On the Complexity of ω-Pushdown Automata

Fu Song[1,2], Min Zhang[1], Wanwei Liu[3], and Yusi Lei[1]

[1] Shanghai Key Laboratory of Trustworthy Computing,
East China Normal University
[2] School of Information Science and Technology,
ShanghaiTech University
[3] School of Computer Science,
National University of Defense Technology

Abstract. Finite automata over infinite words (called ω-automata) play an important role in the automata-theoretic approach to system verification. Since different types of ω-automata differ in their succinctness and in the complexity for their emptiness problems. Theory of ω-automata were well-studied in the literature. Pushdown automata over infinite words (called ω-PDAs), a generalization of ω-automata, are a natural model of recursive programs. Our goal in this paper is to do a relatively complete investigation on the complexity of the emptiness problems for variants of ω-PDAs. For this purpose, we consider ω-PDAs of five standard acceptance types: Büchi, Parity, Rabin, Streett and Muller acceptances. Based on the transformation for ω-automata and the efficient algorithm proposed by Esparza et al. in CAV'00 for checking the emptiness problem of ω-PDAs with Büchi acceptance, it is trivial to check the emptiness problem of other ω-PDAs. However, this naive approach is not optimal. In this paper, we propose novel algorithms for the emptiness problem of ω-PDAs based on the observations of structure of accepting runs. Our algorithms outperform algorithms that goes through Büchi PDAs. In particular, the space complexity of the algorithm for Streett acceptance that goes through Büchi acceptance is exponential, while ours is polynomial. The algorithm for Parity acceptance that goes through Büchi acceptance is in $\mathbf{O}(k^3n^2m)$ time and $\mathbf{O}(k^2nm)$ space, while ours is in $\mathbf{O}(kn^2m)$ time and $\mathbf{O}(nm)$ space, where n (resp. m and k) is the number of states (resp. transitions and index). Finally, we show that our algorithms lead to a better solution for the pushdown model checking problem against linear temporal logic with fairness.

1 Introduction

Background. Automata on infinite words were first introduced in the 1960s by Büchi [5] in order to give a decision procedure for SlS, the monadic second-order theory of one successor, where the acceptance condition specifies some "good" state that should be visited infinitely often. Muller used automata on infinite words to describe the behavior of non-stabilizing circuits, in which the acceptance condition specifies explicitly all the "good" infinity sets [27]. McNaughton proved a fundamental result in the theory of ω-automata which related different models of automata considered by Büchi and Muller [25], namely the classes of languages accepted by nondeterministic Büchi automata and by deterministic Muller automata are identical. McNaughton introduced a special case of Muller's condition when converting any Büchi automaton into a deterministic Muller automaton which was later formalized by Rabin [30]. The Rabin acceptance condition is a set of pairs of states, in which a run is *accepting* if and only if there is a pair such that no "bad" state in the pair is visited infinitely often and some good state in the pair is visited infinitely often. There has been a lot of research in the theory of ω-automata. Many variants of acceptance conditions were introduced for ω-automata such as Streett condition [35], Parity condition [26] and Emerson-Lei condition [10] (see the survey [36]).

ω-automata play an important role in the automata-theoretic approach to system verification, as a unifying paradigm for the specification, verification, and synthesis of nonterminating systems [38, 37]. The basic idea is that to verify a system against some property, the system is modeled as a finite state automaton, the property is described by an ω-automaton which may be constructed from an LTL formula [29], and the verification problem is reduced to the emptiness problem of an ω-automaton, which is the "product" of the system automaton and the property ω-automaton. Therefore, a lot of effort has already been put into the theory of ω-automata: in particular, transformation, expressiveness and emptiness problem. For example, it was proved that the expressiveness of deterministic Büchi automata are weaker than nondeterministic Büchi automata, while the later are as expressive as deterministic/nondeterministic ω-automata with Parity, Rabin, Streett or Muller acceptance [24, 23, 28]. Deterministic Rabin/Streett/Muller automata are exponentially more succinct than deterministic Parity automata, while nondeterministic Streett automata are exponentially more succinct than nondeterministic Büchi automata. The emptiness problem for nondeterministic Büchi automata (resp. Parity, Rabin, Streett and Muller automata) can be solved in linear time or NLOGSPACE (resp. polynomial time) [17, 2, 18].

Pushdown automata over infinite words (called ω-PDAs) are a generation of standard ω-automata [20–22, 7–9]. The languages recognized by ω-PDAs are called ω-context-free languages. These works concentrated on the relation between ω-grammars and ω-PDAs. It was independently shown in [21] and [7] that ω-PDAs with Muller acceptance can be transformed into the one with Büchi acceptance causing an exponentially blowup in the size of the automaton. ω-PDAs are a very useful model, in particular, for the verification of recursive programs [3, 13, 11]. Therefore, efficient algorithm for the emptiness problem of ω-PDAs with Büchi acceptance was developed in [11]. Besides these results, the algorithmic theory of ω-PDAs has not been investigated very much.

Contribution. In this work, we investigate the emptiness problem of ω-PDAs with respect to the five standard acceptance conditions, namely, Büchi, Parity, Rabin, Streett and Muller acceptances. We use the acronyms in $\{B, P, R, S, M\} \times \{PDA\}$ to denote the different types of ω-PDAs. The first symbol stands for the type of acceptance condition: B for Büchi, P for Parity, R for Rabin, S for Streett

Acceptance	Büchi	Rabin/Parity	Streett	Muller
Time	$\mathbf{O}(n^2m)$	$\mathbf{O}(k^3n^2m)$	$\mathbf{O}(k^3n^2m2^{3\mathsf{min}(k,n)})$	$\mathbf{O}(n^2m(k+\sum_{i=1}^{k}\|F_i\|)^3)$
Space	$\mathbf{O}(nm)$	$\mathbf{O}(k^2nm)$	$\mathbf{O}(k^2nm2^{2\mathsf{min}(k,n)})$	$\mathbf{O}(nm(k+\sum_{i=1}^{k}\|F_i\|)^2)$
Our result	-	Thm. 4	Thm. 5	Thm. 6
Time	-	$\mathbf{O}(kn^2m)$	$\mathbf{O}(k^3n^2m2^{\mathsf{min}(n,k)})$	$\mathbf{O}(m \cdot \max(n^2, \sum_{i\in[k]}\|F_i\|^5))$
Space	-	$\mathbf{O}(nm)$	$\mathbf{O}(k^2nm)$	$\mathbf{O}(m \cdot \max\{n, \|F_i\|^3 \mid i \in [k]\})$

Table 1. Complexities of the emptiness problem, where n (resp. m and k) is the number of states (resp. transitions and index or pairs).

and M for Muller. For example PPDAs denote ω-PDAs with Parity acceptance and SPDAs denote ω-PDAs with Streett acceptance.

The emptiness problem of BPDAs was studied in [3, 11] which was applied to perform LTL model checking on pushdown systems. One can easily transform ω-PDAs with other acceptance types to BPDAs by adapting the transformation for ω-automata (e.g. [24, 28]). In a nutshell, the transformation is acting on the control states of ω-PDAs and the stack does not play a role. However, this approach is not optimal. We propose direct and efficient algorithms to the emptiness problems of PPDAs, RPDAs, SPDAs and MPDAs. The results are shown in Table 1. For instance, the best known algorithm for the emptiness problem of BPDAs is in $\mathbf{O}(n^2m)$ time and $\mathbf{O}(nm)$ space, where n (resp. m) denotes the number of states (resp. transitions) [11]. The resulting BPDA from an SPDA with n states, m transitions and k pairs of accepting states using the best transformation (to the best of our knowledge) has $\mathbf{O}(n2^{\mathsf{min}(k,n)})$ states and $\mathbf{O}(m2^{\mathsf{min}(k,n)})$ transitions. Therefore, the best algorithm to the emptiness problem of SPDAs that goes through BPDAs is in $\mathbf{O}(k^3n^2m2^{3\mathsf{min}(k,n)})$ time and $\mathbf{O}(k^2nm2^{2\mathsf{min}(k,n)})$ space. Our efficient algorithm works in $\mathbf{O}(k^3n^2m2^{\mathsf{min}(n,k)})$ time and $\mathbf{O}(k^2nm)$ space.

The benefit of using our emptiness checking algorithm is even greater when we consider the model checking problem with fairness. It was shown that model checking linear temporal logic (LTL for short) on pushdown systems can be reduced to the emptiness problem of BPDAs [3, 11]. When fairness is considered, it is better to solve the model checking problem by reducing to the emptiness problem of SPDAs rather than BPDAs. In particular, the time complexity is polynomial in the size of the weak fairness when reducing to the emptiness problem of SPDAs, while it is exponential if we use BPDAs.

Related work. ω-PDAs were first proposed and studied by Linna, Cohen and Gold in [20–22, 7–9]. Expressiveness of BPDAs and MPDAs were studied in several works such as [7–9]. After about two decades later, the emptiness problem of BPDAs and PPDAs were studied for model checking pushdown systems against linear-time temporal logics such as LTL and linear-time μ-calculus in [3, 14, 11, 13]. Emptiness problem of BPDAs was independently studied by Bouajjani et al. in [3] and by Finkel et al. in [14] for model checking pushdown systems against LTL. Esparza et al. [11] proposed an efficient implementation algorithm for the emptiness problem of BPDAs which results in a software model checker Moped [13].

Besides the above works, the emptiness problem of an generalization of ω-PDAs with one letter (a.k.a. alphabet-free), called alternating ω-PDAs, was investigated for model checking pushdown systems against branching-time temporal logics such as CTL, CTL* and μ-calculus, or equivalent problems. Walukiewicz investigated the pushdown game with Parity objectives and pushdown model checking for μ-calculus in [39] which are equivalent to the emptiness problem of alternating PPDAs. Hague and Ong proposed efficient algorithms for winning regions of pushdown game with Parity objectives and the model checking problem of pushdown systems against μ-calculus respectively in [15] and [16] which lead to the software model checker PDSolver. The another equivalent problem of this problem, that is the emptiness problem of pushdown tree automata was studied by Kupferman et al. in [19].

Cachat studied the pushdown game with Büchi objectives which is also equivalent to the emptiness problem of alternating BPDAs. Song and Touili proposed an efficient algorithm for the emptiness problem of alternating BPDAs and applied it to do CTL model checking on pushdown systems [32, 34]. This algorithm was later implemented in the software model checker PuMoC [33].

The rest of this paper is structured as follows. Section 2 introduces basic notations, and revisits the results of transformation of ω-PDAs. In Section 3, we propose efficient algorithms to the emptiness problems of RPDAs, PPDAs, SPDAs and MPDAs. Section 4 presents reductions from pushdown model checking against LTL formulae with fairness to the emptiness problem of ω-PDAs and discusses the efficiency of the underlying approaches. Section 5 contains conclusion and future work.

