The Termination/Boundedness Problem of Well-Structured Pushdown Systems

Suhua Lei\textsuperscript{1}, Xiaojuan Cai\textsuperscript{1}, and Mizuhoit\textsuperscript{2} Ogawa

\textsuperscript{1} BASICS Lab, Shanghai Jiao Tong University, China
leisuhua, cxj@sjtu.edu.cn

\textsuperscript{2} Japan Advanced Institute of Science and Technology, Japan
mizuhito@jaist.ac.jp

Abstract. Recently, models that combine vector addition systems and pushdown systems have attracted lots of attentions. They can model some classes of concurrent recursive computations, still enjoying certain decidability properties. A well-structured pushdown system (WSPDS) \cite{1} has been proposed to unify such models. It extends a pushdown system with (possibly infinitely many) well-quasi-ordered states and stack alphabet. Several properties, such as coverability, termination, and boundedness, were shown to be decidable under restrictions of either a finite set of states or finite stack alphabet. Inspired by the work \cite{2} of Leroux et al., this paper shows that the termination and boundedness problems of a general WSPDS are decidable, and gives the lower and upper bounds for a subclass of WSPDS, which are both Hyper-Ackermannian. Compared with \cite{2}, our generalization considers well-quasi-ordered stack alphabet rather than finite stack alphabet, whereas \cite{2} considers a WSPDS with finite stack alphabet (but with well-quasi-ordered states).

1 Introduction

A pushdown system has finitely many control states equipped with a stack for storing words over finite stack alphabet. Recently, it has been combined with vector addition systems by Sen et al \cite{3} and Bouajani et al \cite{4} to model multi-thread recursive programs. A well-structured pushdown system (WSPDS) extends a pushdown system with well-quasi-ordered states and stack alphabet to unify existing works, such as \cite{3} and \cite{4}. Several properties, such as coverability \cite{1}, termination, and boundedness \cite{2}, were shown to be decidable under restrictions of either a finite set of states or finite stack alphabet. In this paper,

\begin{itemize}
  \item the termination problem of a WSPDS and the boundedness problem of a strict WSPDS are shown to be decidable by extending the standard reachability tree techniques.
  \item the lower and upper bounds of these problems are both Hyper-Ackermannian for a subclass of WSPDS, which has natural number vectors as both states and stack alphabet.
\end{itemize}

Our proof techniques are mostly rely on \cite{1} and \cite{2}; however, the former only handles the coverability, and the latter assumes that the stack alphabet is finite. Our generalization extends finite stack alphabet to well-quasi-ordered one.
2 Well-structured pushdown systems

A quasi-ordering \((X, \leq)\) is a reflexive and transitive relation \(\leq\) on \(X\). It is a well-quasi-ordering (WQO) if any infinite sequence over \(X\) contains an increasing pair. A partial order is an antisymmetric quasi-ordering. We denote \(x < y\) if \(x \leq y\) and \(x \neq y\).

Since the transition rules manipulate words over stack alphabet \((\Gamma, \leq)\), we extend the binary relation \(\leq\) on alphabet to relation \(\leq\) on words such that \(a_1a_2\ldots a_n \leq b_1b_2\ldots b_m\) if and only if \(m = n\) and \(a_i \leq b_i\) for each \(i\). Note that for any WQO \((\Gamma, \leq)\), the extending order \((\Gamma^*, \leq)\) may not be a WQO.

**Definition 1.** \([1]\) A WSPDS is a triplet \(M = (\langle P, \leq \rangle, \langle \Gamma, \leq \rangle, \Delta)\) where

1. \((P, \leq)\) and \((\Gamma, \leq)\) are WQOs, and
2. \(\Delta \subseteq  \mathcal{F}(P \times \Gamma^\leq, P \times \Gamma^\leq)\) is the finite set of monotonic transition rules (w.r.t. \(\leq \times \leq\)), where \(\mathcal{F}(X,Y)\) denotes the set of partial functions from \(X\) to \(Y\).

We denote \((p, w \mapsto p', w')\) if there exists \(\psi(p, u) = (p', u')\) for \(u, u' \in \Gamma^\leq\) when \(w = u.\varepsilon\) and \(w' = u'.\varepsilon\).

A configuration \(c\) is a pair \(\langle p, w \rangle\), where \(p\) is a state and \(w\) is a stack word.

