrightarrow^*$ contains finitely many P-automata transitions, and $\Rrightarrow^*$ is finitely branching, thus by König's lemma, the $\Rrightarrow$-tree is finite. We obtain the following theorem.

**Theorem 2.** *Starting from a finite P-automaton $\mathcal{A}_0$ accepting $C_0$ for any PDVAS,*

- *the $Post_F^*$-saturation procedure finitely converges to $Post_F^*(\mathcal{A}_0)$, and*
- *$L(Post_F^*(\mathcal{A}_0))^{\downarrow} \cap (\mathbb{N}^k)^* = (post^*(C_0))^{\downarrow}$.*

### 4.3 Coverability of RVASS

In this section, we show that Recursive Vector Addition Systems with States (RVASS) are special cases of PDVASs, and our results is directly applied to prove decidability of its coverability. RVASS is proposed by Bouajjani and Emmi [1] for modeling interleaving-insensitive multi-thread programs.

**Definition 9.** *[1] Fix $k \in \mathbb{N}$. A* RVASS *$\langle Q, \delta \rangle$ is a finite set of state $\mathcal{Q}$ along with a finite set of transitions $\delta$. We denote*

- *$q \xrightarrow{\boldsymbol{z}} q'$ if $(q, q', \boldsymbol{z}) \in \delta$ for $\boldsymbol{z} \in \mathbb{Z}^k$, and*
- *$q \xrightarrow{q_1 q_2} q'$ if $(q, q_1, q_2, q') \in \delta$.*

*The configuration $c \in (Q \times \mathbb{N}^k)^*$ represents a stack of pairs $\langle p, \boldsymbol{n} \rangle$ where $p \in Q$ and $\boldsymbol{n} \in \mathbb{N}^k$. The semantics is defined by following rules:*

$$\frac{q \xrightarrow{\boldsymbol{z}} q' \quad \boldsymbol{n} + \boldsymbol{z} \in \mathbb{N}^k}{\langle q, \boldsymbol{n} \rangle c \hookrightarrow \langle q', \boldsymbol{n} + \boldsymbol{z} \rangle c} \qquad \frac{q \xrightarrow{q_1 q_2} q'}{\langle q, \boldsymbol{n} \rangle c \hookrightarrow \langle q_1, \mathbf{0} \rangle \langle q, \boldsymbol{n} \rangle c} \qquad \frac{q \xrightarrow{q_1 q_2} q'}{\langle q_2, \boldsymbol{n'} \rangle \langle q, \boldsymbol{n} \rangle c \hookrightarrow \langle q', \boldsymbol{n} + \boldsymbol{n'} \rangle c}$$

The *state-reachability* problem of an RVASS is to determine given two states, $q_0, q_f$, whether there exist a vector $\boldsymbol{n}$ and a configuration $c$, such that $\langle q_0, \mathbf{0} \rangle \hookrightarrow^* \langle q_f, \boldsymbol{n} \rangle c$. This problem is shown to be decidable in Lemma 3 of [1] by a reduction to the state-reachability of a Branching VASS. Our Corollary 1 shows the decidability of the coverability. Note that the state reachability is regarded as the coverability from $\langle q_0, \mathbf{0} \rangle$ to $\{\langle q_f, \mathbf{0}^+ \rangle\}$.

The encoding from an RVASS to a PDVAS is straightforward by considering the configuration of an RVASS as the stack contents in a PDVAS. Therefore, the three semantic rules of RVASS are simulated by 1, 2(a) and 2(b) in Definition 10.

**Definition 10.** *Given $k \in \mathbb{N}$ and a* RVASS *$R = \langle Q, \delta \rangle$, we define a PDVAS $M_R = (P, \Gamma, \Delta)$ for $R$ where*

- *$P = Q$, and $\Gamma = \mathbb{N}^{|Q|+k}$ where $|Q|$ is the number of states in $Q$, we use $(p, \boldsymbol{n})$ to range over $\Gamma$;*
- *The finite set of transition rules $\Delta \subseteq P \times P \times \mathcal{P}Fun(\Gamma^{\leq 2}, \Gamma)$ consists of*

1. if $(q, q', \boldsymbol{z}) \in \delta$, then $q, (q, \boldsymbol{n}) \to q', (q', \boldsymbol{n}+\boldsymbol{z})$ for all $\boldsymbol{n}$ with $\boldsymbol{n}+\boldsymbol{z} \in \mathbb{N}^k$;
2. if $(q, q_1, q_2, q') \in \delta$, then
   (a) $q, \epsilon \to q_1, (q_1, \boldsymbol{0})$
   (b) $q_2, (q_2, \boldsymbol{n})(q, \boldsymbol{m}) \to q', (q', \boldsymbol{n}+\boldsymbol{m})$ for all $\boldsymbol{n}$ and $\boldsymbol{m}$.

**Lemma 5.** *Assume that $R$ is an RVASS, and $M_R$ is the encoded PDVAS for $R$. Given two configuration $c_0 = \langle p_0, \boldsymbol{n}_0\rangle c'_0$ and $c_f = \langle p_f, \boldsymbol{n}_f\rangle c'_f$ of $R$. Then, $c_0 \hookrightarrow^* c_f$ in $R$ if and only if $\langle p_0, c_0\rangle \hookrightarrow^* \langle p_f, c_f\rangle$ in $M_R$.*

**Corollary 1.** *The coverability of an RVASS is decidable.*

## 5 $\boldsymbol{Pre^*}$-automata for coverability

When $\Delta$ contains only standard pop rules, $Pre^*$ does not introduce new states except from $P$, which leads to the finite convergence by ideal representations. In this section, we assume that $\Delta$ does not contain non-standard pop rules.

### 5.1 Ideal representation of $\boldsymbol{Pre^*}$-automata

As we mentioned in Section 3.2, we need to construct $Pre^*$-automaton for an ideal $C_0^{\uparrow}$ of $C_0$. A naive representation of an initial P-automaton accepting $C_0^{\uparrow}$ may be infinite. Therefore, we use the ideal representation where transition labels and states are ideals. Thanks to WQO, an ideal has a finite representation by its minimal elements, and ideals are well founded wrt set inclusion.

