1 Introduction

2 Preliminaries

Let $\mathbb{N} = \{1, 2, \ldots\}$, $\mathbb{N}_0 = \{0\} \cup \mathbb{N}$ and $[n] = \{1, \cdots, n\}$ for $n \in \mathbb{N}$. For a set A, let $\mathscr{P}(A)$ be the power set of A, A^* and A^ω be the sets of finite and infinite words over A, and we denote $A^\infty = A^* \cup A^\omega$. For a word $\alpha \in A^\infty$ over a set A, let $\alpha(i) \in A$ be the i-th element of α ($i \geq 0$). Let $\langle u, w\rangle = u(0)w(0)u(1)w(1)\cdots \in A^\infty$ for words $u, w \in A^\infty$ and $\langle B, C\rangle = \{\langle u, w\rangle \mid u \in B, w \in C\}$ for sets $B, C \subseteq A^\infty$. By $|\beta|$, we mean the cardinality of β if β is a set and the length of β if β is a finite sequence.

In this paper, disjoint sets $\Sigma_\mathbb{I}, \Sigma_\mathbb{O}$ and Γ denote a (finite) input alphabet, an output alphabet and a stack alphabet, respectively, and $\Sigma = \Sigma_\mathbb{I} \cup \Sigma_\mathbb{O}$. For a set Γ, let $Com(\Gamma) = \{pop, skip\} \cup \{push(z) \mid z \in \Gamma\}$ be the set of stack commands over Γ. Let $Z : Com(\Gamma) \times \Gamma \to \Gamma^*$ be a function defined as $Z(pop, z) = \varepsilon$, $Z(skip, z) = z$ and $Z(push(z'), z) = z'z$.

3 Pushdown Transducers, Pushdown Automata and Pushdown Games

3.1 Pushdown Transducers

Definition 1. *A pushdown transducer (PDT) over finite alphabets $\Sigma_\mathbb{I}$, $\Sigma_\mathbb{O}$ and Γ is $\mathcal{T} = (P, p_0, z_0, \Delta)$ where P is a finite set of states, $p_0 \in P$ is the initial state, $z_0 \in \Gamma$ is the initial stack symbol and $\Delta : P \times \Sigma_\mathbb{I} \times \Gamma \to P \times \Sigma_\mathbb{O} \times Com(\Gamma)$ is a finite set of deterministic transition rules having one of the following forms:*

- $(p, a, z) \to (q, b, pop)$ *(pop rule)*
- $(p, a, z) \to (q, b, skip)$ *(skip rule)*
- $(p, a, z) \to (q, b, push(z))$ *(push rule)*

where $p, q \in P$, $a \in \Sigma_\mathbb{I}$, $b \in \Sigma_\mathbb{O}$ and $z \in \Gamma$.

For a state $p \in P$ and a stack $w \in \Gamma^*$, (p, w) is called a *configuration* or *instantaneous description (abbreviated as ID)* of PDT $\mathcal{T}$. Let $ID_\mathcal{T}$ denote the set of all IDs of $\mathcal{T}$. For two IDs $(p, w), (q, w') \in ID_\mathcal{T}$ and $(a, b) \in \Sigma_\mathbb{I} \times \Sigma_\mathbb{O}$, we say that (p, w) can transit to (q, w') with an input a and an output b, written as $(p, w) \Rightarrow^{a,b}_\mathcal{T} (q, w')$, if there exist a rule $r = (p, a, z) \to (q, b, com) \in \Delta$ and a sequence $u \in \Gamma^*$ such that $w = zu$ and $w' = Z(com, z)u$. If $\mathcal{T}$ is clear from the context, we abbreviate $\Rightarrow^{a,b}_\mathcal{T}$ as $\Rightarrow^{a,b}$. If a sequence of IDs $(q_0, w_0), \cdots, (q_n, w_n) \in ID_\mathcal{T}$ and $a_1, \cdots, a_n \in \Sigma_\mathbb{I}, b_1, \cdots, b_n \in \Sigma_\mathbb{O}$ satisfy $(q_{i-1}, w_{i-1}) \Rightarrow^{a_i,b_i} (q_i, w_i)$ for all $i \in [n]$, we write $(q_0, w_0) \Rightarrow^{w^\mathbb{I}, w^\mathbb{O}} (q_n, w_n)$ where $w^\mathbb{I} = a_1 \cdots a_n$ and $w^\mathbb{O} = b_1 \cdots b_n$.

By definition, any ID $(p, \varepsilon) \in ID_\mathcal{T}$ has no successor. That is, there is no transition from an ID with empty stack. A *run* of PDT $\mathcal{T}$ is a pair (ρ, w) of an infinite sequence of IDs $\rho \in (ID_\mathcal{T})^\omega$ and $w = a_1 b_1 \cdots \in (\Sigma_\mathbb{I} \cdot \Sigma_\mathbb{O})^\omega$

that satisfy $\rho(i-1) \Rightarrow_{\mathcal{T}}^{a_i, b_i} \rho(i)$ for $i \geq 1$. Let $RUN_{\mathcal{T}}$ denote the set of all runs of $\mathcal{T}$. We define $L(\mathcal{T}) = \{w \in (\Sigma_{\mathbb{I}} \cdot \Sigma_{\mathbb{O}})^{\omega} \mid$ there exists a run (ρ, w) for some $\rho \in (ID_{\mathcal{T}})^{\omega}$ where $\rho(0) = (p_0, z_0)\}$. Let **PDT** be the class of PDT.