2 Preliminaries

We denote by $[k]$ the set $\{1, ..., k\}$ for a natural number k. Given a finite alphabet Σ consisting of *letters*, a *finite word* over Σ is a sequence $w = a_0...a_n$ of letters from Σ, while an *infinite word* over Σ is an infinite sequence $w = a_0a_1...$ of letters from Σ. We denote by Σ^* and Σ^ω respectively the sets of all the finite words and infinite words over Σ, ϵ the empty word with $|\epsilon| = 0$, and Given a sequence $w = a_0a_1...$, we use w_i to denote the letter a_i.

2.1 ω-Pushdown Automata

Definition 1. *An* ω-Pushdown Automaton *(ω-PDA for short) is a tuple $\mathcal{P} = (\Sigma, P, \Gamma, \Delta, p_0, \bot, \mathcal{F})$, where Σ is the finite input alphabet, P is a finite set of states, $p_0 \in P$ is the initial state, Γ is a finite stack alphabet, $\bot \in \Gamma$ is the initial stack content, $\Delta \subseteq (P \times \Gamma) \times \Sigma \times (P \times \Gamma^*)$ is a transition relation such that for every $((p, \bot), a, (p', u)) \in \Delta$, u is in the form of $u'\bot$, and $\mathcal{F}$ is an acceptance condition which will be defined below.*

A *configuration* of the ω-PDA $\mathcal{P}$ is a tuple $\langle p, u\rangle \in P \times \Gamma^*$, $\langle p_0, \bot\rangle$ is the *initial configuration*. Let $\mathbb{C}_{\mathcal{P}}$ denote the set $P \times \Gamma^*$. For every $((p,\gamma), a, (p', u)) \in \Delta$, we will use $\langle p, \gamma\rangle \overset{a}{\hookrightarrow}_{\mathcal{P}} \langle p', u\rangle$ instead. For every $u' \in \Gamma^*$, if $\langle p, \gamma\rangle \overset{a}{\hookrightarrow}_{\mathcal{P}} \langle p', u\rangle$, then the configuration $\langle p', uu'\rangle$ is an *immediate successor* of the configuration $\langle p, \gamma u'\rangle$, denoted by, $\langle p, \gamma u'\rangle \overset{a}{\Longrightarrow}_{\mathcal{P}} \langle p', uu'\rangle$. Intuitively, if $\langle p, \gamma\rangle \overset{a}{\hookrightarrow}_{\mathcal{P}} \langle p', u\rangle$, then the ω-PDA $\mathcal{P}$ can move into the configuration $\langle p', uu'\rangle$ when it is at the configuration $\langle p, \gamma u'\rangle$ and reads the input letter a. A pair $\langle p, \gamma\rangle$ such that $p \in P$ and $\gamma \in \Gamma$ is called *head*. We also abbreviate $\mathbb{C}_{\mathcal{P}}$, $\hookrightarrow_{\mathcal{P}}$ and $\Longrightarrow_{\mathcal{P}}$ as $\mathbb{C}$, $\hookrightarrow$ and $\Longrightarrow$ if $\mathcal{P}$ is clear from the context.

Let $\Longrightarrow^+, \Longrightarrow^* \subseteq \mathbb{C} \times \Sigma^* \times \mathbb{C}$ be the smallest relations such that the following conditions hold:

- $c \overset{\epsilon}{\Longrightarrow}{}^* c$ for every $c \in \mathbb{C}$,
- $c \overset{aw}{\Longrightarrow}{}^* c''$ and $c \overset{aw}{\Longrightarrow}{}^+ c''$, if $c \overset{a}{\Longrightarrow} c'$ and $c' \overset{w}{\Longrightarrow}{}^* c''$.

Intuitively, $c \overset{w}{\Longrightarrow}{}^* c'$ denotes that the ω-PDA $\mathcal{P}$ can move into the configuration c' when $\mathcal{P}$ is at the configuration c and reads the input word w.

Let $pre^* : 2^{\mathbb{C}} \to 2^{\mathbb{C}}$ be the function such that $pre^*(C) = \{c \in \mathbb{C} \mid \exists w \in \Sigma_\epsilon^*, \exists c' \in C : c \overset{w}{\Longrightarrow}{}^* c'\}$ for every $C \subseteq \mathbb{C}$. Given a set of configurations $C \subseteq \mathbb{C}_{\mathcal{P}}$ and a configuration $c \in \mathbb{C}$, the *reachability problem* is to determine whether $c \in pre^*(C)$ or not.

Given an infinite word $w = a_0a_1... \in \Sigma^\omega$, a *run* of the ω-PDA $\mathcal{P}$ on w is an infinite sequence $\rho = c_0c_1...$ of configurations such that $c_0 = \langle p_0, \bot\rangle$, for every $i \geq 0$, there is $a'_i \in \Sigma$ with $c_i \overset{a'_i}{\Longrightarrow} c_{i+1}$, and $w = a'_0a'_1....$ A *path* of the ω-PDA $\mathcal{P}$ is a finite sequence of configurations which is a prefix of a run. A path starting from a configuration c and end with a configuration c' over a word w is sometimes abbreviated as $c \overset{w}{\Longrightarrow}{}^* c'$. Given a run ρ, let $\mathsf{Inf}(\rho)$ denote the set of states visited infinitely often in the run ρ. In this work, we consider the following five well-known acceptance conditions in the literature.

- Büchi acceptance [5]: $\mathcal{F} \subseteq P$ is a finite set of accepting states. A run ρ is *accepting* if and only if $\mathsf{Inf}(\rho) \cap \mathcal{F} \neq \emptyset$.
- Parity acceptance [26]: $\mathcal{F} : P \to [k]$ is a function assigning to each state a priority, where k is some natural number called by *index*. A run ρ is *accepting* if and only if $\mathsf{min}(\{\mathcal{F}(p) \mid p \in \mathsf{Inf}(\rho)\})$ is *even*.
- Rabin acceptance [30]: $\mathcal{F} = \{(E_1, F_1), ..., (E_k, F_k)\}$ is a set of pairs of states with $E_i, F_i \subseteq P$ for every $i \in [k]$. A run ρ is *accepting* if and only if there exists some $i \in [k]$ such that $E_i \cap \mathsf{Inf}(\rho) = \emptyset$ and $F_i \cap \mathsf{Inf}(\rho) \neq \emptyset$.
- Streett acceptance [35]: $\mathcal{F} = \{(E_1, F_1), ..., (E_k, F_k)\}$ is a set of pairs of states with $E_i, F_i \subseteq P$ for every $i \in [k]$. A run ρ is *accepting* if and only if for all $i \in [k]$, $E_i \cap \mathsf{Inf}(\rho) \neq \emptyset$ or $F_i \cap \mathsf{Inf}(\rho) = \emptyset$.
- Muller acceptance [27]: $\mathcal{F} = \{F_1, ..., F_k\}$ is a set of sets of states with $F_i \subseteq P$ for every $i \in [k]$. A run ρ is *accepting* if and only if $\mathsf{Inf}(\rho) \in \mathcal{F}$.

The ω-PDA $\mathcal{P}$ *accepts* an infinite word $w \in \Sigma^\omega$ if and only if it has an *accepting* run over the input word w. Let $\mathcal{L}(\mathcal{P}) \subseteq \Sigma^\omega$ denote the set of infinite words accepted by the ω-PDA $\mathcal{P}$, called the *language* of $\mathcal{P}$. An ω-PDA $\mathcal{P}$ is *empty* if and only if $\mathcal{L}(\mathcal{P}) = \emptyset$. Two ω-PDAs $\mathcal{P}_1$ and $\mathcal{P}_2$ are *equivalent* if $\mathcal{L}(\mathcal{P}_1) = \mathcal{L}(\mathcal{P}_2)$. Given an ω-PDA $\mathcal{P}$, its *emptiness problem* is to determine whether $\mathcal{L}(\mathcal{P}) = \emptyset$ or not. W.l.o.g., we assume that the input alphabet is singleton. Indeed, for the emptiness problem, the finite alphabet can be abstracted as a singleton set.

We will use $\mathcal{P}$-automata as "data structures" to finitely represent infinite sets of configurations.

2.2 $\mathcal{P}$-Automata

Definition 2. *[3] Given an ω-PDA $\mathcal{P} = (\Sigma, P, \Gamma, \Delta, p_0, \bot, \mathcal{F})$, a $\mathcal{P}$-automaton is a tuple $\mathcal{A} = (Q, \Gamma, \delta, Q_0, F)$, where Q is a finite set of states with $P \subseteq Q$, $Q_0 \subseteq Q$ is a finite set of initial states, $F \subseteq Q$ is a finite set of final states and $\delta \subseteq Q \times \Gamma \times Q$ is a transition relation.*

We write $p \xrightarrow{\gamma} q$ instead of $(p, \gamma, q) \in \delta$. We define the transition relation $\longrightarrow^* \subseteq Q \times \Gamma^* \times Q$ as the smallest relation such that

- $q \xrightarrow{\epsilon}^* q$, for every $q \in Q$,
- $q \xrightarrow{\gamma u}^* q''$, if $q \xrightarrow{\gamma} q'$ and $q' \xrightarrow{u}^* q''$.

A configuration $\langle p, u \rangle \in \mathbb{C}$ is *recognised* (accepted) by $\mathcal{A}$ if $p \xrightarrow{u}^* q$ such that $p \in Q_0$ and $q \in F$. Let $\mathcal{L}(\mathcal{A})$ be the set of configurations recognised by $\mathcal{A}$. A set of configurations $C \subseteq \mathbb{C}$ is *regular* if there is a $\mathcal{P}$-automaton $\mathcal{A}$ such that $\mathcal{L}(\mathcal{A}) = C$.

Theorem 1. *[11] Given an ω-PDA $\mathcal{P} = (\Sigma, P, \Gamma, \Delta, p_0, \bot, \mathcal{F})$ and a regular set of configurations $C \subseteq \mathbb{C}$ recognised by a $\mathcal{P}$-automaton $\mathcal{A} = (Q, \Gamma, \delta, Q_0, F)$, a new $\mathcal{P}$-automaton $\mathcal{A}'$ can be constructed in $\mathbf{O}(|Q|^2 \cdot |\Delta|)$ time and $\mathbf{O}(|Q| \cdot |\Delta| + |\delta|)$ space such that $\mathcal{L}(\mathcal{A}') = pre^*(C)$.*

The emptiness problem of BPDAs was shown in PTIME by Bouajjani et al. [3]. Later, Esparza et al. proposed an efficient algorithm for this problem [11].