We denote \(w[1]\) and \(|w|\) for the top element and the length of \(w\), respectively. We define the head of a configuration as \(h((p, w)) = (p, w[1])\) if \(w \neq \varepsilon\), otherwise, \(h((p, w)) = (p, \bot)\). If we require \(\bot \leq \alpha\) for each \(\alpha \in \Gamma\), then the set of heads is a WQO over \(\Gamma = \leq \times \leq\) by Dickson’s Lemma.

A WSPDS is strict if \((P, \leq)\) and \((\Gamma, \leq)\) are partial orders, and the transition functions are strictly monotonic, i.e., for \((p, u) < (p', u')\) with \(p, p' \in P\) and \(u, u' \in \Gamma^\leq\), if \(\psi(p, u)\) is defined, then \(\psi(p, u) < \psi(p', u')\).

A run starting from the initial configuration \(c_0\) is a sequence \(c_0, c_1, \cdots\) of configurations, where \(c_{i-1} \mapsto c_i\) for every index \(i > 0\). The reachability set of \(c_0\) is the set of all configurations that occur on some run from \(c_0\).

- **Termination** asks whether all runs starting from \(c_0\) are finite.
- **Boundedness** asks whether the reachability set of \(c_0\) is finite.

**Example 1.** \(M = (\langle \mathbb{N}, \leq \rangle, \langle \mathbb{N}, \leq \rangle, \Delta)\) is a WSPDS with both control states and stack alphabet being natural numbers. It is also a strict WSPDS since every transition function is strictly monotonic.

\[
\Delta = \left\{ \begin{array}{l}
  r_1 : p, \alpha \rightarrow p + 1, (\alpha - 1)(\alpha - 1) \\
  r_2 : p, \epsilon \rightarrow p + 2, 0 \text{ if } p \geq 2 \\
  r_3 : p, \alpha \rightarrow p - 3, \alpha + 3 \\
  r_4 : p, \alpha\beta \rightarrow p, \alpha + \beta - 2
\end{array} \right\}
\]

Assume \(c_0 = (1, 1)\), this is an infinite run with infinite reachability set:

\((1, 1) \xrightarrow{r_1} (2, 0) \xrightarrow{r_3} (4, 000) \xrightarrow{r_3} (1, 300) \xrightarrow{r_3} (1, 10) \xrightarrow{r_1} (2, 000) \cdots\)

If we change rule \(r_1\) to \(r'_1 : p, \alpha \rightarrow p + 1, (\alpha - 1)\), we have an infinite run with finite reachability set: \((1, 1) \xrightarrow{r'_1} (2, 0) \xrightarrow{r_3} (4, 000) \xrightarrow{r_3} (1, 30) \xrightarrow{r_3} (1, 1) \xrightarrow{r_1} (2, 0) \cdots\).

If we further remove the rule \(r_2\), all runs starting from \(c_0\) terminate.
3 Termination/Boundedness problem

The reduced reachability tree is a standard technique for vector addition systems. Leroux et al. [2] extend it to solve the termination/boundedness problem of a WSPDS with finite stack alphabet. Here we study a general WSPDS (i.e., both with well-quasi-ordered states and stack alphabet). The reachability tree of a WSPDS with the initial configuration $c_0$ is a directed (unordered) tree with root $r : c_0$, and each node $n : c_n$ has $m : c_m$ as a child if $c_n \hookrightarrow c_m$.

Termination problem

In this part, we give a necessary and sufficient condition for non-termination. This condition can be checked in a finite prefix of the reachability tree, which implies the decidability of termination problem by König’s Lemma.

Definition 2. A node $s : \langle p, w \rangle$ pumps a node $t : \langle q, v \rangle$ if

- there is a path from $s$ to $t$, and every node $t' : \langle p', w' \rangle$ on it satisfies $|w'| \geq |w|$.
- $h(\langle p, w \rangle) \preceq h(\langle q, v \rangle)$, i.e., $p \preceq q$ and either $w = \epsilon$ or $w[1] \leq v[1]$.

We call a node pumpable if there exists some node that pumps it. The notion of pumpable nodes is similar to subsumed nodes in [2], but we consider the increasing of heads, other than just states. Intuitively, the first condition of Definition 2 means that the run from $\langle p, w \rangle$ to $\langle q, v \rangle$ never touches the content below $w[1]$, and the second implies that configuration $\langle q, v \rangle$ with larger-or-equal head than $\langle p, w \rangle$ can repeat and pump a configuration. In Example 1, configuration $\langle 1, 1 \rangle$ pumps $\langle 1, 10 \rangle$, and configuration $\langle 1, 1 \rangle$ pumps $\langle 1, 1 \rangle$ with rule $r'_1$.