3.2 Pushdown Automata

Definition 2. *A pushdown automata (PDA) over finite alphabets* $\Sigma_{\mathbb{I}}$*,* $\Sigma_{\mathbb{O}}$ *and* Γ *is* $\mathcal{A} = (Q, Q_{\mathbb{I}}, Q_{\mathbb{O}}, q_0, z_0, c, \delta)$ *where* Q*,* $Q_{\mathbb{I}}, Q_{\mathbb{O}}$ *are finite sets of states that satisfy* $Q = Q_{\mathbb{I}} \cup Q_{\mathbb{O}}$ *and* $Q_{\mathbb{I}} \cap Q_{\mathbb{O}} = \emptyset$*,* $q_0 \in Q_{\mathbb{I}}$ *is the initial state,* $z_0 \in \Gamma$ *is the initial stack symbol,* $c : Q \to [n]$ *is a coloring function where* $n \in \mathbb{N}$ *is the number of priorities and* $\delta : Q \times \Sigma \times \Gamma \to \mathscr{P}(Q \times Com(\Gamma))$ *is a finite set of transition rules, having one of the following forms:*

- $(q_{\mathbb{I}}, a_{\mathbb{I}}, z) \to (q_{\mathbb{O}}, com)$
- $(q_{\mathbb{O}}, a_{\mathbb{O}}, z) \to (q_{\mathbb{I}}, com)$

where $q_{\mathbb{I}} \in Q_{\mathbb{I}}, q_{\mathbb{O}} \in Q_{\mathbb{O}}, a_{\mathbb{I}} \in \Sigma_{\mathbb{I}}, a_{\mathbb{O}} \in \Sigma_{\mathbb{O}}, z \in \Gamma$ *and* $com \in Com(\Gamma)$*.*

We define an ID, transition relation and a run of $\mathcal{A}$ in a similar way to those of $\mathcal{T}$. More concretely, we assume $ID_{\mathcal{A}} = ID_{\mathcal{T}}$, $(q, u) \vdash_{\mathcal{A}}^{a} (q', u')$ iff there exist a rule $(p, a, z) \to (q, com) \in \delta$ and a sequence $u \in \Gamma^*$ such that $w = zu$ and $w' = Z(com, z)u$.

A *run* of PDA $\mathcal{T}$ is a pair (ρ, w) of an infinite sequence of IDs $\rho \in (ID_{\mathcal{A}})^{\omega}$ and $w = a_1 a_2 \cdots \in (\Sigma_{\mathbb{I}} \cdot \Sigma_{\mathbb{O}})^{\omega}$ that satisfy $\rho(i-1) \Rightarrow_{\mathcal{T}}^{a_i} \rho(i)$ for $i \geq 1$. Let $RUN_{\mathcal{A}}$ denote the set of all runs of $\mathcal{A}$. For $\rho \in (ID_{\mathcal{A}})^{\omega}$, let $state(\rho) \subseteq Q$ be the set of states that occur infinitely in ρ.

We define $L_N(\mathcal{A}) = \{w \in (\Sigma_{\mathbb{I}} \cdot \Sigma_{\mathbb{O}})^{\omega} \mid$ there exists a run $(\rho, w) \in RUN_{\mathcal{A}}$ where $\rho(0) = (p_0, z_0)$ and $\min_{q \in Q}\{c(q) \mid q \in state(\rho)\}$ is even$\}$ and $L_U(\mathcal{A}) = \{w \in (\Sigma_{\mathbb{I}} \cdot \Sigma_{\mathbb{O}})^{\omega} \mid$ every run $(\rho, w) \in RUN_{\mathcal{A}}$ where $\rho(0) = (p_0, z_0)$ satisfies that $\min_{q \in Q}\{c(q) \mid q \in state(\rho)\}$ is even$\}$.

We call a PDA $\mathcal{A} = (P, p_0, z_0, \delta, c)$ deterministic if $|\delta(p, a, z)| \leq 1$ for all $p \in Q, a \in \Sigma$ and $z \in \Gamma$, nondeterministic if the language of $\mathcal{A}$ is defined as $L_N(\mathcal{A})$ and universal if the language of $\mathcal{A}$ is defined as $L_U(\mathcal{A})$. For those cases, we denote $\mathcal{A}$ as DPDA, NPDA and UPDA, respectively. Let **DPDA**, **NPDA** and **UPDA** be the class of DPDA, NPDA and UPDA, respectively.