Theorem 2. *[11] The emptiness problem of BPDAs $\mathcal{P} = (\Sigma, P, \Gamma, \Delta, p_0, \bot, \mathcal{F})$ can be decided in $\mathbf{O}(|P|^2 \cdot |\Delta|)$ time and $\mathbf{O}(|P| \cdot |\Delta|)$ space.*

By leveraging the well-known transformation for ω-automata (e.g. [24, 28]), it is easy to obtain the results shown in Table 2 (c.f. Appendix for self-contained proofs). In a nutshell, the transformation is acting on the control states of ω-PDAs and the stack does not play a role. By applying Theorem 2, we can get that:

Acceptance	R2B	P2B	S2B	M2B
♯Ctrl. States.	$\mathbf{O}(kn)$	$\mathbf{O}(kn)$	$\mathbf{O}(kn2^{\mathsf{min}(k,n)})$	$\mathbf{O}(n(k+\Vert\mathcal{F}\Vert))$
♯Trans.	$\mathbf{O}(km)$	$\mathbf{O}(km)$	$\mathbf{O}(km2^{\mathsf{min}(k,n)})$	$\mathbf{O}(m(k+\Vert\mathcal{F}\Vert))$

Table 2. Summary of the transformation for ω-PDAs, where n (resp. m and k) is the number of states (resp. transitions and index/pairs), ♯Ctrl. (resp. ♯Trans.) denotes the number of states (resp. transition rules) in the resulting ω-PDAs.

Theorem 3. *The emptiness problems of ω-PDAs $\mathcal{P} = (\Sigma, P, \Gamma, \Delta, p_0, \bot, \mathcal{F})$ (with k index/pairs in $\mathcal{F}$) can be solved in:*

- $\mathbf{O}(k^3|P|^2 \cdot |\Delta|)$ *time and* $\mathbf{O}(k^2|P| \cdot |\Delta|)$ *space for RPDAs and PPDAs.*
- $\mathbf{O}(k^3|P|^2 \cdot |\Delta| \cdot 2^{3min(k,|P|)})$ *time and* $\mathbf{O}(k^2|P| \cdot |\Delta| \cdot 2^{2min(k,|P|)})$ *space for SPDAs.*
- $\mathbf{O}(|P|^2 \cdot |\Delta| \cdot (k + \sum_{i=1}^{k} |F_i|)^3)$ *time and* $\mathbf{O}(|P| \cdot |\Delta| \cdot (k + \sum_{i=1}^{k} |F_i|)^2)$ *space for MPDAs.*

However, the approach going through BPDAs is not optimal. In the rest of this paper, we propose more efficient algorithms.

3 Efficient Solution to the Emptiness Problem

In this section, we propose efficient solutions to the emptiness problems of RPDAs, PPDAs, SPDAs and MPDAs. For this purpose, we first recall the concept of repeating heads and some results for the emptiness problem of BPDAs from [11].

3.1 Repeating Heads

Given an BPDA $\mathcal{P} = (\Sigma, P, \Gamma, \Delta, p_0, \bot, \mathcal{F})$, let $\Longrightarrow^{\mathsf{B}} \subseteq \mathbb{C} \times \Sigma^* \times \mathbb{C}$ be the relation defined as follows: $c \overset{w}{\Longrightarrow}{}^{\mathsf{B}} c'$ if and only if $c \overset{w_1}{\Longrightarrow}{}^{*} c_f \overset{w_2}{\Longrightarrow}{}^{+} c'$, for some $c_f \in \mathcal{F} \times \Gamma^*$ and $w = w_1 w_2$. Intuitively, $c \overset{w}{\Longrightarrow}{}^{\mathsf{B}} c'$ denotes that the BPDA $\mathcal{P}$ can move into the configuration c' when $\mathcal{P}$ is at the configuration c and reads the input word w. Furthermore, a configuration whose first component is an accepting state is visited.

A head $\langle p, \gamma \rangle \in P \times \Gamma$ is *repeating* if there exist some $u \in \Gamma^*, w \in \Sigma^*$ such that $\langle p, \gamma \rangle \overset{w}{\Longrightarrow}{}^{\mathsf{B}} \langle p, \gamma u \rangle$. Let $\mathbf{R}_{\mathsf{B}}$ be the set of repeating heads in $\mathcal{P}$. Let $\mathbf{R}_{\mathsf{B}}\Gamma^*$ denote the set of configurations $\{\langle p, \gamma u \rangle \in \mathbb{C} \mid \langle p, \gamma \rangle \in \mathbf{R}_{\mathsf{B}}, u \in \Gamma^*\}$.

Proposition 1. *[3, 11] The BPDA $\mathcal{P}$ is empty if and only if $\langle p_0, \bot \rangle \notin pre^*(\boldsymbol{R}_{\mathsf{B}}\Gamma^*)$.*

To compute the repeating heads, Esparza et. al. proposed a polynomial-time algorithm which is an adaption of *pre** algorithm.

Lemma 1. *[11] The set $\boldsymbol{R}_{\mathsf{B}}$ of repeating heads of an BPDA $\mathcal{P} = (\Sigma, P, \Gamma, \Delta, p_0, \bot, \mathcal{F})$ can be computed in* $\mathbf{O}(|P|^2 \cdot |\Delta|)$ *time and* $\mathbf{O}(|P| \cdot |\Delta|)$ *space.*

3.2 Efficient Solution to Emptiness Problem of RPDAs

We fix the RPDA $\mathcal{P} = (\Sigma, P, \Gamma, \Delta, p_0, \bot, \mathcal{F})$ with $\mathcal{F} = \{(E_1, F_1), ..., (E_k, F_k)\}$. Our approach is based on the observation that for every accepting run ρ, there is a position $m \geq 0$ and an indicator $i \in [k]$ such that for every $n \geq m$, the configuration $\rho(n)$ is from $(P \setminus E_i) \times \Gamma^*$, namely, the states from E_i will not appear after the position m. In addition, some states from F_i appear infinitely often in ρ after the position m. Therefore, the problem can be reduced to compute heads $\langle p, \gamma \rangle$ the form of $\langle p, \gamma \rangle \stackrel{w}{\Longrightarrow}^+ \langle p, \gamma u \rangle$ such that there exists the indicator $i \in [k]$ with no state from E_i appearing in $\langle p, \gamma \rangle \stackrel{w}{\Longrightarrow}^+ \langle p, \gamma u \rangle$, but some states from F_i appearing in $\langle p, \gamma \rangle \stackrel{w}{\Longrightarrow}^+ \langle p, \gamma u \rangle$. We show that such kind of heads can be computed via applying Lemma 1.

For every $i \in [k]$, let $\Longrightarrow^*_{\bar{i}}$, $\Longrightarrow^+_{\bar{i}} \subseteq \mathbb{C} \times \Sigma^* \times \mathbb{C}$ be the smallest relations such that the following conditions hold:

- $\langle p, u \rangle \stackrel{\epsilon}{\Longrightarrow}^*_{\bar{i}} \langle p, u \rangle$, for every $p \notin E_i$ and $u \in \Gamma^*$;
- $\langle p, u \rangle \stackrel{aw}{\Longrightarrow}^*_{\bar{i}} \langle p'', u'' \rangle$ and $\langle p, u \rangle \stackrel{aw}{\Longrightarrow}^+_{\bar{i}} \langle p'', u'' \rangle$, if $\langle p, u \rangle \stackrel{a}{\Longrightarrow} \langle p', u' \rangle$ with $p, p' \notin E_i$ and $\langle p', u' \rangle \stackrel{w}{\Longrightarrow}^*_{\bar{i}} \langle p'', u'' \rangle$.

Intuitively, $\langle p, u \rangle \stackrel{w}{\Longrightarrow}^*_{\bar{i}} \langle p', u' \rangle$ indicates that the PPDA $\mathcal{P}$ can move into the configuration $\langle p', u' \rangle$ when $\mathcal{P}$ is at the configuration $\langle p, u \rangle$ and reads the input word w. Moveover, no state from E_i appears in the path $\langle p, u \rangle \stackrel{w}{\Longrightarrow}^*_{\bar{i}} \langle p', u' \rangle$.

For every $i \in [k]$, a head $\langle p, \gamma \rangle \in P \times \Gamma$ is *$\bar{i}$-repeating* if $\langle p, \gamma \rangle \stackrel{w_1}{\Longrightarrow}^*_{\bar{i}} \langle p_f, u \rangle \stackrel{w_2}{\Longrightarrow}^+_{\bar{i}} \langle p, \gamma v \rangle$ for some configuration $\langle p_f, u \rangle \in F_i \times \Gamma^*$. For every $i \in [k]$, let $\mathbf{R}^i_{\mathsf{R}}$ be the set of all the $\bar{i}$-repeating heads in $\mathcal{P}$.

Proposition 2. *Let $\boldsymbol{R}_{\mathsf{R}}$ denote the set $\bigcup_{i=1}^k \boldsymbol{R}^i_{\mathsf{R}}$, $\mathcal{P}$ is empty if and only if $\langle p_0, \bot \rangle \notin pre^*(\boldsymbol{R}_{\mathsf{R}} \Gamma^*)$.*

For every $i \in [k]$, let $\mathcal{P}_{\bar{i}} = (\Sigma, P_{\bar{i}}, \Gamma, \Delta_{\bar{i}}, p_0, \bot, F_i)$ be the BPDA, where $P_{\bar{i}} = P \setminus E_i$, $\Delta_{\bar{i}} = \Delta \cap (P_{\bar{i}} \times \Gamma) \times \Sigma \times (P_{\bar{i}} \times \Gamma^*)$, then, $\mathcal{P}_{\bar{i}}$ do not use any state from E_i and we can get that:

Lemma 2. *For each $i \in [k]$, let $\boldsymbol{R}^{\bar{i}}_{\mathsf{B}}$ be the set of repeating heads in the BPDA $\mathcal{P}_{\bar{i}}$, $\boldsymbol{R}^{\bar{i}}_{\mathsf{B}} = \boldsymbol{R}^i_{\mathsf{R}}$.*

Applying Lemma 1, for every $i \in [k]$, $\mathbf{R}^i_{\mathsf{R}}$ can be computed in $\mathbf{O}(|P|^2 \cdot |\Delta|)$ time and $\mathbf{O}(|P| \cdot |\Delta|)$ space. This implies that $\mathbf{R}_{\mathsf{R}} = \bigcup_{i=1}^k \mathbf{R}^i_{\mathsf{R}}$ can be computed $\mathbf{O}(k|P|^2 \cdot |\Delta|)$ time and $\mathbf{O}(|P| \cdot |\Delta|)$ space. Applying Theorem 1, we get that:

Theorem 4. *The emptiness problem of RPDAs $\mathcal{P} = (\Sigma, P, \Gamma, \Delta, p_0, \bot, \mathcal{F})$ with $\mathcal{F} = \{(E_1, F_1), ..., (E_k, F_k)\}$ can be decided in $\mathbf{O}(k|P|^2 \cdot |\Delta|)$ time and $\mathbf{O}(|P| \cdot |\Delta|)$ space.*

3.3 Efficient Solution to Emptiness Problem of SPDAs

Let us fix an SPDA $\mathcal{P} = (\Sigma, P, \Gamma, \Delta, p_0, \bot, \mathcal{F})$ with $\mathcal{F} = \{(E_1, F_1), ..., (E_k, F_k)\}$. We propose an efficient solution to the emptiness problem of SPDAs by guessing the set of states which can be visited after some point. Similar to the transformation from Streett automata to Büchi automata, for every subset S of states, we compute a head $\langle p, \gamma\rangle$ of the form $\langle p, \gamma\rangle \overset{w}{\Longrightarrow}^{+} \langle p, \gamma u\rangle$. Furthermore, only states from S are visited in $\langle p, \gamma\rangle \overset{w}{\Longrightarrow}^{+} \langle p, \gamma u\rangle$ and for every $i \in [k]$, if $F_i \cap S \neq \emptyset$, at least one state from E_i is visited in $\langle p, \gamma\rangle \overset{w}{\Longrightarrow}^{+} \langle p, \gamma u\rangle$. These heads can be computed via applying Lemma 1 on some proper BPDAs. Then, by applying pre^* computation to these heads, we can compute an $\mathcal{P}$-automaton $\mathcal{A}$ recognizing the set of configurations from which the SPDA has an accepting run.