**Definition 11.** *For a monotonic WSPDS $\langle (P, \preceq), (\Gamma, \leq), \Delta\rangle$, as in Definition 3 we define a $Pre^*_F$-automaton $\mathcal{A} = (S, \mathcal{I}(\Gamma), \nabla, F)$ by replacing $\Gamma$ with $\mathcal{I}(\Gamma)$ and $\mathcal{I}(P) \subseteq S$.*

As notational convention, let $s, t$ to range over $S$, ideals $K, K'$ to range over $\mathcal{I}(P)$, and $I, I'$ over $\mathcal{I}(\Gamma)$. We denote $w \in \widetilde{I}$ for $\widetilde{I} = I_1 I_2 \cdots I_n$, if $w = \alpha_1\alpha_2\cdots\alpha_n$ and $\alpha_i \in I_i$ for each $i$. We say that $\mathcal{A}$ accepts a configuration $\langle p, w\rangle$, if there is a path $K \overset{\widetilde{I}}{\Longmapsto} f \in F$ in $\mathcal{A}$ and $p \in K$, $w \in \widetilde{I}$.

The ideal representation of an initial P-automaton accepting $C_0^{\uparrow}$ is obtained from a P-automaton accepting $C_0$ by replacing each state $p$ with $\{p\}^{\uparrow}$ and each transition label $\alpha$ with $\{\alpha\}^{\uparrow}$.

**Definition 12.** *Let $\mathcal{A}_0$ be an initial $Pre^*_F$-automaton accepting $C_0^{\uparrow}$. $Pre^*_F(\mathcal{A}_0)$ is the result of repeated applications of the following $Pre^*_F$-saturation rules*

$$\frac{(S, \mathcal{I}(\Gamma), \nabla, F), \quad K \overset{\widetilde{I}}{\Longmapsto} s}{(S, \mathcal{I}(\Gamma), \nabla, F) \oplus \{\phi^{-1}(K) \overset{\psi^{-1}(\widetilde{I})}{\longmapsto} s\}} \quad \text{if } \widetilde{I} \in \mathcal{I}(\Gamma^{\leq 2}) \text{ and } (\phi, \psi) \in \Delta$$

*where $\phi^{-1}(K) \neq \emptyset$ and $\psi^{-1}(\widetilde{I}) \neq \emptyset$ and $(S, \Sigma, \nabla, F) \oplus \{K \overset{I}{\mapsto} s\}$ is*

$$\begin{cases} (S, \Sigma, \nabla, F) & \text{if } (K' \overset{I'}{\mapsto} s) \in \nabla \text{ with } K \subseteq K' \text{ and } I \subseteq I' \\ (S, \Sigma, (\nabla \setminus \{K \overset{I'}{\mapsto} s\}) \cup \{K \overset{I' \cup I}{\mapsto} s\}, F) & \text{if } (K \overset{I'}{\mapsto} s) \in \nabla. \\ (S \cup \{K\}, \Sigma, \nabla \cup \{K \overset{I}{\mapsto} s\}, F) & \text{otherwise} \end{cases}$$

The $\oplus$ operator merges ideals associated to transitions. Assume we generate a new transition $K \overset{I}{\mapsto} s$. If there is transition $K' \overset{I'}{\mapsto} s$ for the same $s$ with $K \subseteq K'$ and $I \subseteq I'$, the ideal of configurations starting from $K \overset{I}{\mapsto} s$ is included in that from $K' \overset{I'}{\mapsto} s$. Thus, we do not add it. If there is transition $K \overset{I'}{\mapsto} s$ between the same pair $K, s$, then take the union $I \cup I'$. Otherwise, we add a new transition.

It is easy to see that if $\phi \in \mathcal{P}Fun(X, Y)$ is monotonic, then, for any $I \in \mathcal{I}(Y)$, $\phi^{-1}(I)$ is an ideal in $\mathcal{I}(X)$. Soundness $pre^*(C_0^{\uparrow}) \subseteq L(Pre_F^*(\mathcal{A}_0))$ follows immediately by induction on saturation steps. Completeness $pre^*(C_0^{\uparrow}) \supseteq L(pre^*(\mathcal{A}_0))$ is guaranteed by Lemma 6, which is an invariant during the saturation procedure.

**Lemma 6.** *Assume* $K \overset{\widetilde{I}}{\Longmapsto} s$ *in* $Pre_F^*(\mathcal{A}_0)$*. For each* $p \in K$, $w \in \widetilde{I}$,

- *if* $s = K' \in \mathcal{I}(P)$*, then* $\langle p, w\rangle \hookrightarrow^* \langle q, \epsilon\rangle$ *for some* $q \in K'$*.*
- *if* $s \notin \mathcal{I}(P)$*, there exists* $K' \overset{\widetilde{I'}}{\Longmapsto} s$ *in* $\mathcal{A}_0$ *such that* $\langle p, w\rangle \hookrightarrow^* \langle p', w'\rangle$ *for some* $p' \in K'$ *and* $w' \in \widetilde{I'}$*.*

**Theorem 3.** *For an initial P-automaton* $\mathcal{A}_0$ *accepting* $C_0^{\uparrow}$, $L(Pre_F^*(\mathcal{A}_0)) = pre^*(C_0^{\uparrow})$*.*

Note that Thereom 3 only shows the correctness of $Pre_F^*$-saturation. We will discuss its finite convergence in next two subsections.

### 5.2 Coverability of Multi-set PDS

As an example of the finite convergence, we show *Multi-set pushdown system* (Multi-set PDS) proposed by [2, 3], which is an extension of PDS by attaching a multi-set into the configuration. We directly give the definition of a Multi-set PDS as a WSPDS. Note that, although a Multi-set PDS has infinitely many control states, it finitely converges because of restrictions on decreasing rules.