3.3 Pushdown Games

Definition 3. *A Pushdown Games (PDG) of PDA* $\mathcal{A} = (Q, Q_{\mathbb{I}}, Q_{\mathbb{O}}, q_0, z_0, \delta, c)$ *is* $\mathcal{G}_{\mathcal{A}} = (V, V_{\mathbb{I}}, V_{\mathbb{O}}, E, C)$ *where* $V = Q \times \Gamma^*$ *is the set of vertices with* $V_{\mathbb{I}} = Q_{\mathbb{I}} \times \Gamma^*, V_{\mathbb{O}} = Q_{\mathbb{O}} \times \Gamma^*$*,* $E \subseteq V \times V$ *is the set of edges defined as* $E = \{(v, v') \mid v \vdash^{a} v'$ *for some* $a \in \Sigma\}$ *and* $C : V \to [n]$ *is the coloring function such that* $C((q, u)) = c(q)$ *for all* $(q, u) \in V$*.*

The game starts with some $(q_0, z_0) \in V_{\mathbb{I}}$. When the current vertex is $v \in V_{\mathbb{I}}$, Player I chooses a successor $v' \in V_{\mathbb{O}}$ of v as the next vertice. When the current vertex is $v \in V_{\mathbb{O}}$, Player II chooses a successor $v' \in V_{\mathbb{I}}$ of v. A finite or infinite sequence $\rho \in V^{\infty}$ is valid if $\rho(0) = (q_0, z_0)$ and satisfy $(\rho(i-1), \rho(i)) \in E$ for every $i \geq 1$. A play of $\mathcal{G}_{\mathcal{A}}$ is an infinite and valid sequence $\rho \in V^{\omega}$. A play ρ is winning for Player I iff $\mathit{state}(\rho)$ is even.

By the definition of $\mathcal{G}_{\mathcal{A}}$, every choice of a successor by players can be also expressed as a choice of a pair $(q, \mathit{com}) \in Q \times \mathit{Com}(\Gamma)$. Furthermore, a choice of a successor can be expressed as a choice of $a \in \Sigma$ if $\mathcal{A}$ is deterministic. Thus, every valid sequence $\rho \in V^{\infty}$ corresponds one-to-one with a sequence $\tau \in (Q \times \mathit{Com}(\Gamma))^{\infty}$. In detail, for every $i \geq 0$, $\rho(i) = (q, zu), \tau(i) = (q', \mathit{com})$ and $\rho(i+1) = (q', Z(\mathit{com}, z)u)$ hold for some $q, q' \in Q, z \in \Gamma, u \in \Gamma^*$ and $\mathit{com} \in \mathit{Com}(\Gamma)$. We call τ valid if the corresponding ρ is valid.

Theorem 4. *[Walukiewucz, 2001] If player I has a winning strategy of $\mathcal{G}_{\mathcal{A}}$, we can construct a PDT $\mathcal{T}$ over $Q_{\mathbb{I}} \times \mathit{Com}(\Gamma), Q_{\mathbb{O}} \times \mathit{Com}(\Gamma)$ and an stack alphabet Γ' that gives a winning strategy of $\mathcal{G}_{\mathcal{A}}$. That is, for every $\tau \in L(\mathcal{T})$, the corresponding play $\rho \in V^{\infty}$ is winning for Player I.*

When $\mathcal{A}$ is deterministic, there is also a one-to-one correspondence between a valid sequence $\rho \in V^{\infty}$ and a sequence of input and output alphabets $u \in \Sigma^{\infty}$. In detail, for every $i \geq 0$, $\rho(i) = (q, zu), \rho(i+1) = (q', Z(\mathit{com}, z)u)$ and $(q, u(i), z) \to (q', \mathit{com}) \in \delta$ hold for some $q, q' \in Q, z \in \Gamma, u \in \Gamma^*$ and $\mathit{com} \in \mathit{Com}(\Gamma)$.

By the correspondence, the following lemma holds.

Lemma 5. *A play ρ is winning for Player I iff the corresponding sequence $w \in \Sigma^{\omega}$ of ρ satisfies $w \in L(\mathcal{A})$.*

In a similar way to Theorem 4, we can obtain the following lemma.

Lemma 6. *If $\mathcal{A}$ is deterministic and player I has a winning strategy of $\mathcal{G}_{\mathcal{A}}$, we can construct a PDT $\mathcal{T}$ over $\Sigma_{\mathbb{I}}, \Sigma_{\mathbb{O}}$ and Γ' that gives a winning strategy of $\mathcal{G}_{\mathcal{A}}$. That is, for every $w \in L(\mathcal{T})$, the corresponding play $\rho \in V^{\infty}$ is winning for Player I.*

4 Realizability problems for PDA and PDT

For a specification S and an implementation I, we write $I \models S$ if $L(I) \subseteq L(S)$.

Definition 7. *Realizability problem* Real*$(\mathcal{S}, \mathcal{I})$ for a class of specifications $\mathcal{S}$ and of implementations $\mathcal{I}$: For a specification $S \in \mathcal{S}$, is there an implementation $I \in \mathcal{I}$ such that $I \models S$?*

Theorem 8. Real(**DPDA**, **PDT**) *is decidable.*

Proof. Let $\mathcal{A}$ be a given DPDA. By Lemmas 5 and 6, we can construct a PDT $\mathcal{T}$ such that $\mathcal{T} \models \mathcal{A}$ if player I has a winning strategy for the game $\mathcal{G}_{\mathcal{A}}$. Because there is an algorithm for constructing $\mathcal{T}$ [Walukiewucz, 2001], Real(**DPDA**, **PDT**) is decidable.