Given a set $S \subseteq P$, let $\mathsf{In}(S) = \{E_i \mid i \in [k], F_i \cap S \neq \emptyset\}$. Let $\{S_1, ..., S_n\} \subseteq 2^P$ be the set of maximum sets such that for every $i, j \in [n]$, $i = j$ if $\mathsf{In}(S_i) = \mathsf{In}(S_j)$, and the range of the function In covers all the subsets of $\{E_1, ..., E_k\}$. We can see that n is at most $2^{\mathsf{min}(k,|P|)}$. Suppose, elements in $\mathsf{In}(S)$ are well-ordered and we use $\mathsf{In}(S, i)$ to denote the i^{th} element.

Let $R^S \subseteq P$ be the repeating heads of the BPDA $\mathcal{P}_S = (\Sigma, S, \Gamma, \Delta', (p_0, 0), \bot, \mathcal{F}')$, where $\mathcal{F}' = S \times \{|\mathsf{In}(S)|\}$ and Δ' is defined as follows: for every $\langle p, \gamma\rangle \overset{a}{\hookrightarrow} \langle p', u'\rangle \in \Delta$ with $p, p' \in S$,

- $\langle (p, i), \gamma\rangle \overset{a}{\hookrightarrow} \langle (p', i + 1), u\rangle \in \Delta'$, for every $i \in \{0, ..., |\mathsf{In}(S)| - 1\}$ with $p' \in \mathsf{In}(S, i + 1)$;
- $\langle (p, i), \gamma\rangle \overset{a}{\hookrightarrow} \langle (p', i), u\rangle \in \Delta'$, for $i \in \{0, ..., |\mathsf{In}(S)| - 1\}$;
- $\langle (p, |\mathsf{In}(S)|), \gamma\rangle \overset{a}{\hookrightarrow} \langle (p', 0), u\rangle \in \Delta'$.

Proposition 3. *Let $\boldsymbol{R_S} = \{\langle p, \gamma\rangle \in P \times \Gamma \mid \exists S' \subseteq P : \langle (p, 0), \gamma\rangle \in R^{S'}\}$, the SPDA $\mathcal{P}$ is empty if and only if $\langle p_0, \bot\rangle \notin pre^*(\boldsymbol{R_S}\Gamma^*)$.*

For each set $S \subseteq P$, the set R^S can be computed in $\mathbf{O}(|S|^2 \cdot |\Delta| \cdot |\mathsf{In}(S)|^3)$ time and $\mathbf{O}(|S| \cdot |\Delta| \cdot |\mathsf{In}(S)|^2)$ space by applying Lemma 1. The number of sets S that should be considered is bounded by $2^{\mathsf{min}(k,|P|)}$. Therefore, $\mathbf{R_S}$ can be computed in $\mathbf{O}(k^3|P|^2 \cdot |\Delta| \cdot 2^{\mathsf{min}(|P|,k)})$ time and $\mathbf{O}(k^2|P| \cdot |\Delta|)$ space. Applying Theorem 1, we get that:

Theorem 5. *The emptiness problem of SPDAs $\mathcal{P} = (\Sigma, P, \Gamma, \Delta, p_0, \bot, \mathcal{F})$ can be decided in $\mathbf{O}(k^3|P|^2 \cdot |\Delta| \cdot 2^{min(|P|,k)})$ time and $\mathbf{O}(k^2|P| \cdot |\Delta|)$ space.*

3.4 Efficient Solution to Emptiness Problem of MPDAs

We fix the MPDA $\mathcal{P} = (\Sigma, P, \Gamma, \Delta, p_0, \bot, \mathcal{F})$ with $\mathcal{F} = \{F_1, ..., F_k\}$. Our approach is based on the observation that the MPDA $\mathcal{P}$ has an accepting run ρ on some infinite word if and only if there is a position $m \geq 0$ and an indicator $i \in [k]$ such that for every $n \geq m$, the set of states in the configurations $\rho(n)$

is the set F_i. This observation allows us to reduce the emptiness problem of the MPDA $\mathcal{P}$ to compute heads $\langle p, \gamma \rangle \in F_i \times \Gamma$ the form of $\langle p, \gamma \rangle \xRightarrow{w}^{+} \langle p, \gamma u \rangle$ such that the set of states appearing in $\langle p, \gamma \rangle \xRightarrow{w}^{+} \langle p, \gamma u \rangle$ is F_i for some indicator $i \in [k]$. The later problem is solved by applying Lemma 1 on a BPDA $\mathcal{P}_i$ which counts the number of visited states in F_i.

Given an index $i \in [k]$, let $\mathcal{P}_i = (\Sigma, F_i \times \{0, ..., |F_i|\}, \Gamma, \Delta_i, (p_0, 0), \perp, F_i \times \{|F_i|\})$ be the BPDA, where Δ_i is defined as follows: for every $\langle p, \gamma \rangle \overset{a}{\hookrightarrow} \langle p', u \rangle \in \Delta$ with $p, p' \in F_i$, every $j \in \{0, ..., |F_i| - 1\}$,

- $\langle (p, j), \gamma \rangle \overset{a}{\hookrightarrow} \langle (p', j+1), u \rangle \in \Delta_i$, if $p' = p_i^{j+1}$;
- $\langle (p, j), \gamma \rangle \overset{a}{\hookrightarrow} \langle (p', j), u \rangle \in \Delta_i$;
- $\langle (p, |F_i|), \gamma \rangle \overset{\epsilon}{\hookrightarrow} \langle (p', 0), u \rangle \in \Delta_i$.

Let $\mathbf{R}_{\mathsf{B}}^i$ be the set of repeating heads in the BPDA $\mathcal{P}_i$, $\mathbf{R}_{\mathsf{M}}$ be the set $\{\langle p, \gamma \rangle \in P \times \Gamma \mid \langle (p, 0), \gamma \rangle \in \mathbf{R}_{\mathsf{B}}^i, i \in [k]\}$.

Proposition 4. *The MPDA $\mathcal{P}$ is empty if and only if $\langle p_0, \perp \rangle \notin pre^*(\boldsymbol{R}_{\mathsf{M}} \Gamma^*)$.*

Applying Lemma 1, the set $\mathbf{R}_{\mathsf{B}}^i$ can be computed in $\mathbf{O}(|F_i|^5 \cdot |\Delta|)$ time and $\mathbf{O}(|\Delta| \cdot |F_i|^3)$ space. This implies that $\mathbf{R}_{\mathsf{M}}$ can be computed in $\mathbf{O}(|\Delta| \cdot \sum_{i \in [k]} |F_i|^5)$ time and $\mathbf{O}(|\Delta| \cdot \max\{|F_i|^3 \mid i \in [k]\})$ space. Applying Theorem 1, we get that:

Theorem 6. *The emptiness problem of MPDAs $\mathcal{P} = (\Sigma, P, \Gamma, \Delta, p_0, \perp, \mathcal{F})$ can be decided in $\mathbf{O}(|\Delta| \cdot \max(|P|^2, \sum_{i \in [k]} |F_i|^5))$ time and $\mathbf{O}(|\Delta| \cdot \max\{|P|, |F_i|^3 \mid i \in [k]\})$ space.*

The emptiness algorithm presented in this section outperforms the one that goes through BPDAs. In particular, in the worst-case when the maximum set in $\mathcal{F}$ is P and $|\mathcal{F}| = 2^{|P|}$, this efficient approach works in $\mathbf{O}(|P|^5 \cdot |\Delta| \cdot 2^{|P|})$ time and $\mathbf{O}(|P|^3 \cdot |\Delta|)$ space, while the later works in $\mathbf{O}(|P|^5 \cdot |\Delta| \cdot 2^{3|P|})$ time and $\mathbf{O}(|P|^3 \cdot |\Delta| \cdot 2^{2|P|})$ space.

4 Model Checking Linear Temporal Logic with Fairness

In this section, we show one of the benefits of our efficient algorithms by applying to model checking linear temporal logic with fairness. We investigate complexity of the model checking problem for linear temporal logic on pushdown systems with fairness. It is better to verify LTL formulae by reducing to the emptiness problem of SPDAs than the problem of BPDAs.

A *pushdown system* (PDS) $\mathcal{P} = (P, \Gamma, \Delta, p_0, \perp)$ is an ω-PDA by omitting the input alphabet Σ and acceptance condition $\mathcal{F}$. The transition relation and runs are defined as for ω-PDAs.

4.1 Linear Temporal Logic and Büchi Automata

Let us fix the PDA $\mathcal{P} = (P, \Gamma, \Delta, p_0, \bot)$ in this section. Let AP be a finite set of atomic propositions

Definition 3. *Formulae of the Linear Temporal Logic (LTL for short) are defined by the following rules:*

$$\phi \ ::= \ \sigma \ \mid \neg\phi \ \mid \ \phi \wedge \phi \ \mid \ X \mid \ \phi U \phi$$

where $\sigma \in AP$*.*

Given an infinite word $w = w_0 w_1 ... \in (2^{\mathsf{AP}})^\omega$, let $w^i = w_i w_{i+1} ...$ be the suffix of w. The satisfaction relation $w \models \phi$ between an infinite word w and an LTL formula ϕ is inductively defined as follows:

$$\begin{aligned}
w \models \sigma &\Longleftrightarrow \sigma \in w_0;\\
w \models \neg\phi &\Longleftrightarrow w \not\models \phi;\\
w \models \phi_1 \wedge \phi_2 &\Longleftrightarrow w \models \phi_1 \text{ and } w \models \phi_2;\\
w \models X\phi &\Longleftrightarrow w^1 \models \phi;\\
w \models \phi_1 U \phi_2 &\Longleftrightarrow \exists i \geq 0 \text{ such that } w^i \models \phi_2 \text{ and } \forall j : 0 \leq j < i,\ w^j \models \phi_1.
\end{aligned}$$

We use abbreviations $F\phi \equiv \mathit{true}U\phi$, $G\phi \equiv \neg F \neg \phi$ and $\phi_1 \rightarrow \phi_2 \equiv \neg\phi_1 \vee \phi_2$. Let $l : \mathsf{AP} \rightarrow 2^P$ be a labeling function that assigns to each atomic proposition a set of states. Given a run $\rho = c_0 c_1 ...$ of $\mathcal{P}$, let $l(\rho)$ be the infinite word w such that for every $i \geq 0$, $w_i = \{\sigma \in \mathsf{AP} \mid c_i \in l(\sigma) \times \Gamma^*\}$. A run $\rho = c_0 c_1 ...$ of $\mathcal{P}$ *satisfies* an LTL formula ϕ if and only if $l(\rho) \models \phi$. A configuration c *satisfies* an LTL formula ϕ if and only if for each run ρ of $\mathcal{P}$ starting from c, ρ satisfies ϕ. Let $\|\phi\|^{\mathcal{P}}$ be the set of configurations that satisfy the formula ϕ.