**Definition 13.** *A* Multi-set pushdown system (Multi-set PDS) *is a WSPDS* $((Q \times \mathbb{N}^k, \preceq), \Gamma, \delta)$*, where*

- $Q$*,* $\Gamma$ *are finite and* $k = |\Gamma|$*,*
- $\delta$ *is a finite set of transition rules consisting of two kinds:*
  1. *Increasing rules* $\delta_1 : (p, \gamma, q, w, \boldsymbol{n})$ *for* $\boldsymbol{n} \in \mathbb{N}^k$*;*
  2. *Decreasing rules* $\delta_2$*:* $(p, \bot, q, \bot, \boldsymbol{n})$ *for* $\boldsymbol{n} \in \mathbb{N}^k$*.*

*Configuration transitions are defined by:*

$$\frac{(p, \gamma, q, w, \boldsymbol{n}) \in \delta_1}{\langle (p, \boldsymbol{m}), \gamma w'\rangle \hookrightarrow \langle (q, \boldsymbol{n} + \boldsymbol{m}), ww'\rangle} \qquad \frac{(p, \bot, q, \bot, \boldsymbol{n}) \in \delta_2, \boldsymbol{m} \geq \boldsymbol{n}}{\langle (p, \boldsymbol{m}), \bot\rangle \hookrightarrow \langle (q, \boldsymbol{m} - \boldsymbol{n}), \bot\rangle}$$

Note the decreasing rules are applied only when the stack is empty. A state in $Pre_F^*$-automata is in $\mathcal{I}(Q \times \mathbb{N}^k)$. Since $Q$ is finite, we can always separate one state into finitely many states, each of which is in the form of $Q \times \mathcal{I}(\mathbb{N}^k)$. From Definition 12, we have two observations.

1. If transition $(p, I) \overset{\gamma}{\mapsto} s$ is added from $(q, I') \overset{w}{\Longmapsto} s$ and some increasing rule in $\delta_1$, then $I \supseteq I'$.
2. If transition $(p, I) \overset{\perp}{\mapsto} s$ is added from $(q, I') \overset{\perp}{\mapsto} s$ and some decreasing rule in $\delta_2$, then $I \subseteq I'$ and $s$ is a final state.

There are only finitely many final states in $\mathcal{A}_0$ and $Q$ is finite. Thus, by the definition of $\oplus$ operator and Lemma 1, there are finitely many states $(p, I)$ adjacent to final states paired with $\perp$ in a $Pre^*_F$-automaton. Other states added by decreasing rules are also finitely many by observation 1 and Lemma 1. Therefore, we have i) the total states of the converged $Pre^*_F$-automaton is finite and ii) the labels between pairs of states are finite ($\Gamma$ is finite). The decidable coverability of Multi-set PDS is a corollary of Theorem 3 .

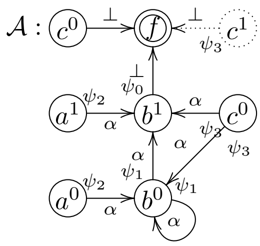
**Corollary 2.** *The* coverability problem *for a Multi-set PDS is decidable.*

*Example 2.* Let $\langle(\{a, b, c\} \times \mathbb{N}, \preceq), \{\alpha\}, \delta\rangle$ be an Multi-set PDS with transition rules given in the following graph. The set of configurations covering $\langle c^0, \perp\rangle$ is computed by $Pre^*_F$-automaton $\mathcal{A}$. We abbreviate ideal $\{p^n\}^\uparrow$ by $p^n$ for $p \in \{a, b, c\}$ and $n \geq 0$.

Transition $c^1 \overset{\perp}{\mapsto} f$ is generated from $a^1 \overset{\alpha\perp}{\Longmapsto} f$ by applying $\psi_3$ It is omitted because we already have $c^0 \overset{\perp}{\mapsto} f$ and $\{c^1\}^\uparrow \subseteq \{c^0\}^\uparrow$.

$$\boxed{\begin{aligned} \delta_1 = \{\ & \psi_1 : (b^{\boldsymbol{n}}, \alpha \to a^{\boldsymbol{n}+1}, \alpha), \\ & \psi_2 : (a^{\boldsymbol{n}}, \alpha \to b^{\boldsymbol{n}}, \epsilon), \\ & \psi_3 : (c^{\boldsymbol{n}}, \epsilon \to a^{\boldsymbol{n}}, \alpha)\} \\ \delta_2 = \{\ & \psi_0 : (b^{\boldsymbol{n}}, \perp \to c^{\boldsymbol{n}-1}, \perp)\} \end{aligned}}$$

$$\mathcal{A}_0 : c^0 \xrightarrow{\perp} f$$




### 5.3 Finite control states

Assume that, for a monotonic WSPDS $M = \langle P, (\Gamma, \leq), \Delta\rangle$, $P$ is finite and $\Delta$ does not contain non-standard pop rules. Then, we observe that, in the $Pre^*_F$-saturation for $M$, i) the set of states is bounded by the state in $\mathcal{A}_0$ and $P$, and ii) transitions between any pair of states are finitely many by Lemma 1. Hence, $Pre^*_F$ saturation procedure finitely converges.

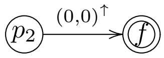
**Theorem 4.** *Assume that $M = \langle P, (\Gamma, \leq), \Delta\rangle$ is a monotonic WSPDS, $P$ is finite, and $\psi^{-1}(I)$ is computable for any $(p, p', \psi) \in \Delta$. Then, the coverability of $M$ is decidable.*

*Example 3.* Let $M = \langle\{p_i\}, \mathbb{N}^2, \Delta\rangle$ be a monotonic WSPDS with $\Delta$ consists of four rules given in the figure. Automaton $\mathcal{A}$ illustrates the $pre^*$-saturation starting from initial $\mathcal{A}_0$ that accepts $C = \langle p_2, (0, 0)^\uparrow\rangle$.