Theorem 9. Real(**NPDA**, **PDT**) *is undecidable.*

Proof. For NPDA, we reduce the problem from the universality problem of NPDA, which is undecidable. For a given NPDA $\mathcal{A} = (Q, q_0, z_0, \delta, c)$ over Σ and Γ, we can construct an NPDA $\mathcal{A}' = (Q \cup Q', Q, Q', q_0, z_0, \delta', c')$ over $\Sigma, \Sigma_{\mathbb{O}}$ and Γ that satisfies $L(\mathcal{A}) = \Sigma^\omega$ iff there exists $\mathcal{T}$ such that $\mathcal{T} \models \mathcal{A}$. $\Sigma_{\mathbb{O}}$ is an arbitrary alphabet, $Q' = \{q'_i \mid i \in [n], q_i \in Q\}$ where $Q = \{q_1, \cdots, q_n\}$, $c'(q_i) = c'(q'_i) = c(q_i)$ for all $i \in [n]$ and δ' satisfies that $(q_i, a, z) \to (q_j, \mathit{com}) \in \delta$ iff $(q_i, a, z) \to (q'_j, \mathit{com}) \in \delta'$, and $(q'_j, b, z) \to (q_j, \mathit{skip}) \in \delta'$ for all $b \in \Sigma_{\mathbb{O}}$. By the construction of $\mathcal{A}'$, $L(\mathcal{A}') = \langle L(\mathcal{A}), \Sigma_{\mathbb{O}}^\omega \rangle$ holds. If $L(\mathcal{A}) = \Sigma^\omega$, then $L(\mathcal{A}') = \langle \Sigma^\omega, \Sigma_{\mathbb{O}}^\omega \rangle$ and thus $\mathcal{T} \models \mathcal{A}$ holds for every $\mathcal{T}$. If $L(\mathcal{A}) \neq \Sigma^\omega$, there exists a word $w \in \Sigma^\omega$ such that $w \notin L(\mathcal{A})$. Every language of PDT contains a word $\langle u, v \rangle$ for every $u \in \Sigma^\omega$ and some $v \in \Sigma_{\mathbb{O}}^\omega$, but $\langle w, v \rangle \notin L(\mathcal{A}')$ for any $v \in \Sigma_{\mathbb{O}}^\omega$. Hence, $\mathcal{T} \not\models \mathcal{A}'$ holds for any PDT $\mathcal{T}$. In conclusion, this reduction holds and the realizability problem for PDT and NPDA is undecidable.

5 Register Pushdown Transducers and Register Pushdown Automata

5.1 Data words and registers

We assume a countable set D of *data values*. For finite alphabets $\Sigma_{\mathbb{I}}, \Sigma_{\mathbb{O}}$ and a countable set D, an infinite sequence $(a_1^{\mathbb{I}}, d_1)(a^{\mathbb{O}}, d'_1) \cdots \in ((\Sigma_{\mathbb{I}} \times D) \cdot (\Sigma_{\mathbb{O}} \times D))^\omega$ is called a *data word*. We write $DW(\Sigma_{\mathbb{I}}, \Sigma_{\mathbb{O}}, D) = ((\Sigma_{\mathbb{I}} \times D) \cdot (\Sigma_{\mathbb{O}} \times D))^\omega$.

For $k \in \mathbb{N}_0$, a mapping $\theta : [k] \to D$ is called an *assignment* (of data values to k registers). For $j_1 \cdots j_n \in [k]^*$, we define $\theta(j_1 \cdots j_n) = \theta(j_1) \cdots \theta(j_n)$ and $\theta(\varepsilon) = \varepsilon$. Let Θ_k denote the collection of assignments to k registers. We denote $\mathit{Tst}_k = \mathscr{P}([k] \cup \{\mathit{top}\})$ and $\mathit{Asgn}_k = \mathscr{P}([k])$ where $\mathit{top} \notin \mathbb{N}$ is the unique symbol that represents a stack top value. Tst_k is the set of guard conditions. For $\mathit{tst} \in \mathit{Tst}_k$, $\theta \in \Theta_k$ and $d, e \in D$, we denote $\theta, d, e \models \mathit{tst}$ if $\theta(i) = d \Leftrightarrow i \in \mathit{tst}$ and $e = d \Leftrightarrow \mathit{top} \in \mathit{tst}$ hold. (In definitions of register pushdown transducer (automaton) in the next section, the data values d and e represent an input data value and a stack top data value, respectively.) Asgn_k is the set of assignment conditions. For $\mathit{asgn} \in \mathit{Asgn}_k$, $\theta, \theta' \in \Theta_k$ and $d \in D$, let $\theta[\mathit{asgn} \leftarrow d]$ be the assignment θ' such that $\theta'(i) = d$ for $i \in \mathit{asgn}$ and $\theta'(i) = \theta(i)$ for $i \notin \mathit{asgn}$. Let $Z_D : \mathit{Com}([k]) \times \Theta_k \times D \to D^*$ be a function defined as $Z_D(\mathit{pop}, \theta, d) = \varepsilon$, $Z_D(\mathit{skip}, \theta, d) = d$ and $Z_D(\mathit{push}(j), \theta, d) = \theta(j)d$.