The *local model checking problem* for LTL is to decide whether $c \in \|\phi\|^{\mathcal{P}}$ for a given LTL formula ϕ and a configuration $c \in \mathbb{C}$, while the *global model checking problem* is to compute $\|\phi\|^{\mathcal{P}}$. In this work, we solve the global model checking using an automata-theoretic approach which reduces the problem to the emptiness problem of ω-PDAs.

Definition 4. *An* Büchi-automaton *(BA)* $\boldsymbol{M}$ *is a tuple* $(G, \Sigma, \delta, g^0, \mathcal{F})$ *where* G *is a finite set of states,* Σ *is the input alphabet,* $\delta : G \times \Sigma \rightarrow 2^G$ *is a transition function,* $g^0 \in G$ *is the initial state and* $\mathcal{F} \subseteq G$ *is the Büchi acceptance.*

A *run* π of $\mathbf{M}$ over an infinite word $\alpha_0 \alpha_1 ... \in \Sigma^\omega$ is an infinite sequence of states $\pi = g_0 g_1 ...$ such that $g_0 = g^0$, and for every $i \geq 0$, $g_{i+1} \in \delta(g_i, \alpha_i)$. Let $\mathsf{Inf}(\pi)$ be the set of states visited infinitely often in π. A run π is *accepting* if and only if $\mathsf{Inf}(\pi) \cap \mathcal{F} \neq \emptyset$.

Theorem 7. *[38] For every LTL formula* ϕ*, we can construct a BA* $\boldsymbol{M} = (G, 2^{AP}, \delta, g^0, F)$ *with* $2^{\mathbf{O}(|\phi|)}$ *states/transitions such that* $\boldsymbol{M}$ *recognizes all of the infinite words satisfying* ϕ*, where* $|\phi|$ *denotes the number of subformulae of* ϕ*.*

Theorem 8. *[11] Given a PDS* $\mathcal{P} = (P, \Gamma, \Delta, p_0, \bot)$ *and an LTL formula* ϕ*,* $\|\phi\|^{\mathcal{P}}$ *can be computed in* $\mathbf{O}(|P|^2 \cdot |\Delta| \cdot 2^{\mathbf{O}(|\phi|)})$ *time and* $\mathbf{O}(|P| \cdot |\Delta| \cdot 2^{\mathbf{O}(|\phi|)})$ *space.*

4.2 Fairness

Fairness is the key of concurrent systems which rules out infinite behaviors that are considered unrealistic, and are often necessary to establish liveness properties frequently encountered problem in concurrent systems. Fairness assumption can be expressed in an LTL formula. In this work, we consider the following fairness assumptions [1]:

- Unconditional fairness: $\Phi \equiv \bigwedge_{1\leq i\leq k} GF\phi_i$,
- Weak fairness: $\Phi \equiv \bigwedge_{1\leq i\leq k} (FG\phi_i \rightarrow GF\psi_i)$,
- Strong fairness: $\Phi \equiv \bigwedge_{1\leq i\leq k} (GF\phi_i \rightarrow GF\psi_i)$,

where ϕ_i and ψ_i are LTL formulae.

Given a PDA $\mathcal{P} = (P, \Gamma, \Delta, p_0, \bot)$ and an LTL formula ϕ with the fairness assumption Φ, then the global model checking problem is to compute $\|\Phi \rightarrow \phi\|^{\mathcal{P}}$. By applying Theorem 8, $\|\Phi \rightarrow \phi\|^{\mathcal{P}}$ can be computed in $\mathbf{O}(|P|^2 \cdot |\Delta| \cdot 2^{\mathbf{O}(|\phi|+7k)})$ time and $\mathbf{O}(|P|\cdot|\Delta|\cdot 2^{\mathbf{O}(|\phi|+7k)})$ space. In the next section, we will present a more efficient approach via reducing the model checking problem to the emptiness problem of SPDAs.

For the sake of simplifying presentation, we assume that the formulae ϕ_i and ψ_i in fairness assumptions are Boolean combinations of atomic propositions. Given a state $p \in P$ and the labeling function l, $p \models \psi_i$ denotes that ψ_1 holds when replacing each atomic proposition σ in ψ_1 by *true* (resp. *false*) if $p \in l(\sigma)$ (resp. $p \notin l(\sigma)$). Our approach can be generalized to cope with general fairness assumptions using regular valuations [12].

It is well-known that weak fairness $\bigwedge_{1\leq i\leq k} (FG\phi_i \rightarrow GF\psi_i)$ can be written into an unconditional fairness $\bigwedge_{1\leq i\leq k} GF(\neg\phi_i \vee \psi_i)$ and unconditional fairness $\bigwedge_{1\leq i\leq k} GF\phi_i$ can be seen as a special form of strong fairness $\bigwedge_{1\leq i\leq k}(GF\ true \rightarrow GF\phi_i)$. Therefore, it is enough to consider strong fairness.

4.3 An Efficient Model Checking Approach for LTL with Fairness

Inspired by the model checking approach for LTL on PDSs which is reduced to the emptiness problem of BPDAs, and fairness Φ can be transformed into Streett acceptance, we can reduce the LTL model checking problem on PDSs with fairness to the emptiness problem of SPDAs.

Fix a PDS $\mathcal{P} = (P, \Gamma, \Delta, p_0, \bot)$ and an LTL formula ϕ with the strong fairness assumption $\Phi \equiv \bigwedge_{1\leq i\leq k}(GF\phi_i \rightarrow GF\psi_i)$. Let $\mathbf{M}_{\neg\phi} = (G, 2^{\mathsf{AP}}, \delta, g^0, F)$ be the NBA that recognizes all of the infinite words satisfying $\neg\phi$. We define a SPDA $\mathcal{P}_{\neg\phi} = (\{a\}, P \times G, \Gamma, \Delta', [p_0, g^0], \bot, \mathcal{F})$, where

- Δ' is defined as: for every $\langle p, \gamma\rangle \hookrightarrow \langle p', u\rangle \in \Delta$ and every $g' \in \delta(g, A)$ with $A = \{\sigma \in \mathsf{AP} \mid p \in l(\sigma)\}$, $\langle [p, g], \gamma\rangle \overset{a}{\hookrightarrow} \langle [p', g'], u\rangle \in \Delta'$,
- $\mathcal{F} = \{(E_1, F_1), ..., (E_{k+1}, F_{k+1})\}$, where for every $i : 1 \leq i \leq k$, $E_i = \{p \in P \mid p \models \psi_i\} \times G$ and $F_i = \{p \in P \mid p \models \phi_i\} \times G$, $F_{k+1} = P$ and $E_{k+1} = F$.

Lemma 3. *For every configuration $\langle p, u\rangle \in P \times \Gamma^*$, $\langle p, u\rangle \notin \|\Phi \rightarrow \phi\|^{\mathcal{P}}$ if and only if $\mathcal{P}_{\neg\phi}$ has an accepting run starting from $\langle [p, g^0], u\rangle$.*

Proof. Suppose $\langle p,u\rangle \notin \|\Phi \to \phi\|^{\mathcal{P}}$, then $\mathcal{P}$ has a run ρ starting from $\langle p,u\rangle$ such that $\rho \models \Phi$, but $\rho \not\models \phi$. The later implies that $\rho \models \neg\phi$, i.e., $\mathbf{M}_{\neg\phi}$ has an accepting run π on the infinite word $l(\rho)$. Let $\rho_\pi \in (P \times G)^\omega$ be the run such that the projection of ρ_π on P is the run ρ, the projection of ρ_π on G is π. It is easy to see that ρ_π is an accepting run of $\mathcal{P}_{\neg\phi}$.

Suppose $\mathcal{P}_{\neg\phi}$ has an accepting run ρ starting from $\langle [p,g^0],u\rangle$, let ρ_P be the projection of ρ on P and ρ_G be the projection of ρ on G. For every $i : 1 \leq i \leq k$, since either $F_i \cap \mathsf{Inf}(\rho) = \emptyset$ or $E_i \cap \mathsf{Inf}(\rho) \neq \emptyset$ holds, then $\rho_P \models \Phi$. On the other hand, it is easy to verify that ρ_G is an accepting run of $\mathcal{P}_{\neg\phi}$ over the infinite word $l(\rho_P)$ and ρ_P is a run of $\mathcal{P}$. The result immediately follows. $\square$

Applying Theorem 5, we get that:

Theorem 9. *Given an LTL formula ϕ and a PDS $\mathcal{P} = (P, \Gamma, \Delta, p_0, \perp)$ with the fairness Φ, $\|\Phi \to \phi\|^{\mathcal{P}}$ can be computed in*

- $\mathbf{O}((k+1)^3|P|^2 \cdot |\Delta| \cdot 2^{\mathbf{O}(|\phi|)+\min(|P|,k+1)})$ *time and* $\mathbf{O}((k+1)^2|P| \cdot |\Delta| \cdot 2^{\mathbf{O}(|\phi|)})$ *space if Φ is a strong fairness,*
- $\mathbf{O}((k+1)^3|P|^2 \cdot |\Delta| \cdot 2^{\mathbf{O}(|\phi|)})$ *time and* $\mathbf{O}((k+1)^2|P| \cdot |\Delta| \cdot 2^{\mathbf{O}(|\phi|)})$ *space if Φ is a weak or unconditional fairness.*

The complexity for weak/unconditional fairness is obtained by a careful analysis of the algorithm for emptiness checking of SPDAs and the fact that the Streett acceptance will be $\mathcal{F} = \{(E_1, P \times G), ..., (E_{k+1}, P \times G)\}$, where for every $i : 1 \leq i \leq k$, $E_i = \{p \in P \mid p \models \psi_i\} \times G$ and $E_{k+1} = F$ (which is a generalized Büchi acceptance).

5 Conclusion and Future Work

In this paper, we proposed direct and efficient algorithms for the emptiness problems of PPDAs, RPDAs, SPDAs and MPDAs, which are crucial for model checking recursive programs with fairness. In these recursive programs with fairness, one thread is recursive, the other threads are finite and all the threads are executed in some "fair" manner.