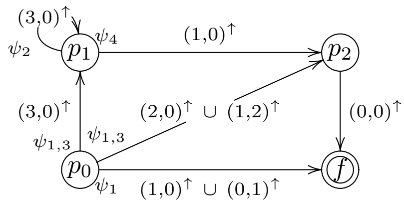
For instance, $p_1 \overset{(3,0)^\uparrow}{\longmapsto} p_1$ in $\mathcal{A}$ is generated by $\psi_2$, and $p_0 \overset{(3,2)^\uparrow}{\longmapsto} p_1$ is added by $\psi_3$. Then repeatedly apply $\psi_1$ twice to $p_0 \overset{(3,2)^\uparrow}{\longmapsto} p_1 \overset{(3,0)^\uparrow}{\longmapsto} p_1$, we obtain $p_0 \overset{(3,0)^\uparrow}{\mapsto} p_1$.

$$\boxed{\begin{array}{l}
\psi_1 : \langle p_0, \boldsymbol{n}\rangle \to \langle p_0, (\boldsymbol{n} + (1,1))\boldsymbol{n}\rangle \\
\psi_2 : \langle p_1, \boldsymbol{n}\rangle \to \langle p_1, \epsilon\rangle \text{ if } \boldsymbol{n} \geq (3,0) \\
\psi_3 : \langle p_0, \boldsymbol{n}\rangle \to \langle p_1, \boldsymbol{n} - (0,2)\rangle \text{ if } \boldsymbol{n} \geq (0,2) \\
\psi_4 : \langle p_1, \boldsymbol{n}\rangle \to \langle p_2, \epsilon\rangle \text{ if } \boldsymbol{n} \geq (1,0)
\end{array}}$$

$\mathcal{A}_0$ :




$\mathcal{A}$ :




## 6 Conclusion

This paper investigated *well-structured pushdown systems* (WSPDSs), pushdown systems with well-quasi-ordered control states and stack alphabet, and developed two proof techniques to investigate the coverability based on extensions of P-automata techniques. They are,

- when a WSPDS has no standard push rules, the forward P-automata construction $Post^*$ with Karp-Miller acceleration, and
- when a WSPDS has no non-standard pop rules, the backward P-automata construction $Pre^*$ with ideal representations.

We show decidability results of coverability, which include *recursive vector addition system with states* [1], *multi-set pushdown systems* [2, 3], and a WSPDS with finite control states and WQO stack alphabet. The first one extends the decidability of the state reachability in [1] to that of the coverability.

Our current results just opened the possibility of WSPDSs. Among lots of things to do, we list two for future works. Our decidability proofs contain algorithms to compute, however the estimation of their complexity is not easy, since we rely their termination on WQO arguments. We hope that a general theoretical observation [21] would give hints for complexity estimation. Our current forward method is restricted to VASs. We also hope to apply Finkel and Goubault-Larrecq's work on $\omega^2$-WSTS [14, 15] to generalize.


## Acknowledgements

The authors would like to thank Prof.Alain Finkel for valuable comments at the oral presentation in an early stage of this research.


## References


1. Bouajjani, A., Emmi, M.: Analysis of recursively parallel programs. In: Principles of Programming Languages (POPL'12), ACM (2012) 203–214

2. Sen, K., Viswanathan, M.: Model checking multithreaded programs with asynchronous atomic methods. In: Computer Aided Verification (CAV'06). Volume 4144 of LNCS., Springer (2006) 300–314
3. Jhala, R., Majumdar, R.: Interprocedural analysis of asynchronous programs. In: Principles of Programming Languages (POPL'07). ACM (2007) 339–350
4. Abdulla, P., Cerans, K., Jonsson, C., Yih-Kuen, T.: Algorithmic analysis of programs with well quasi-ordered domains. Information and Computation **160**(1–2) (2000) 109–127
5. Finkel, A., Schnoebelen, P.: Well-structured transition systems everywhere! Theoretical Computer Science **256**(1–2) (2001) 63–92
6. Atig, M., Bouajjani, A., Qadeer, S.: Context-bounded analysis for concurrent programs with dynamic creation of threads. In: Tools and Algorithms for the Construction and Analysis of Systems (TACAS'09). LNCS, Springer (2009) 107–123
7. Ramalingam, G.: Context-sensitive synchronization-sensitive analysis is undecidable. ACM Transactions on Programming Languages and Systems **22**(2) (2000) 416–430
8. Qadeer, S., Rehof, J.: Context-bounded model checking of concurrent software. In: Tools and Algorithms for the Construction and Analysis of Systems (TACAS05). LNCS, Springer (2005) 93–107
9. Büchi, J.: Regular canonical systems. Archive for Mathematical Logic **6**(3) (1964) 91–111
10. Finkel, A., Willems, B., Wolper, P.: A direct symbolic approach to model checking pushdown systems. Electronic Notes Theoretical Computer Science **9**(1997) 27–37
11. Bouajjani, A., Esparza, J., Maler, O.: Reachability analysis of pushdown automata: Application to model-checking. In: Concurrency Theory (CONCUR'97). LNCS, Springer (1997) 135–150
12. Esparza, J., Hansel, D., Rossmanith, P., Schwoon, S.: Efficient algorithms for model checking pushdown systems. In: Computer Aided Verification (CAV'00). LNCS, Springer (2000) 232–247
13. Finkel, A.: A generalization of the procedure of karp and miller to well structured transition systems. In: International Colloquium on Automata, Languages and Programming (ICALP'87). (1987) 499–508
14. Finkel, A., Goubault-Larrecq, J.: Forward analysis for WSTS, Part I: Completions. In: International Symposium on Theoretical Aspects of Computer Science (STACS'09). (2009) 433–444
15. Finkel, A., Goubault-Larrecq, J.: Forward analysis for WSTS, Part II: Complete WSTS. In: International Colloquium on Automata, Languages and Programming (ICALP'09), Springer (2009) 188–199
16. Mayr, E.: An algorithm for the general petri net reachability problem. SIAM Journal Computing **13**(3) (1984) 441–460
17. Leroux, J.: Vector addition system reachability problem. In: Principles of Programming Languages (POPL'11), ACM (2011) 307–316
18. Mayr, R.: Process rewrite systems. Information and Computation **156** (1999) 264–286
19. Verma, K., Goubault-Larrecq, J.: Karp-Miller trees for a branching extension of VASS. Discrete Mathematics & Theoretical Computer Science **7**(1)(2005)217–230
20. Chadha, R., Viswanathan, M.: Decidability results for well-structured transition systems with auxiliary storage. In: Concurrency Theory (CONCUR'07). Volume 4703 of LNCS., Springer (2007) 136–150
21. Weiermann, A.: Complexity bounds for some finite form of Kruskal's theorem. Journal of Symbolic Computation **18** (1994) 463–488