5.2 Register pushdown transducers

Definition 10. *A k-register pushdown transducer (k-RPDT) over finite alphabets $\Sigma_{\mathbb{I}}, \Sigma_{\mathbb{O}}$ and an infinite set D of data values is $\mathcal{T} = (P, p_0, \Delta)$ where P is a finite set of states, $p_0 \in P$ is the initial state, $\Delta : P \times \Sigma_{\mathbb{I}} \times \mathit{Tst}_k \to P \times \Sigma_{\mathbb{O}} \times \mathit{Asgn}_k \times [k] \times \mathit{Com}([k])$ is a finite set of deterministic transition rules.*

D is used as a stack alphabet. For a state $q \in Q$, an assignment $\theta \in \Theta_k$ and a stack $u \in D^*$, (q, θ, u) is called a *configuration* or *instantaneous description (abbreviated as ID)* of k-RPDT $\mathcal{T}$. For two IDs $(p, \theta, u), (q, \theta', u') \in ID_{\mathcal{T}}$ and

$(a, d^{\mathbb{I}}) \in \Sigma_{\mathbb{I}} \times D, (b, d^{\mathbb{O}}) \in \Sigma_{\mathbb{O}} \times D$, we say that (p, θ, u) can transit to (q, θ', u') with an input $(a, d^{\mathbb{I}})$ and an output $(b, d^{\mathbb{O}})$, written as $(p, \theta, u) \Rightarrow_{\mathcal{T}}^{(a,d^{\mathbb{I}}),(b,d^{\mathbb{O}})} (q, \theta', u')$, if there exist a rule $(p, a, tst) \rightarrow (q, b, asgn, j, com) \in \Delta$ a data value $e \in D$, and a sequence of data values $w \in D^*$ where $u = ew$ that satisfy the follows: $d^{\mathbb{I}}, e, \theta \models tst$, $\theta' = \theta[asgn \leftarrow d^{\mathbb{I}}]$, $\theta'(j) = d^{\mathbb{O}}$ and $u' = Z_D(com, \theta', e)w$.

If $\mathcal{T}$ is clear from the context, we abbreviate $\Rightarrow_{\mathcal{T}}^{(a,d^{\mathbb{I}}),(b,d^{\mathbb{O}})}$ as $\Rightarrow^{(a,d^{\mathbb{I}}),(b,d^{\mathbb{O}})}$.

A *run* of k-RPDT $\mathcal{T}$ is a pair (ρ, u) of an infinite sequence of IDs $\rho \in (ID_{\mathcal{T}})^{\omega}$ and data word $u = (a_1^{\mathbb{I}}, d_1^{\mathbb{I}})(a_1^{\mathbb{O}}, d_1^{\mathbb{O}}) \cdots \in DW(\Sigma_{\mathbb{I}}, \Sigma_{\mathbb{O}}, D)$ that satisfy $\rho(i-1) \Rightarrow_{\mathcal{T}}^{(a_i^{\mathbb{I}},d_i^{\mathbb{I}}),(a_i^{\mathbb{O}},d_i^{\mathbb{O}})} \rho(i)$ for $i \geq 1$. We define $L(\mathcal{T}) = \{u \in DW(\Sigma_{\mathbb{I}}, \Sigma_{\mathbb{O}}, D) \mid$ there exists a run (ρ, u) for some $\rho \in (ID_{\mathcal{T}})^{\omega}$ which begins with the initial state $p_0\}$. Let $\mathbf{RPDT}[k]$ be the class of k-RPDT and $\mathbf{RPDT} = \bigcup_{k \in \mathbb{N}_0} \mathbf{RPDT}[k]$. Note that $\mathbf{RPDT}[0] = \mathbf{PDT}$.

5.3 Register pushdown automata

Definition 11. *A k-register pushdown automaton (k-RPDA) over $\Sigma_{\mathbb{I}}, \Sigma_{\mathbb{O}}$ and D is $\mathcal{A} = (Q, Q_{\mathbb{I}}, Q_{\mathbb{O}}, q_0, \delta, c)$, where*

- *Q is a finite set of states,*
- *$Q_{\mathbb{I}} \cup Q_{\mathbb{O}} = Q, Q_{\mathbb{I}} \cap Q_{\mathbb{O}} = \emptyset$,*
- *$q_0 \in Q$ is the initial state, and*
- *$\delta : Q \times (\Sigma \cup \{\varepsilon\}) \times Tst_k \rightarrow \mathscr{P}(Q \times Asgn_k \times Com([k]))$ is a transition function having one of the forms:*
  - *$(q_{\mathbb{I}}, a_{\mathbb{I}}, tst) \rightarrow (q_{\mathbb{O}}, asgn, com)$ (input rule)*
  - *$(q_{\mathbb{O}}, a_{\mathbb{O}}, tst) \rightarrow (q_{\mathbb{I}}, asgn, com)$ (output rule)*
  - *$(q, \varepsilon, tst) \rightarrow (q', asgn, com)$ (ε rule)*

  where $q_{\mathbb{I}} \in Q_{\mathbb{I}}$, $q_{\mathbb{O}} \in Q_{\mathbb{O}}$, $q, q' \in Q$, $a_{\mathbb{I}} \in \Sigma_{\mathbb{I}}$, $a_{\mathbb{O}} \in \Sigma_{\mathbb{O}}$, $tst \in Tst_k$, $j \in [k]$ and $com \in Com([k])$.
- *$c : Q \rightarrow [n]$ where $n \in \mathbb{N}$ is the number of priorities.*