Future work includes the implementation of proposed algorithms, investigation of ω-PDAs in which the sets in acceptance conditions are sets of configurations rather than states, and investigation of the emptiness problem of alternating ω-PDAs (ω-APDAs). Although, its emptiness problem is undecidable [4] in general, the problem for alphabet-free ω-APDAs with Büchi or Parity acceptance were already studied in many work [32, 6, 19, 39, 31, 16]. There is no direct approach to the emptiness problem for alphabet-free ω-APDAs with other acceptances. It is also interesting to investigate efficient algorithms for the emptiness problem of alphabet-free ω-APDAs with related acceptances, which are related to the pushdown model checking problem against branching-time temporal logic with fairness.

References

1. Christel Baier and Joost-Pieter Katoen. *Principles of model checking*. MIT Press, 2008.
2. Roderick Bloem, Harold N. Gabow, and Fabio Somenzi. An algorithm for strongly connected component analysis in $n \log n$ symbolic steps. *Formal Methods in System Design*, 28(1):37–56, 2006.
3. Ahmed Bouajjani, Javier Esparza, and Oded Maler. Reachability Analysis of Pushdown Automata: Application to Model Checking. In *CONCUR'97*. LNCS 1243, 1997.
4. Laura Bozzelli. Complexity results on branching-time pushdown model checking. *Theor. Comput. Sci.*, 379(1-2):286–297, 2007.
5. Julius Richard Büchi. On a decision method in restricted second order arithmetic. In *Proceedings of the International Congress on Logic, Method and Philos. Sci.*, pages 1–11, 1960.
6. Thierry Cachat. Symbolic strategy synthesis for games on pushdown graphs. In *ICALP*, volume 2380 of *Lecture Notes in Computer Science*, pages 704–715. Springer, 2002.
7. Rina S. Cohen and Arie Y. Gold. Theory of omega-languages. I. characterizations of omega-context-free languages. *J. Comput. Syst. Sci.*, 15(2):169–184, 1977.
8. Rina S. Cohen and Arie Y. Gold. Theory of omega-languages. II. A study of various models of omega-type generation and recognition. *J. Comput. Syst. Sci.*, 15(2):185–208, 1977.
9. Rina S. Cohen and Arie Y. Gold. Omega-computations on deterministic pushdown machines. *J. Comput. Syst. Sci.*, 16(3):275–300, 1978.
10. E. Allen Emerson and Chin-Laung Lei. Modalities for model checking: Branching time strikes back. In *POPL*, pages 84–96, 1985.
11. Javier Esparza, David Hansel, Peter Rossmanith, and Stefan Schwoon. Efficient algorithm for model checking pushdown systems. In *CAV*, volume 1885 of *LNCS*, 2000.
12. Javier Esparza, Antonín Kucera, and Stefan Schwoon. Model checking LTL with regular valuations for pushdown systems. *Inf. Comput.*, 186(2):355–376, 2003.
13. Javier Esparza and Stefan Schwoon. A BDD-based model checker for recursive programs. In *CAV*, LNCS, pages 324–336, 2001.
14. A. Finkel, B. Willems, and P. Wolper. A Direct Symbolic Approach to Model Checking Pushdown Systems. In *Infinity'97, ENTCS 9*. Elsevier Sci. Pub., 1997.
15. Matthew Hague and C.-H. Luke Ong. Winning regions of pushdown parity games: A saturation method. In *Proceedings of the 20th International Conference on Concurrency Theory*, pages 384–398, 2009.
16. Matthew Hague and C.-H. Luke Ong. A saturation method for the modal μ-calculus over pushdown systems. *Inf. Comput.*, 209(5):799–821, 2011.
17. Monika Rauch Henzinger and Jan Arne Telle. Faster algorithms for the nonemptiness of Streett automata and for communication protocol pruning. In *SWAT*, pages 16–27, 1996.
18. Valerie King, Orna Kupferman, and Moshe Y. Vardi. On the complexity of parity word automata. In *FOSSACS*, pages 276–286, 2001.
19. Orna Kupferman, Nir Piterman, and Moshe Y. Vardi. Pushdown specifications. In *Proceedings of the 9th International Conference on Logic for Programming, Artificial Intelligence, and Reasoning (LPAR)*, pages 262–277, 2002.
20. Matti Linna. *On [omega]-words and [omega]-computations*. Turun yliopisto, 1975.

21. Matti Linna. On omega-sets associated with context-free languages. *Information and Control*, 31(3):272–293, 1976.
22. Matti Linna. A decidability result for deterministic *omega*-context-free languages. *Theor. Comput. Sci.*, 4(1):83–98, 1977.
23. Christof Löding. *Methods for the Transformation of ω-Automata: Complexity and Connection to Second Order Logic*. PhD thesis, Christian-Albrechts-University of Kiel, 1998.
24. Christof Löding. Optimal bounds for transformations of omega-automata. In *FSTTCS*, pages 97–109, 1999.
25. Robert McNaughton. Testing and generating infinite sequences by a finite automaton. *Information and Control*, 9(5):521–530, 1966.
26. Andrzej Wlodzimierz Mostowski. Regular expressions for infinite trees and a standard form of automata. In *STOC*, pages 157–168, 1984.
27. David E. Muller. Infinite sequences and finite machines. In *SWCT(FOCS)*, pages 3–16, 1963.
28. Nir Piterman. From nondeterministic Büchi and Streett automata to deterministic parity automata. *Logical Methods in Computer Science*, 3(3), 2007.
29. Amir Pnueli. The temporal logic of programs. In *FOCS*, pages 46–57, 1977.
30. Michael O. Rabin. Decidability of second-order theories and automata on infinite trees. *Transactions of the American Mathematical Society*, pages 1–35, 1969.
31. Olivier Serre. Note on winning positions on pushdown games with [omega]-regular conditions. *Inf. Process. Lett.*, 85(6):285–291, 2003.
32. Fu Song and Tayssir Touili. Efficient CTL model-checking for pushdown systems. In *CONCUR*, 2011.
33. Fu Song and Tayssir Touili. Pumoc: a CTL model-checker for sequential programs. In *IEEE/ACM International Conference on Automated Software Engineering, ASE'12, Essen, Germany, September 3-7, 2012*, pages 346–349, 2012.
34. Fu Song and Tayssir Touili. Efficient CTL model-checking for pushdown systems. *Theor. Comput. Sci.*, 549:127–145, 2014.
35. Robert S. Streett. Propositional dynamic logic of looping and converse is elementary decidable. *Information and Control*, pages 121–141, 1982.
36. Wolfgang Thomas. Automata on infinite objects. In *Handbook of Theoretical Computer Science, Volume B: Formal Models and Sematics (B)*, pages 133–192. 1990.
37. Moshe Y. Vardi. Verification of concurrent programs: The automata-theoretic framework. In *Proceedings of the Symposium on Logic in Computer Science*, pages 167–176, 1987.
38. Moshe Y. Vardi and Pierre Wolper. Automata-theoretic techniques for modal logics of programs. *J. Comput. Syst. Sci.*, 32(2):183–221, 1986.
39. Igor Walukiewicz. Pushdown processes: Games and model checking. In *CAV*, pages 62–74, 1996.

A Appendix

Let $|\langle p, u\rangle|$ denote $|u|$ for $p \in P, u \in \Gamma^*$.

Theorem 10. *Given an RPDA $\mathcal{P} = (\Sigma, P, \Gamma, \Delta, p_0, \bot, \mathcal{F})$ with $\mathcal{F} = \{(E_1, F_1), ..., (E_k, F_k)\}$, one can construct an equivalent BPDA $\mathcal{P}' = (\Sigma, P', \Gamma, \Delta', p_0, \bot, \mathcal{F}')$ such that $\mathcal{L}(\mathcal{P}) = \mathcal{L}(\mathcal{P}')$, where $|P'| = \mathbf{O}(k|P|)$ and $|\Delta'| = \mathbf{O}(k|\Delta|)$.*

Proof. $\mathcal{P}'$ consists of $k+1$ copies of $\mathcal{P}$ in which one copy is full behaving the same as $\mathcal{P}$. The other k copies are partial and each of them is associated to a pair (E_i, F_i), its states are labeled by i and it contains the states from $P \setminus E_i$. A run of $\mathcal{P}'$ starting from the initial configuration $\langle p_0, \bot\rangle$ initially runs at the full copy. During the run, $\mathcal{P}'$ can nondeterministically choose between staying in the full copy or moving to one of the other k copies which guesses a proper pair (E_i, F_i) in order to have an accepting run in $\mathcal{P}$. Once, $\mathcal{P}'$ enters a copy associated to a pair (E_i, F_i), the run will not go into any configurations with states in E_i. The Büchi acceptance $\mathcal{F}' = \bigcup_{i=1}^{k} F_i \times \{i\}$ ensures that the run of $\mathcal{P}'$ will eventually leave the full copy and move to some configurations labeled by i. After that, it will visit some configurations with states in F_i infinitely often.

Formally, for every $i \in [k]$, let $P_i = (P \setminus E_i) \times \{i\}$. We define $P' = P \cup \bigcup_{i=1}^{k} P_i$, $\mathcal{F}' = \bigcup_{i=1}^{k} F_i \times \{i\}$, Δ' is the smallest transition relation such that the following conditions hold:

- For every $\langle p, \gamma\rangle \overset{a}{\hookrightarrow} \langle p', u\rangle \in \Delta$, $\langle p, \gamma\rangle \overset{a}{\hookrightarrow} \langle p', u\rangle \in \Delta'$ and for every $i \in [k]$ with $p' \notin E_i$, $\langle p, \gamma\rangle \overset{a}{\hookrightarrow} \langle (p', i), u\rangle \in \Delta'$;
- For every $i \in [k]$ and $\langle p, \gamma\rangle \overset{a}{\hookrightarrow} \langle p', u\rangle \in \Delta$ such that $p, p' \notin E_i$, $\langle (p, i), \gamma\rangle \overset{a}{\hookrightarrow} \langle (p', i), u\rangle \in \Delta'$.