# A Proof of Lemma 2

We prove Lemma 2', which generalizes the statement in Lemma 2 from $p \in P$ to $p \in P \cup Q$, where $Q = \{q_{p,\alpha} \mid p \in P, \alpha \in \Gamma\}$. Let $w \in \Gamma^*$. We define the following function from $P \cup Q$ to $P \times \Gamma^*$ for notational convenience.

$$con_w(p) = \begin{cases} \langle p, w \rangle & \text{if } p \in P \\ \langle p', \alpha w \rangle & \text{if } p = q_{p',\alpha} \in Q \end{cases}$$

**Lemma 2'.** Let $\langle P, \Gamma, \Delta \rangle$ be a PDS, and let $\mathcal{A}_0$ be an initial P-automaton accepting $C_0$. Assume that $p \overset{w}{\Longmapsto} q$ in $Post^*(\mathcal{A}_0)$.

1. If $q \in P \cup Q$, $con_\epsilon(q) \hookrightarrow^* con_w(p)$;
2. If $q \in S(\mathcal{A}_0) \setminus P$, there exists $q' \overset{v}{\Longmapsto} q$ in $\mathcal{A}_0$ with $q' \in P$ and $\langle q', v \rangle \hookrightarrow^* con_w(p)$.

*Proof.* By induction on steps of the saturation procedure $\mathcal{A}_0, \mathcal{A}_1, \mathcal{A}_2, \cdots$. For $\mathcal{A}_0$, the statement holds immediately. Assume above statements hold for $\mathcal{A}_i$, and $\mathcal{A}_{i+1}$ is constructed by adding new transitions (denoted by $\mapsto_{i+1}$) by either of the following rules. We also denote the transitions in $\mathcal{A}_0$ by $\mapsto_0$ and $(\cup_{j \le i} \mapsto_j)^*$ by $\longmapsto_i$, respectively.

$$\frac{(S, \Gamma, \nabla, F), \quad p' \overset{v}{\Longmapsto} q}{(S \cup \{p\}, \Gamma, \nabla \cup \{p \overset{\gamma}{\mapsto} q\}, F)} \quad (p', v \to p, \gamma) \in \Delta, |v| \le 2 \tag{1}$$

$$\frac{(S, \Gamma, \nabla, F), \quad p' \overset{\gamma}{\mapsto} q}{(S \cup \{p, q_{p,\alpha}\}, \Gamma, \nabla \cup \{p \overset{\alpha}{\mapsto} q_{p,\alpha} \overset{\beta}{\mapsto} q\}, F)} \quad (p', \gamma \to p, \alpha\beta) \in \Delta \tag{2}$$

Let $p_0 \overset{w}{\Longmapsto} q_0$ be a path in $\mathcal{A}_{i+1}$ with $p_0 \in P \cup Q$. Assume that $p_0 \overset{w}{\Longmapsto} q_0$ contains $\mapsto_{i+1}$ $k$-times. We prove by (nested) induction on $k$. If $k = 0$, obvious. Let $k > 0$ and let the leftermost occurence of $\mapsto_{i+1}$ in $p_0 \overset{w}{\Longmapsto} q_0$ be $p'' \overset{\delta}{\mapsto}_{i+1} q''$. Thus,

$$p_0 \overset{w_1}{\Longmapsto}_i p'' \overset{\delta}{\mapsto}_{i+1} q'' \overset{w_2}{\Longmapsto}_{i+1} q_0$$

where $q'' \overset{w_2}{\Longmapsto}_{i+1} q_0$ contains $\mapsto_{i+1}$ at most $k - 1$ times.

We will prove only the statement 1. ($q \in P \cup Q$) in Lemma 2‘, since the statement 2. ($q \in S(\mathcal{A}_0) \setminus P$) follows similarly. We have three cases.

1. The rule (1) is used, and $p'' \in P$.
2. The rule (2) is used, $p'' \in P$, and $q'' \in Q$.
3. The rule (2) is used, $p'' \in Q$.

**Case 1.** Following to the notation of the rule 1, let $p'' = p$, $q'' = q$, and $\delta = \gamma$. By induction hypothesis on $p_0 \overset{w_1}{\Longmapsto}_i p$ and $p' \overset{v}{\Longmapsto}_i q \overset{w_2}{\Longmapsto}_{i+1} q_0$, we have $\langle p, \epsilon \rangle \hookrightarrow^* con_{w_1}(p_0)$ and $con_\epsilon(q_0) \hookrightarrow^* \langle p', vw_2 \rangle$, respectively. By the rule 1, we have $\langle p', v \rangle \hookrightarrow \langle p, \gamma \rangle$. Thus,

$$con_\epsilon(q_0) \hookrightarrow^* \langle p', vw_2 \rangle \hookrightarrow \langle p, \gamma w_2 \rangle \hookrightarrow^* con_{w_1 \gamma w_2}(p_0)$$

**Case 2.** Following to the notation of the rule 2, let $p'' = p$, $q'' = q_{p,\alpha}$, and $\delta = \alpha$. By induction hypothesis on $p_0 \xmapsto{w_1}_i p$ and $q_{p,\alpha} \xmapsto{w_2}_{i+1} q_0$, we have $\langle p, \epsilon\rangle \hookrightarrow^* con_{w_1}(p_0)$ and $con_\epsilon(q_0) \hookrightarrow^* \langle p, \alpha w_2\rangle$, respectively. Thus,

$$con_\epsilon(q_0) \hookrightarrow^* \langle p, \alpha w_2\rangle \hookrightarrow^* con_{w_1 \alpha w_2}(p_0)$$