Conversely, assume $\langle p_0, w_0 \rangle \hookrightarrow \langle p_1, w_1 \rangle \cdot \cdot \cdot$ is an infinite run, we can extract an infinite subsequence, say $\langle p_{i_0}, w_{i_0} \rangle, \langle p_{i_1}, w_{i_1} \rangle, \cdot \cdot \cdot$ where each node is chosen if it has the minimal length in the run after it. Note that each pair $\langle p_{i_k}, w_{i_k} \rangle$ and $\langle p_{i_j}, w_{i_j} \rangle$ with $k < j$ in this subsequence satisfies the first condition of pumpable nodes. By the fact that the set of heads is a WQO over $\preceq$, the infinite subsequence we obtain above must contain a pumpable node.

Let the reduced reachability tree be the largest prefix of the reachability tree such that every pumpable node has no child. The following theorem provides an algorithm to decide the termination of a WSPDS.

Theorem 1. A WSPDS has an infinite run if, and only if, its reduced reachability tree contains a pumpable node.

Boundedness problem

If a pumpable node is exactly the same as the one that pumps it, then the infinite run induced by this pumpable node will have a finite reachability set. Therefore, we need more than a pumpable node for the boundedness problem.

Definition 3. A node $s : \langle p, w \rangle$ strictly pumps a node $t : \langle q, v \rangle$ if $s$ pumps $t$ and either $|w| < |v|$ or $h(\langle p, w \rangle) \preceq h(\langle q, v \rangle)$. 

In Example 1, configuration $⟨1,1⟩$ strictly pumps $⟨1,10⟩$. However, with rule $r_1'$, $⟨1,1⟩$ does not strictly pump $⟨1,1⟩$.

**Theorem 2.** A strict WSPDS has an infinite reachability set if, and only if, its reduced reachability tree contains a strictly pumpable node.

The proof is similar to Theorem 1. We need the strictness of WSPDS for the proof because only in a partial order can we get $a < b$ from $a \neq b$ and $a \leq b$ (this is not true for a quasi-order); and only strictly monotonic transition rules guarantee the strict growth of configurations.

### 4 Complexity: lower and upper bounds

The estimation on the size of reduced reachability tree is difficult for arbitrary WQOs. We make the following assumption for WSPDS:

- the WQOs are restricted to vectors, i.e., $P = \mathbb{N}^d$ and $\Gamma = \mathbb{N}^k$.
- the changes of vectors caused by one-step computation should be controlled: each non-pop rule $(p, w, q, v) \in \Delta$ satisfies $\|q - p\|_{\infty} \leq 1$ and $\|\Sigma(v) - \Sigma(w)\|_{\infty} \leq 1$, where $\|p\|_{\infty}$ maps a vector $p$ to its maximum component, and $\Sigma(w) = \sum_{i=1}^{n} w[i]$ if $w = w[1] \cdots w[n]$.

With these restrictions, any run can be mapped to a $n$-controlled nested sequence, and the height of a reduced reachability tree is the maximal length of $n$-controlled bad nested sequence, which is proved to be Hyper-Ackermannian [2]. Given a WSPDS $A = (⟨\mathbb{N}^d, \leq⟩, ⟨\mathbb{N}^k, \leq⟩, \Delta)$ with initial configuration $⟨p_0, w_0⟩$, the size of $A$ can be defined as

$$|A| = d + k + (d + k) \cdot \max\{\|p_0\|_{\infty}, \|\Sigma(w_0)\|_{\infty}\} + (d + k) \cdot |\Delta|.$$  

**Theorem 3.** The reduced reachability tree of this subclass of WSPDS has at most $F_{\omega}(|A|)$ nodes, where $F_{\omega}$ is Hyper-Ackermannian function defined by diagonalization over the ordinals $\alpha < \omega^\omega$.

The lower bound of the size of the reduced reachability tree follows the same paper [2]. Leroux et al. have proved the lower bound of the reduced reachability tree is Hyper-Ackermannian for their model, which is a subclass of the restricted WSPDS in this section.

### References