It is easy to verify that $\mathcal{L}(\mathcal{P}) = \mathcal{L}(\mathcal{P}')$. □

Theorem 11. *Given an PPDA $\mathcal{P} = (\Sigma, P, \Gamma, \Delta, p_0, \bot, \mathcal{F})$ with $\mathcal{F} : P \to [k]$, one can construct an equivalent BPDA $\mathcal{P}' = (\Sigma, P', \Gamma, \Delta', p_0, \bot, \mathcal{F}')$ such that $\mathcal{L}(\mathcal{P}) = \mathcal{L}(\mathcal{P}')$, $|P'| = \mathbf{O}(k|P|)$ and $|\Delta'| = \mathbf{O}(k|\Delta|)$.*

Proof. W.l.o.g., we assume that $k = 2m+1$ for some natural number m. For the Parity acceptance $\mathcal{F} : P \to [k]$, we can construct the Rabin chain $\mathcal{F}' = \{(E_1, F_1), ..., (E_m, F_m)\}$ such that for every $i \in [m]$,

$$E_i = \{p \in P \mid \mathcal{F}(p) \leq 2i-1\} \text{ and } F_i = \{p \in P \mid \mathcal{F}(p) \leq 2i\}.$$

Let $\mathcal{P}' = (\Sigma, P, \Gamma, \Delta, p_0, \bot, \mathcal{F}')$ be the RPDA. Hence, $\mathcal{L}(\mathcal{P}) = \mathcal{L}(\mathcal{P}')$. We can construct an BPDA $\mathcal{P}'' = (\Sigma, P', \Gamma, \Delta', p_0, \bot, \mathcal{F}'')$ such that $\mathcal{L}(\mathcal{P}) = \mathcal{L}(\mathcal{P}'')$, where $|P'| = \mathbf{O}(k|P|)$ and $|\Delta'| = \mathbf{O}(k|\Delta|)$. □

Theorem 12. *Given an SPDA $\mathcal{P} = (\Sigma, P, \Gamma, \Delta, p_0, \bot, \mathcal{F})$ with $\mathcal{F} = \{(E_1, F_1), ..., (E_k, F_k)\}$, we can construct an equivalent BPDA $\mathcal{P}' = (\Sigma, P', \Gamma, \Delta', p_0, \bot, \mathcal{F}')$ such that $\mathcal{L}(\mathcal{P}) = \mathcal{L}(\mathcal{P}')$, where $|P'| = \mathbf{O}(k|P| \cdot 2^{\mathsf{min}(k,|P|)})$ and $|\Delta'| = \mathbf{O}(k|\Delta| \cdot 2^{\mathsf{min}(k,|P|)})$.*

Proof. We guess a subset $S \subseteq P$ of states such that only states in S will be visited after some point. To meet the Streett acceptance, we only need to ensure that for every $i \in [k]$, if some state in S appears in F_i, then some state from E_i should be infinitely often visited. Intuitively, suppose the state $p \in S \cap F_i$, then some state in E_i should be visited infinitely often no matter p is visited infinitely often or not. This acceptance is stronger than the Streett acceptance and can be regarded as generalized Büchi acceptance which can be transformed into Büchi acceptance via de-generalization.

Formally, we define $P' = P \cup P \times \{S_1, ..., S_n\} \times \{0, 1, ..., k\}$, $\mathcal{F}' = \{(p, S_i, |\mathsf{In}(S_i)|) \mid p \in S_i, i \in [n]\}$, Δ' is the smallest transition relation such that the following conditions hold:

- For every $\langle p, \gamma \rangle \overset{a}{\hookrightarrow} \langle p', u \rangle \in \Delta$,
  - $\langle p, \gamma \rangle \overset{a}{\hookrightarrow} \langle p', u \rangle \in \Delta'$ and
  - $\langle p, \gamma \rangle \overset{a}{\hookrightarrow} \langle (p', S_i, 0), u \rangle \in \Delta'$ for every $i \in [n]$ with $p' \in S_i$;
- For every $i \in [n]$, every $\langle p, \gamma \rangle \overset{a}{\hookrightarrow} \langle p', u \rangle \in \Delta$ with $p, p' \in S_i$ and every $j \in \{0, ..., |\mathsf{In}(S_i)| - 1\}$,
  - $\langle (p, S_i, j), \gamma \rangle \overset{a}{\hookrightarrow} \langle (p', S_i, j+1), u \rangle \in \Delta'$, if $p' \in \mathsf{In}(S_i, j+1)$,
  - $\langle (p, S_i, j), \gamma \rangle \overset{a}{\hookrightarrow} \langle (p', S_i, j), u \rangle \in \Delta'$, if $p' \notin \mathsf{In}(S_i, j+1)$,
  - $\langle (p, S_i, |\mathsf{In}(S_i)|), \gamma \rangle \overset{a}{\hookrightarrow} \langle (p', S_i, 0), u \rangle \in \Delta'$.

In the states P', j is a counter to count the number of sets of $\mathsf{In}(S)$ that have been visited, as done in de-generalization. At least one state in each set of $\mathsf{In}(S)$ is infinitely often visited if and only if the counter is set to $|\mathsf{In}(S)|$ infinitely often. It is easy to verify that $\mathcal{L}(\mathcal{P}) = \mathcal{L}(\mathcal{P}')$.

□

Theorem 13. *Given an MPDA $\mathcal{P} = (\Sigma, P, \Gamma, \Delta, p_0, \bot, \mathcal{F})$ with $\mathcal{F} = \{F_1, ..., F_k)\}$, one can construct an equivalent BPDA $\mathcal{P}' = (\Sigma, P', \Gamma, \Delta', p_0, \bot, \mathcal{F}')$ such that $\mathcal{L}(\mathcal{P}) = \mathcal{L}(\mathcal{P}')$, where $|P'| = \mathbf{O}(|P|(k + \|\mathcal{F}\|))$ and $|\Delta'| = \mathbf{O}(|\Delta|(k + \|\mathcal{F}\|))$.*

Proof. For every $i \in [k]$, suppose $F_i = \{p_i^1, ..., p_i^{|F_i|}\}$. $\mathcal{P}'$ guesses the set F_i and the position in the accepting run of $\mathcal{P}$ from which only configurations with states in F_i are visited. Runs of $\mathcal{P}'$ starting from the initial configuration $\langle p_0, \bot \rangle$ runs as in $\mathcal{P}$. The runs can nondeterministically choose between staying in the copy of $\mathcal{P}$ or moving to one of the other copies labeled by i which guesses a proper set F_i. When moving to a copy labeled by i, the runs set a counter j to 0 and increase the counter by 1 when the state p_i^{j+1} is visited. The counter j will be reset to 0 once j is $|F_i|$ meaning that all the states from F_i are visited. If the counter is infinitely often to $|F_i|$, then the Muller acceptance is met.

Formally, $P' = P \cup \{(p, j, i) \mid p \in F_i, 0 \le j \le |F_i|, i \in [k]\}$, $\mathcal{F}' = \{(p, |F_i|, i) \mid p \in F_i, i \in [k]\}$ and Δ' is the smallest transition relation such that the following conditions hold: for every $\langle p, \gamma \rangle \overset{a}{\hookrightarrow} \langle p', u \rangle \in \Delta$,

- $\langle p, \gamma \rangle \overset{a}{\hookrightarrow} \langle p', u \rangle \in \Delta'$;

- $\langle p, \gamma \rangle \overset{a}{\hookrightarrow} \langle (p', 0, i), u \rangle \in \Delta'$ for every $i \in [k]$ such that $p' \in F_i$;
- $\langle (p, j, i), \gamma \rangle \overset{a}{\hookrightarrow} \langle (p', j+1, i), u \rangle \in \Delta'$, if $p \in F_i$ and $p' = p_i^{j+1}$;
- $\langle (p, j, i), \gamma \rangle \overset{a}{\hookrightarrow} \langle (p', j, i), u \rangle \in \Delta'$, if $p, p' \in F_i, p' \neq p_i^{j+1}$ and $j < |F_i|$;
- $\langle (p, |F_i|, i), \gamma \rangle \overset{a}{\hookrightarrow} \langle (p', 0, i), u \rangle \in \Delta'$, if $p, p' \in F_i$.

We can see that $\mathcal{L}(\mathcal{P}) = \mathcal{L}(\mathcal{P}')$. □

Proposition 5. *The RPDA $\mathcal{P}$ is empty if and only if $\langle p_0, \bot \rangle \notin pre^*(\boldsymbol{R}_{\mathsf{R}} \Gamma^*)$.*

Proof. ($\Rightarrow$) Suppose that $\mathcal{P}$ has an accepting run $\rho = c_0 c_1 ...$ on the infinite word $w = a_0 a_1 ...$, where $c_0 = \langle p_0, \bot \rangle$. We can construct a subsequence $c_{n_1} c_{n_2} ...$ of ρ such that

$$|c_{n_1}| = \mathsf{min}\{|c_j| \mid j \geq 0\},$$
$$|c_{n_i}| = \mathsf{min}\{|c_j| \mid j > n_{i-1}\}, \forall i \geq 2.$$

We can see that the sequence $|c_{n_1}|, |c_{n_2}|, ...,$ is strictly increasing meaning that for every $i \geq 1$, once the configuration c_{n_i} is reached, the rest of the run ρ from the position n_i will never change the stack content in c_{n_i} except for the topmost of the stack. Since the sets P and Γ are finite (hence the number of heads is finite), there must exist a pair $\langle p, \gamma \rangle$ that appears in $c_{n_1} c_{n_2} ...$ infinitely often. Therefore, we can construct a subsequence $c_{j_1} c_{j_2} ...$ of $c_{n_1} c_{n_2} ...$ such that for every $i \geq 1$, the head of c_{j_i} is $\langle p, \gamma \rangle$.

Since $\rho = c_0 c_1 ...$ is an accepting run, let (E_m, F_m) be the pair such that $\mathsf{Inf}(\rho) \cap F_m \neq \emptyset$ and $\mathsf{Inf}(\rho) \cap E_m = \emptyset$. By the above construction, there exist a position $j_e \geq j_1$ and a configuration $\langle p_f, u u_1' \rangle \in F_m \times \Gamma^*$ such that

$$c_{j_e} = \langle p, \gamma u_1 \rangle \overset{w_1}{\Longrightarrow}{}_m^* \langle p_f, u u_1' \rangle \overset{w_2}{\Longrightarrow}{}_m^+ \langle p, \gamma u_2 u_1'' \rangle = c_{j_{e+1}}.$$

Since once the configuration c_{j_e} is reached, the rest of the run ρ from the position j_e will never change the stack content in c_{j_e} except for the topmost of the stack, we get that $u_1 = u_1' = u_1''$ and

$$\langle p, \gamma \rangle \overset{w_1}{\Longrightarrow}{}_m^* \langle p_f, u \rangle \overset{w_2}{\Longrightarrow}{}_m^+ \langle p, \gamma u_2 \rangle.$$

Therefore, $\langle p, \gamma \rangle \in \mathbf{R}_{\mathsf{R}}$. The result immediately follows.