**Case 3.** Following to the notation of the rule 2, let $p'' = q_{p,\alpha}$, $q'' = q$, and $\delta = \beta$. By induction hypothesis on $p_0 \xmapsto{w_1}_i q_{p,\alpha}$ and $p' \overset{\gamma}{\mapsto}_i q \xmapsto{w_2}_{i+1} q_0$, we have $\langle p, \alpha\rangle \hookrightarrow^* con_{w_1}(p_0)$ and $con_\epsilon(q_0) \hookrightarrow^* \langle p', \gamma w_2\rangle$, respectively. By the rule 2, we have $\langle p', \gamma\rangle \hookrightarrow \langle p, \alpha\beta\rangle$. Thus,

$$con_\epsilon(q_0) \hookrightarrow^* \langle p', \gamma w_2\rangle \hookrightarrow \langle p, \alpha\beta w_2\rangle \hookrightarrow^* con_{w_1 \beta w_2}(p_0)$$

□

## B Proof of Lemma 4

To prove Lemma 4, we first prove Lemma 7 which indicates the relationship between dependency pairs in $Post^*_F$-automaton and those in $Post^*$-automaton. For denotational convenience, we will use $\boldsymbol{u}, \boldsymbol{v}$ to denote vectors in $\mathbb{N}^k$ and $\boldsymbol{n}, \boldsymbol{m}$ generally for those in $\mathbb{N}^k_+$. We write $\omega(\boldsymbol{n})$ for $\{j \mid j(\boldsymbol{n}) = \omega\}$.

**Lemma 7.** *Let $\mathcal{A}_0$ be an initial P-automaton of some PDVAS. Assume $p' \overset{\boldsymbol{n}'}{\mapsto} q' \Rrightarrow^* p \overset{\boldsymbol{n}}{\mapsto} q$ in $Post^*_F(\mathcal{A}_0)$, then there exists some $\boldsymbol{v}' <_{\omega(\boldsymbol{n}')} \boldsymbol{n}'$ such that $\forall \boldsymbol{v}_1 <_{\omega(\boldsymbol{n})} \boldsymbol{n}.\ \exists \boldsymbol{v}_2 \geq_{\omega(\boldsymbol{n})} \boldsymbol{v}_1$, and $p' \overset{\boldsymbol{v}'}{\mapsto} q' \Rrightarrow^* p \overset{\boldsymbol{v}_2}{\mapsto} q$ in $Post^*(\mathcal{A}_0)$.*

*Proof.* We proceed by induction on the $Post^*_F$-saturation of $\mathcal{A}_0, \mathcal{A}_1, \mathcal{A}_2, \cdots$. We will write $\Rrightarrow_i$ for dependency relation in $\mathcal{A}_i$. Since $\Rrightarrow_0 = \emptyset$, so the lemma holds for $\mathcal{A}_0$. Assume that $\mathcal{A}_{i+1}$ is constructed by adding a new transition $p \overset{\boldsymbol{n}}{\mapsto} q$. We only consider the acceleration case while applying nonstandard pop rule. Cases without acceleration and with acceleration for internal rule are similar and simpler.

Assume we have nonstandard pop rule $\langle p', \boldsymbol{m}_1\boldsymbol{m}_2 \to p, \boldsymbol{n}'\rangle$, the transition $p \overset{\boldsymbol{n}}{\mapsto} q$ is added from $p' \overset{\boldsymbol{m}_1}{\mapsto} q' \overset{\boldsymbol{m}_2}{\mapsto} q$ and there exist some $\boldsymbol{n}'' <_J \boldsymbol{n}'$, such that $p \overset{\boldsymbol{n}''}{\mapsto} q \Rrightarrow^*_i p' \overset{\boldsymbol{m}_1}{\mapsto} q_1$ or $p \overset{\boldsymbol{n}''}{\mapsto} q \Rrightarrow^*_i q_1 \overset{\boldsymbol{m}_2}{\mapsto} q$, and $\boldsymbol{n} = \boldsymbol{n}'' \upharpoonright \boldsymbol{n}'$. By the definition of $\oplus$ in Definition 8, we add $p \overset{\boldsymbol{n}}{\mapsto} q$ instead of $p \overset{\boldsymbol{n}'}{\mapsto} q$ and

$$\Rrightarrow_{i+1} = \Rrightarrow_i \cup \{(p' \overset{\boldsymbol{m}_1}{\mapsto} q_1, p \overset{\boldsymbol{n}}{\mapsto} q), (q_1 \overset{\boldsymbol{m}_2}{\mapsto} q, p \overset{\boldsymbol{n}}{\mapsto} q)\}.$$

For any $p_0 \overset{\boldsymbol{m}}{\mapsto} q_0 \Rrightarrow^*_{i+1} p \overset{\boldsymbol{n}}{\mapsto} q$, we have $p_0 \overset{\boldsymbol{m}}{\mapsto} q_0 \Rrightarrow^*_i p' \overset{\boldsymbol{m}_1}{\mapsto} q_1$ or $p_0 \overset{\boldsymbol{m}}{\mapsto} q_0 \Rrightarrow^*_i q_1 \overset{\boldsymbol{m}_2}{\mapsto} q$. W.l.o.g., assume

$$(1)\ p_0 \overset{\boldsymbol{m}}{\mapsto} q_0 \Rrightarrow^*_i p' \overset{\boldsymbol{m}_1}{\mapsto} q_1 \Rrightarrow_{i+1} p \overset{\boldsymbol{n}}{\mapsto} q, \quad \text{and (2)}\ p \overset{\boldsymbol{n}''}{\mapsto} q \Rrightarrow^*_i q_1 \overset{\boldsymbol{m}_2}{\mapsto} q \Rrightarrow_{i+1} p \overset{\boldsymbol{n}}{\mapsto} q$$

Other cases are similar. By induction hypothesis on (2), there exists $\boldsymbol{v}'' <_{\omega(\boldsymbol{n})} \boldsymbol{n}''$ such that for all $\boldsymbol{u}_2 <_{\omega(\boldsymbol{m}_2)} \boldsymbol{m}_2$, $p \overset{\boldsymbol{v}''}{\mapsto} q \Rrightarrow^* q_1 \overset{\boldsymbol{u}_2'}{\mapsto} q$ for some $\boldsymbol{u}_2' >_{\omega(\boldsymbol{m}_2)} \boldsymbol{u}_2$ in $Post^*(\mathcal{A}_0)$.