We define the set of IDs of $\mathcal{A}$ as that of $\mathcal{T}$, i.e., $ID_{\mathcal{A}} = ID_{\mathcal{T}}$. By a rule $(q, a, tst) \rightarrow (q', asgn, com) \in \delta$ and $(a, d) \in (\Sigma \cup \{\varepsilon\}) \times D$, an ID $(q, \theta, ew) \in ID_{\mathcal{A}}$ (where $e \in D, w \in D^*$) can transit to $(q', \theta[asgn \leftarrow d], \theta'(J)w)$, written as $(q, \theta, ew) \vdash_{\mathcal{A}}^{(a,d)} (q', \theta[asgn \leftarrow d], Z_D(com, \theta', e)w)$ where $\vdash$ is the transition relation of $\mathcal{A}$. We write $\vdash_{\mathcal{A}}^{(a,d)}$ as $\vdash^{(a,d)}$ if $\mathcal{A}$ is clear from comtext. We also define a run $(\rho, u) \in (ID_{\mathcal{T}})^{\omega} \times ((\Sigma \cup \{\varepsilon\}) \times D)^{\omega}$ of $\mathcal{A}$ as satisfying $\rho(i-1) \vdash_{\mathcal{A}}^{(a_i,d_i)} \rho(i)$ for $i \geq 1$ where $u = (a_1, d_1)(a_2, d_2) \cdots \in ((\Sigma \cup \{\varepsilon\}) \times D)^{\omega}$. For $\rho \in (ID_{\mathcal{T}})^{\omega}$, let $c(\rho) = \max\{j \mid$ there exists $q \in Q$ which appears in ρ infinitely and satiefies $c(q) = j\}$. For $w \in ((\Sigma \cup \{\varepsilon\}) \times D)^{\omega}$, let $ef(w) \in (\Sigma \times D)^{\omega}$ be an epsilon free sequence of w. That is, $ef(w)(i) = w(i)$ if $w(i) \in \Sigma \times D$ and $ef(w)(i) = \varepsilon$ if $w(i) \in \{\varepsilon \times D$ for $i \geq 0$.

We define $L_N(\mathcal{A}) = \{ef(w) \in DW(\Sigma_{\mathbb{I}}, \Sigma_{\mathbb{O}}, D) \mid$ there exists a run $(\rho, w') \in RUN_{\mathcal{A}}$ where $\rho(0) = (p_0, z_0)$ and $\min_{q \in Q}\{c(q) \mid q \in state(\rho)\}$ is even$\}$ and $L_U(\mathcal{A}) = \{ef(w) \in (\Sigma_{\mathbb{I}} \cdot \Sigma_{\mathbb{O}})^{\omega} \mid$ every run $(\rho, w) \in RUN_{\mathcal{A}}$ where $\rho(0) = (p_0, z_0)$ satisfies that $\min_{q \in Q}\{c(q) \mid q \in state(\rho)\}$ is even$\}$.

5.4 Classes of RPDA

k-RPDA $\mathcal{A} = (Q, q_0, \delta, c)$ is said to be deterministic if $|\delta(q, a, tst) \cup \delta(q, \varepsilon, tst)| \leq 1$ for all $q \in Q, a \in \Sigma, tst \in Tst_k$, nondeterministic if the language of $\mathcal{A}$ is defined as $L_N(\mathcal{A})$ and universal if the language of $\mathcal{A}$ is defined as $L_U(\mathcal{A})$. For those cases, we denote $\mathcal{A}$ as k-DRPDA, k-NRPDA and k-URPDA, respectively. An ε-free k-RPDA is an RPDA not having any ε rules. Let **DRPDA**, **NRPDA** and **URPDA** be the class of ε-free k-DRPDA, k-NRPDA and k-URPDA for all $k \in \mathbb{N}_0$, respectively.

Let $Com_v = \{pop, skip, push\}$ and $v : Com([k]) \to Com$ be a function such that $v(push(j)) = push$ for $j \in [k]$ and $v(com) = com$ otherwise. An visible k-RPDA is input and output alphabets are $\Sigma_{\mathbb{I}} \times Com_v$ and $\Sigma_{\mathbb{O}} \times Com_v$, respectively, and every rule has one of the form $(q, (a, v(com))) \to (q', asgn, com)$. Let **DRPDAv** be the class of visible ε-free k-DRPDA for all $k \in \mathbb{N}_0$, respectively.