($\Leftarrow$) Suppose that $\langle p_0, \bot \rangle \in pre^*(\mathbf{R}_{\mathsf{R}} \Gamma^*)$. Let $\langle p, \gamma \rangle \in \mathbf{R}_{\mathsf{R}}$ be the head such that

$$\langle p_0, \bot \rangle \overset{w_0}{\Longrightarrow}{}^* \langle p, \gamma u \rangle \text{ and } \langle p, \gamma \rangle \overset{w_1}{\Longrightarrow}{}_i^* \langle p_f, u' \rangle \overset{w_2}{\Longrightarrow}{}_i^+ \langle p, \gamma u'' \rangle$$

for some $i \in [k]$, $p_f \in F_i$. Therefore, we get the following accepting run:

$$\langle p_0, \bot \rangle \overset{w_0}{\Longrightarrow}{}^* \langle p, \gamma u \rangle \overset{w_1}{\Longrightarrow}{}^* \langle p_f, u' u \rangle \overset{w_2}{\Longrightarrow}{}^+ \langle p, \gamma u'' u \rangle \overset{w_1}{\Longrightarrow}{}^* \langle p_f, u' u'' u \rangle \overset{w_2}{\Longrightarrow}{}^+ \langle p, \gamma u'' u'' u \rangle \cdots .$$

Note that no state from E_i is visited in the path $\langle p, \gamma v u \rangle \overset{w_1}{\Longrightarrow}{}^* \langle p_f, u' v u \rangle \overset{w_2}{\Longrightarrow}{}^+ \langle p, \gamma u'' v u \rangle$ for every $v \in \{u''^i \mid i \geq 0\}$.

□

Proposition 6. *Let* $\boldsymbol{R}_{\mathsf{S}} = \{\langle p, \gamma\rangle \in P \times \Gamma \mid \exists S' \subseteq P : \langle (p, 0), \gamma\rangle \in R^{S'}\}$*, the SPDA* $\mathcal{P}$ *is empty if and only if* $\langle p_0, \perp\rangle \notin pre^*(\boldsymbol{R}_{\mathsf{S}}\Gamma^*)$.

Proof. ($\Rightarrow$) Suppose $\mathcal{P}$ has an accepting run $\rho = c_0c_1...$ starting from $\langle p_0, \perp\rangle$ on some infinite word. Let $\{k_1, ..., k_n\} \subseteq [k]$ be the set of indices such that for every $i \in [k]$, $i \in \{k_1, ..., k_n\}$ if and only if $\mathsf{Inf}(\rho) \cap F_i \neq \emptyset$. Therefore, $\mathsf{Inf}(\rho) \cap F_i = \emptyset$ for every $i \in [k] \setminus \{k_1, ..., k_n\}$ and $\mathsf{Inf}(\rho) \cap E_i \neq \emptyset$ for every $i \in \{k_1, ..., k_n\}$. Let S be the set of states $P \setminus (\bigcup_{i \in [k] \setminus \{k_1,...,k_n\}} F_i)$. Since the number of states is finite, there exists a configuration c_m in ρ such that for every $j \geq m$, $c_j \in S \times \Gamma^*$. Let $\rho_{\geq m}$ be the suffix $c_m c_{m+1}...$ of ρ. We can construct a subsequence $c_{n_1}c_{n_2}...$ of $\rho_{\geq m}$ such that

$$|c_{n_1}| = \mathsf{min}\{|c_j| \mid j \geq m\},$$
$$|c_{n_i}| = \mathsf{min}\{|c_j| \mid j > n_{i-1}\}, \forall i \geq 2.$$

Since the sets S and Γ are finite (i.e., the number of heads is finite), there must exist a pair $\langle p, \gamma\rangle \in S \times \Gamma$ that appears in $c_{n_1}c_{n_2}...$ infinitely often. Therefore, we can construct a subsequence $c_{m_1}c_{m_2}...$ of $c_{n_1}c_{n_2}...$ such that for every $i \geq 1$, the head of c_{m_i} is $\langle p, \gamma\rangle$.

By the above construction, there exist two positions $m_{e'} > m_e \geq m_1$ such that there is a subsequence $\langle p^{k_1}, u^{k_1}\rangle...\langle p^{k_n}, u^{k_n}\rangle$ of $c_{m_e}c_{m_e+1}...c_{m_{e'}}$ with $p^{k_1} \in E_{k_1}, ..., p^{k_n} \in E_{k_n}$. Moreover, for every $i \in [k]$, $\langle p^{k_i}, u^{k_i}\rangle$ is the first configuration from $E_{k_i} \times \Gamma^*$ that occurs in the path $\langle p^{k_{i-1}}, u^{k_{i-1}}\rangle...\langle p^{k_i}, u^{k_i}\rangle$. W.l.o.g., we assume that $c_{m_e} = \langle p, \gamma u_1\rangle \neq \langle p^{k_1}, u^{k_1}\rangle$ and $c_{m_{e'}} = \langle p, \gamma u_2 u_1\rangle \neq \langle p^{k_n}, u^{k_n}\rangle$. Therefore, we get that $\mathcal{P}_S$ has

$$\langle (p, 0), \gamma u_1\rangle \overset{w_1}{\Longrightarrow}^+ \langle (p^{k_n}, |\mathsf{In}(S)|), u^{k_n}\rangle \overset{w_2}{\Longrightarrow}^+ \langle (p, 0), \gamma u_2 u_1'\rangle.$$

Since once the configuration c_{m_e} is reached, the rest of the run ρ from the position m_e will never change the stack content in c_{m_e} except for the topmost of the stack, we get that $u_1 = u_1'$ and $\mathcal{P}_S$ has

$$\langle (p, 0), \gamma\rangle \overset{w_1}{\Longrightarrow}^+ \langle (p^{k_n}, |\mathsf{In}(S)|), u'\rangle \overset{w_2}{\Longrightarrow}^+ \langle (p, 0), \gamma u_2\rangle.$$

Therefore, $\langle p, \gamma\rangle \in \mathbf{R}_{\mathsf{S}}$. It immediately follows that $\langle p_0, \perp\rangle \in pre^*(\mathbf{R}_{\mathsf{S}}\Gamma^*)$.

($\Leftarrow$) Suppose $\langle p_0, \perp\rangle \in pre^*(\mathbf{R}_{\mathsf{S}}\Gamma^*)$, let $\langle p, \gamma\rangle \in \mathbf{R}_{\mathsf{S}}$ be the head such that $\langle p_0, \perp\rangle \overset{w_1}{\Longrightarrow}^* \langle p, \gamma u\rangle$ and $\mathcal{P}_S$ has $\langle (p, 0), \gamma\rangle \overset{w_2}{\Longrightarrow}^+ \langle (p', |\mathsf{In}(S)|), u'\rangle \overset{w_3}{\Longrightarrow}^+ \langle (p, 0), \gamma u_2\rangle$ for some set $S \subseteq P$. The following run is an accepting run ρ:

$$\langle p_0, \perp\rangle \overset{w_1}{\Longrightarrow}^* \langle p, \gamma u\rangle \overset{w_2}{\Longrightarrow}^+ \langle p', u'u\rangle \overset{w_3}{\Longrightarrow}^+ \langle p, \gamma u_2 u\rangle \overset{w_2}{\Longrightarrow}^+ \langle p', u'uu_2u\rangle \overset{w_3}{\Longrightarrow}^+ \langle p, \gamma u_2 u_2 u\rangle \cdots.$$

Notice that for every $i \in [k]$, if $\mathsf{Inf}(\rho) \cap F_i \neq \emptyset$, then $E_i \in \mathsf{In}(S)$ which implies that $E_i \cap \mathsf{Inf}(\rho) \neq \emptyset$. $\square$

Proposition 7. *The MPDA* $\mathcal{P}$ *is empty if and only if* $\langle p_0, \perp\rangle \notin pre^*(\boldsymbol{R}_{\mathsf{M}}\Gamma^*)$.

Proof. ($\Rightarrow$) Suppose that $\mathcal{P}$ has an accepting run $\rho = c_0c_1...$ on the infinite word $w = a_0a_1...$, where $c_0 = \langle p_0, \bot\rangle$. Let $F_m \in \mathcal{F}$ be the set of states such that $\mathsf{Inf}(\rho) = F_m$. Then, there exists a position $m' \geq 0$ such that the states appearing in the sequence $\rho_{\geq m'} = c_{m'}c_{m'+1}...$ come from F_m. We can construct a subsequence $c_{n_1}c_{n_2}...$ of $\rho_{\geq m'}$ such that

$$|c_{n_1}| = \mathsf{min}\{|c_j| \mid j \geq m'\},$$
$$|c_{n_i}| = \mathsf{min}\{|c_j| \mid j > n_{i-1}\}, \forall i \geq 2.$$

Since the sets F_m and Γ are finite (i.e., the number of heads is finite), there must exist a pair $\langle p, \gamma\rangle \in F_m \times \Gamma$ that appears in $c_{n_1}c_{n_2}...$ infinitely often. Therefore, we can construct a subsequence $c_{j_1}c_{j_2}...$ of $c_{n_1}c_{n_2}...$ such that for every $i \geq 1$, the head of c_{j_i} is $\langle p, \gamma\rangle$.

By the above construction, there exist two positions $j_{e'} > j_e \geq j_1$ such that $\mathcal{P}_m$ has

$$c_{j_e} = \langle (p, 0), \gamma u_1\rangle \overset{w_1}{\Longrightarrow}{}^{+} \langle (p', |F_m|), vu_1\rangle \overset{w_2}{\Longrightarrow}{}^{+} \langle (p, 0), \gamma u_2 u_1'\rangle = c_{j_{e'}}.$$

Since once the configuration c_{j_e} is reached, the rest of the run ρ from the position j_e will never change the stack content in c_{j_e} except for the topmost of the stack, we get that $u_1 = u_1'$ and $\mathcal{P}_m$ has

$$\langle (p, 0), \gamma\rangle \overset{w_1}{\Longrightarrow}{}^{+} \langle (p', |F_m|), v\rangle \overset{w_2}{\Longrightarrow}{}^{+} \langle (p, 0), \gamma u_2\rangle.$$

Therefore, $\langle p, \gamma\rangle \in \mathbf{R}_{\mathsf{M}}$. The result immediately follows.

($\Leftarrow$) Suppose $\langle p_0, \bot\rangle \in pre^*(\mathbf{R}_{\mathsf{M}}\Gamma^*)$. Let $\langle p, \gamma\rangle \in \mathbf{R}_{\mathsf{M}}$ be the head such that

$$\langle p_0, \bot\rangle \overset{w_0}{\Longrightarrow}{}^{*} \langle p, \gamma u\rangle \text{ and } \mathcal{P}_i \text{ has } \langle (p, 0), \gamma\rangle \overset{w_1}{\Longrightarrow}{}^{+} \langle (p', |F_i|), v\rangle \overset{w_2}{\Longrightarrow}{}^{+} \langle (p, 0), \gamma u'\rangle$$

for some $i \in [k]$. Therefore, we get the following run:

$$\langle p_0, \bot\rangle \overset{w_0}{\Longrightarrow}{}^{*} \langle p, \gamma u\rangle \overset{w_1}{\Longrightarrow}{}^{+} \langle p', vu\rangle \overset{w_2}{\Longrightarrow}{}^{+} \langle p, \gamma u'u\rangle \overset{w_1}{\Longrightarrow}{}^{+} \langle p', vu'u\rangle \overset{w_2}{\Longrightarrow}{}^{+} \langle p, \gamma u'u'u\rangle \cdots,$$

where only the set F_i of states are visited infinitely often after $\langle p, \gamma u\rangle$. $\square$