Since $\langle p', \boldsymbol{m}_1\boldsymbol{m}_2 \to p, \boldsymbol{n}'\rangle$ and $\boldsymbol{n}' >_J \boldsymbol{n}''$, there must exist some $\boldsymbol{u}_1 <_{\omega(\boldsymbol{m}_1)} \boldsymbol{m}_1$, $\boldsymbol{u}_2 <_{\omega(\boldsymbol{m}_2)} \boldsymbol{m}_2$ and $\boldsymbol{v}' >_{J\cup\omega(\boldsymbol{n}')} \boldsymbol{v}''$ such that $\langle p', \boldsymbol{u}_1\boldsymbol{u}_2 \to p, \boldsymbol{v}'\rangle$. Hence, we have $p \overset{\boldsymbol{v}''}{\mapsto} q \Rrightarrow^* q_1 \overset{\boldsymbol{u}_2'}{\mapsto} q$ for some $\boldsymbol{u}_2' >_{\omega(\boldsymbol{m}_2)} \boldsymbol{u}_2$ in $Post^*(\mathcal{A}_0)$.

Transition $p' \overset{\boldsymbol{m}_1}{\mapsto} q_1$ in $\mathcal{A}_i$ is either i) in $\mathcal{A}_0$, ii) generated by some simple push rule, or iii) connected with some transition in $\mathcal{A}_0$ by finite steps of $\Rrightarrow$. In case i) and ii), $\boldsymbol{m}_1 \in \mathbb{N}^*$ and $p' \overset{\boldsymbol{m}_1}{\mapsto} q_1$ also in $Post^*(\mathcal{A}_0)$. For case iii), by induction hypothesis, there exists $p' \overset{\boldsymbol{u}_1'}{\mapsto} q_1$ in $Post^*(\mathcal{A}_0)$ for some $\boldsymbol{u}_1' >_{\omega(\boldsymbol{m}_1)} \boldsymbol{u}_1$. Note that $\omega(\boldsymbol{n}) = J \cup \omega(\boldsymbol{n}')$. So $Post^*(\mathcal{A}_0)$ must have dependency pair:

$$p \overset{\boldsymbol{v}''}{\mapsto} q \Rrightarrow^* p \overset{\boldsymbol{v}_\bullet'}{\mapsto} q, \text{ where } \langle p', \boldsymbol{u}_1'\boldsymbol{u}_2' \to p, \boldsymbol{v}_\bullet'\rangle \text{ and } \boldsymbol{v}'' <_{\omega(\boldsymbol{n})} \boldsymbol{v}' <_{\omega(\boldsymbol{n})} \boldsymbol{v}_\bullet' \tag{3}$$

By induction hypothesis on (1), there exists $\boldsymbol{u} <_{\omega(\boldsymbol{m})} \boldsymbol{m}$ and $\boldsymbol{u}_1'' >_{\omega(\boldsymbol{m}_1)} \boldsymbol{u}_1$ such that $p_0 \overset{\boldsymbol{u}}{\mapsto} q_0 \Rrightarrow^* p' \overset{\boldsymbol{u}_1''}{\mapsto} q_1$ in $Post^*(\mathcal{A}_0)$. Similarly, there exists $q_1 \overset{\boldsymbol{u}_2''}{\mapsto} q$ for some $\boldsymbol{u}_2'' >_{\omega(\boldsymbol{m}_2)} \boldsymbol{u}_2$ in $Post^*(\mathcal{A}_0)$. Hence we have the following pair in $Post^*(\mathcal{A}_0)$:

$$p_0 \overset{\boldsymbol{u}}{\mapsto} q_0 \Rrightarrow^* p \overset{\boldsymbol{v}_0'}{\mapsto} q, \text{ where } \langle p', \boldsymbol{u}_1''\boldsymbol{u}_2'' \to p, \boldsymbol{v}_0'\rangle \text{ and } \boldsymbol{v}'' <_{\omega(\boldsymbol{n})} \boldsymbol{v}' <_{\omega(\boldsymbol{n})} \boldsymbol{v}_0'$$

From Lemma 3 and (3), we get

$$p_0 \overset{\boldsymbol{u}}{\mapsto} q_0 \Rrightarrow^* p \overset{\boldsymbol{v}_0'}{\mapsto} q \Rrightarrow^* p \overset{\boldsymbol{v}_1'}{\mapsto} q \Rrightarrow^* p \overset{\boldsymbol{v}_2'}{\mapsto} q \Rrightarrow^* \cdots \Rrightarrow^* p \overset{\boldsymbol{v}_i'}{\mapsto} q \Rrightarrow^* \cdots$$

where $\boldsymbol{v}_i' >_{\omega(n)} \boldsymbol{v}_{i-1}'$ for any $i$. Hence, for all $\boldsymbol{v} <_{\omega(\boldsymbol{n})} \boldsymbol{n}$, there exists some $\boldsymbol{v}_i' >_{\omega(\boldsymbol{n})} \boldsymbol{n}$ and $p_0 \overset{\boldsymbol{u}}{\mapsto} q_0 \Rrightarrow^* p \overset{\boldsymbol{v}'}{\mapsto} q$. □

**Lemma 4**. Let $\mathcal{A}_0$ be an initial P-automaton of some PDVAS. If $p \overset{\boldsymbol{n}}{\mapsto} q$ in $Post_F^*(\mathcal{A}_0)$, then for all $\boldsymbol{n}' \leq \boldsymbol{n}$ and $\boldsymbol{n}' \in \mathbb{N}^k$ there exists $p \overset{\boldsymbol{n}''}{\mapsto} q$ in $Post^*(\mathcal{A}_0)$ for some $\boldsymbol{n}''$ such that $\boldsymbol{n}' \leq \boldsymbol{n}'' \leq \boldsymbol{n}$.