6 Realizability problems for RPDA and RPDT

6.1 Finite actions

For $k \in \mathbb{N}_0$, we define the set of visible finite input actions as $A_k^{\mathbb{I}} = \Sigma_{\mathbb{I}} \times \{skip\} \times Tst_k$ and the set of visible output actions as $A_k^{\mathbb{O}} = \{(\sigma_o, v(com), asgn, j, com) \in \Sigma_{\mathbb{O}} \times Com_v \times Asgn_k \times [k] \times Com([k])\}$ for k-RPDT. A sequence $w = ((a_1^{\mathbb{I}}, skip), d_1^{\mathbb{I}})((a_1^{\mathbb{O}}, v(com_1)), d_1^{\mathbb{O}}) \cdots \in DW(\Sigma_{\mathbb{I}}, \Sigma_{\mathbb{O}}, D)$ is compatible with a sequence $\overline{a} = (a_1^{\mathbb{I}}, skip, tst_1)(a_1^{\mathbb{O}}, v(com_1), asgn_1, j_1, com_1) \cdots \in (A_k^{\mathbb{I}} A_k^{\mathbb{O}})^\omega$ if there exists a run (ρ, w) of k-RPDT satisfying follows: For all $i \geq 1$, let $\rho(i-1) = (q, \theta, eu)$ and $\rho(i) = (q', \theta', u'u)$ for some $e \in D, u \in D^*$ and $u' \in D^*$. Then $\theta, d_i^{\mathbb{I}}, e \models tst_i, \theta' = \theta[asgn_i \leftarrow d_i^{\mathbb{I}}], \theta'(j_1) = d_i^{\mathbb{O}}$ and $u' = Z_D(com, \theta', e)$ hold. Let $Comp(\overline{a}) = \{w \in DW(\Sigma_{\mathbb{I}}, \Sigma_{\mathbb{O}}, D) \mid w \text{ is compatible with } \overline{a}\ \}$. For specification $S \subseteq DW(\Sigma_{\mathbb{I}}, \Sigma_{\mathbb{O}}, D)$, we define $W_{S,k} = \{\overline{a} \mid Comp(\overline{a}) \subseteq S\}$.

Theorem 12. *For a specification $S \subseteq DW(\Sigma_{\mathbb{I}}, \Sigma_{\mathbb{O}}, D)$, the following statements are equivalent.*

- *There exists a k-RPDT $\mathcal{T}$ such that $L(\mathcal{T}) \subseteq S$.*
- *There exists a PDT $\mathcal{T}'$ such that $L(\mathcal{T}') \subseteq W_{S,k}$.*

6.2 Decidability and undecidability of realizability problems

Lemma 13. *$L_k = \{w \otimes \overline{a} \mid w \in Comp(\overline{a})\}$ is definable as a language of $(k+2)$-DRPDA.*

Proof. Let $(k+2)$-DRPDA $\mathcal{A}_k = (\{p, q\} \cup (Asgn_k \times [k] \times Com([k])) \cup [k], \{p\}, \{q\} \cup (Asgn_k \times [k] \times Com([k])) \cup [k], p, \delta_k, c_k)$ over $A_k^{\mathbb{I}}, A_k^{\mathbb{O}}$ and D where $c_k(s) = 2$ for all state s and δ_k consists of rules of the form

$$(p, (a_{\mathbb{I}}, skip, tst), tst) \to (q, \{k+1\}, skip) \tag{1}$$

$$(q, (a_{\mathbb{O}}, v(com), asgn, j, com), tst') \to ((asgn, j, com), \{k+2\}, skip) \tag{2}$$

$$((asgn, j, com), \varepsilon, \{k+1\} \cup tst') \to (j, asgn, com) \tag{3}$$

$$(j, \varepsilon, \{j, k+2\} \cup tst') \to (p, \emptyset, skip) \tag{4}$$

for all $(a_{\mathbb{I}}, tst) \in A_k^{\mathbb{I}}$, $(a_{\mathbb{O}}, asgn, j, com) \in A_k^{\mathbb{O}}$ and $tst' \in Tst_{k+2}$. Then, $L(\mathcal{A}_k) = L_k$ holds.

Lemma 14. *For specification $\mathcal{S}$ definable by some visible ε-free k'-DRPDA. $L_{k,\overline{S}} = \{w \otimes \overline{a} \mid w \in Comp(\overline{a}) \cap \overline{S}\}$ is definable as a language of visible $(k+k'+4)$-DRPDA.*

Proof. Let $L_{\overline{S}} = \{w \otimes \overline{a} \mid w \in \overline{S}\}$, $\mathcal{A}_{\overline{S}}$ be a visible ε-free k'-DRPDA such that $L(\mathcal{A}_{\overline{S}}) = L_{\overline{S}}$ and $\mathcal{A}_k$ be a $(k+2)$-DRPDA defined in Lemma 13. Because $L_{k,\overline{S}} = L_k \cap L_{\overline{S}}$ and both L_k and $L_{\overline{S}}$ are visible DRPDA, it is enough to show we can construct visible $(k+k'+4)$-DRPDA $\mathcal{A}$ such that $L(\mathcal{A}) = L(\mathcal{A}_{\overline{S}}) \cap L(\mathcal{A}_k)$.

For simplicity, we rewrite $\mathcal{A}_k$ as k_1-DRPDA $\mathcal{A}_1 = (Q_1, Q_1^{\mathbb{I}}, Q_1^{\mathbb{O}}, q_1^0, \delta_1, c_1)$ and $\mathcal{A}_{\overline{S}}$ as k_2-DRPDA $\mathcal{A}_2 = (Q_2, Q_2^{\mathbb{I}}, Q_2^{\mathbb{O}}, q_2^0, \delta_2, c_2)$, but they satisfy that $c_1(q)$ is even for all $q \in Q_1$ and every rules in δ_1 forms triple sequencial rules

$$(q_1, (a, v(com_1)), tst_1) \to (q_2, asgn_1, skip) \tag{2'}$$

$$(q_2, \varepsilon, tst_2) \to (q_3, asgn_2, com_1) \tag{3'}$$

$$(q_3, \varepsilon, tst_3) \to (q_4, asgn_3, skip) \quad . \tag{4'}$$

Note that $(2'), (3')$ and $(4')$ correspond to $(2), (3)$ and (4), respectively, and (1) can be devided in three rules of the form $(2'), (3')$ and $(4')$.

We construct $(k_1 + k_2 + 2)$-DRPDA $\mathcal{A} = (Q_1 \times Q_2 \times [5], Q_1^{\mathbb{I}} \times Q_2^{\mathbb{I}} \times [5], Q_1^{\mathbb{O}} \times Q_2^{\mathbb{O}} \times [5], (q_0^1, q_0^2, 1), \delta, c)$ where $c((q_1, q_2, i)) = c_2(q_2)$ for all $(q_1, q_2, i) \in Q$. For all rules

- $(q_1, (a, v(com_1)), tst_1) \to (q_2, asgn_1, skip)$,
- $(q_2, \varepsilon, tst_2) \to (q_3, asgn_2, com_1)$,
- $(q_3, \varepsilon, tst_3) \to (q_4, asgn_3, skip) \in \delta_1$ and
- $(q, (a, v(com)), tst) \to (q', asgn, com) \in \delta_2$

$(v(com_1) = v(com))$ for $a \in \Sigma$, let $tst^{+k_1} = \{i + k_1 \mid i \in tst\} \cup \{top \mid top \in tst \setminus [k_1]\}$, $asgn^{+k_1} = \{i + k_1 \mid i \in asgn\}$ and $com^{+k_1} = push(j + k_1)$ if $com = push(j)$ and $com^{+k_1} = com$ otherwise, then δ consists of the rules

- $((q_1, q, 1), \varepsilon, tst' \cup \{top\}) \to ((q_1, q, 2), \{k_1 + k_2 + 1\}, pop)$
- $((q_1, q, 2), \varepsilon, tst' \cup \{top\}) \to ((q_1, q, 3), \{k_1 + k_2 + 2\}, push(k_1 + k_2 + 1))$
- $((q_1, q, 3), (a, v(com_1)), (tst_1 \cup tst^{+k_1}) \setminus top \cup \{k_1 + k_2 + t \mid t = 1$ if $top \in tst_1$ and $t = 2$ if $top \in tst$ $\}) \to ((q_2, q', 4), asgn_1 \cup asgn^{+k_1}, com^{+k_1})$
- $((q_2, q', 4), \varepsilon, tst_2 \cup tst') \to ((q_3, q', 5), asgn_2, com_1)$
- $((q_3, q', 5), \varepsilon, tst_3 \cup tst') \to ((q_4, q', 0), asgn_3, skip)$

for all $tst' \in Tst_{k_1+k_2+2}$. Then, $L(\mathcal{A}) = L(\mathcal{A}_1) \cap L(\mathcal{A}_2)$.

Lemma 15. $W_{S,k} = \overline{Lab(L_{\overline{S},k})}$.

Proof. For every $\overline{a} \in (A_k^{\mathbb{I}} A_k^{\mathbb{O}})^\omega$, $\overline{a} \notin W_{S,k} \Leftrightarrow Comp(\overline{a}) \not\subseteq S \Leftrightarrow \exists w.w \in Comp(\overline{a}) \cap \overline{S} \Leftrightarrow \exists w.w \otimes \overline{a} \in L_{\overline{S},k} \Leftrightarrow \overline{a} \in Lab(L_{\overline{S},k})$. Thus, $W_{S,k} = \overline{Lab(L_{\overline{S},k})}$ holds.

Theorem 16. *For all* $k \geq 0$*,* Real(**DRPDAv**, **RPDT**$[k]$) *is decidable.*

Proof. By Lemma 14, $L_{\overline{S},k}$ is definable by some visible DRPDA. Because every language recognized by some visible DRPDA can be converted to the language of visible DPDA by taking a projection on its label, $W_{S,k}$ is definable by some visible DPDA by Lemma 15. By Theorem 12, we can check Real(**DPDA**, **PDT**) for $W_{S,k}$, which is shown to be decidable in Theorem 8, instead of checking Real(**DRPDAv**, **RPDT**$[k]$).

Theorem 17. *For all* $k \geq 0$*,* Real(**NRPDA**, **RPDT**$[k]$) *is undecidable.*

Proof. We can easily reduce the problem from Real(**NPDA**, **PDT**), whose undecidability is proved in Theorem 9.

7 Conclusion

References