*Proof.* For any transition $p \overset{\boldsymbol{n}}{\mapsto} q$ in $Post_F^*(\mathcal{A}_0)$, it is either i) in $\mathcal{A}_0$, ii) generated by some simple push rule, or iii) connected with some transition in $\mathcal{A}_0$ by finite steps of $\Rrightarrow$. In case i) and ii) $\boldsymbol{n} \in \mathbb{N}^*$ and $p \overset{\boldsymbol{n}}{\mapsto} q$ also in $Post^*(\mathcal{A}_0)$. For case iii), directly by Lemma 7, we know for any $\boldsymbol{n}' \leq \boldsymbol{n}$ and $\boldsymbol{n}' \in \mathbb{N}^k$ there exists $\boldsymbol{n}'' \geq_{\omega(\boldsymbol{n})} \boldsymbol{n}'$ and $p \overset{\boldsymbol{n}''}{\mapsto} q$ in $Post^*(\mathcal{A}_0)$. Since $\boldsymbol{n}'' \geq_{\omega(\boldsymbol{n})} \boldsymbol{n}'$ we have $\boldsymbol{n}'' \leq \boldsymbol{n}$. □

## C Proof of Lemma 6

**Lemma 6**. Assume $K \overset{\widetilde{I}}{\Longmapsto} s$ in $Pre_F^*(\mathcal{A}_0)$. For each $p \in K$, $w \in \widetilde{I}$,

– if $s = K' \in \mathcal{I}(P)$, then $\langle p, w\rangle \hookrightarrow^* \langle q, \epsilon\rangle$ for some $q \in K'$.

– if $s \notin \mathcal{I}(P)$, there exists $K' \stackrel{\widetilde{I}'}{\Longmapsto} s$ in $\mathcal{A}_0$ such that $\langle p, w\rangle \hookrightarrow^* \langle p', w'\rangle$ for some $p' \in K'$ and $w' \in \widetilde{I}'$.

*Proof.* By induction on steps of the $Pre^*_F$ saturation procedure $\mathcal{A}_0, \mathcal{A}_1, \mathcal{A}_2, \cdots$. For $\mathcal{A}_0$, the statement holds immediately. Assume statements above hold for $\mathcal{A}_i$, and $\mathcal{A}_{i+1}$ is contrcuted by adding new transition $K_0 \stackrel{I_0}{\mapsto} s_0$ with $K_0 = \phi^{-1}(K'_0) \neq \emptyset$ and $I_0 = \psi^{-1}(\widetilde{I}_0) \neq \emptyset$.

$$\frac{(S, \mathcal{I}(\Gamma), \nabla, F), \quad K'_0 \stackrel{\widetilde{I}_0}{\Longmapsto} s_0}{(S, \mathcal{I}(\Gamma), \nabla, F) \oplus \{\phi^{-1}(K'_0) \stackrel{\psi^{-1}(\widetilde{I}_0)}{\longmapsto} s_0\}} \text{ if } \widetilde{I}_0 \in \mathcal{I}(\Gamma^{\leq 2}) \text{ and } (\phi, \psi) \in \Delta$$

The statement 2. in Lemma 6 is similarly proved as the statement 1., and we give a proof only for the statement 1. According to the definition of $\oplus$ (in Definition 12), there are three cases:

– There exists $(K_1 \stackrel{I_1}{\mapsto} s_0) \in \nabla$ with $K_0 \subseteq K_1 \wedge I_0 \subseteq I_1$. Then, no new edges are added.
– There exists $(K_0 \stackrel{I_1}{\mapsto} s_0) \in \nabla$. Then, $K_0 \stackrel{I_1}{\mapsto} s_0$ is updated with $K_0 \stackrel{I_0 \cup I_1}{\longmapsto} s_0$.
– Otherwise, $K_0 \stackrel{I_0}{\mapsto} s_0$ is added.

The first case is immediate. The second case is the most complex, and the third case follows similarly. Here we focus on the second case.

Assume that a path $K \stackrel{\widetilde{I}}{\Longmapsto} s$ contains $K_0 \stackrel{I_0 \cup I_1}{\longmapsto} s_0$ $k$-times. We apply (nested) induction on $k$, and we focus on its leftermost occurence. We only need to consider elements in $I_0$ since those in $I_1$ is by induction hypothesis.

Let $\widetilde{I} = \widetilde{I}_l I_0 \widetilde{I}_r$, and let $K \stackrel{\widetilde{I}_l}{\Longmapsto} K_0 \stackrel{I_0}{\mapsto} s_0 \stackrel{\widetilde{I}_r}{\Longmapsto} s$ for $w_l \in \widetilde{I}_l$, $\gamma \in I_0$, and $w_r \in \widetilde{I}_r$. For each $p \in K$, by induction hypothesis on $K \stackrel{\widetilde{I}_l}{\Longmapsto} K_0$, there exists $p_0 \in K_0$ with $\langle p, w_l\rangle \hookrightarrow^* \langle p_0, \epsilon\rangle$. By the definition of saturation rules, we have $\langle p_0, \gamma\rangle \hookrightarrow \langle \phi(p_0), \psi(\gamma)\rangle$ for $\phi(p_0) \in K'_0$ and $\psi(\gamma) \in \widetilde{I}_0$. Again, by induction hypothesis on $K'_0 \stackrel{\widetilde{I}_0}{\Longmapsto} s_0 \stackrel{\widetilde{I}_r}{\Longmapsto} s$, there exists $q \in s$ with $\langle \phi(p_0), \psi(\gamma) w_r\rangle \hookrightarrow^* \langle q, \epsilon\rangle$. Thus, we have $\langle p, w_l \gamma w_r\rangle \hookrightarrow^* \langle p_0, \gamma w_r\rangle \hookrightarrow \langle \phi(p_0), \psi(\gamma) w_r\rangle \hookrightarrow^* \langle q, \epsilon\rangle$. $\square$