Descriptional Complexity of Pushdown Store Languages

Andreas Malcher   Katja Meckel
Carlo Mereghetti   Beatrice Palano

Institut für Informatik, Universität Giessen, Germany
Dipartimento di Informatica, Università degli Studi di Milano
Milano, Italy

ICTCS 2012, Varese, Italy

Descriptional complexity: questions

Take the length of description as complexity measure.

- How succinctly can a model represent a formal language in comparison with other models?
- What is the maximum blow-up when changing from one model to another? (Upper bounds)
- Are there languages such that a maximum blow-up is achieved? (Lower bounds)

Descriptional complexity: questions

Take the length of description as complexity measure.

- How succinctly can a model represent a formal language in comparison with other models?
- What is the maximum blow-up when changing from one model to another? (Upper bounds)
- Are there languages such that a maximum blow-up is achieved? (Lower bounds)

Results

- Recursive trade-offs
- Non-recursive trade-offs

Devices and related structures

Not only devices themselves are of interest, but also structures related to them.

Devices and related structures

Not only devices themselves are of interest, but also structures related to them.

Examples:

- Turing machines and the set of valid computations (Hartmanis 1967)

Devices and related structures

Not only devices themselves are of interest, but also structures related to them.

Examples:

- Turing machines and the set of valid computations (Hartmanis 1967)
- Quantum finite automata and control languages (Mereghetti, Palano 2006)

Devices and related structures

Not only devices themselves are of interest, but also structures related to them.

Examples:

- Turing machines and the set of valid computations (Hartmanis 1967)
- Quantum finite automata and control languages (Mereghetti, Palano 2006)
- Pushdown automata with context-dependent nondeterminism (Kutrib, Malcher 2006)

Devices and related structures

Not only devices themselves are of interest, but also structures related to them.

Examples:

- Turing machines and the set of valid computations (Hartmanis 1967)
- Quantum finite automata and control languages (Mereghetti, Palano 2006)
- Pushdown automata with context-dependent nondeterminism (Kutrib, Malcher 2006)
- Grammars and regulated rewriting (Dassow, Păun 1989)

Devices and related structures

Not only devices themselves are of interest, but also structures related to them.

Examples:

- Turing machines and the set of valid computations (Hartmanis 1967)
- Quantum finite automata and control languages (Mereghetti, Palano 2006)
- Pushdown automata with context-dependent nondeterminism (Kutrib, Malcher 2006)
- Grammars and regulated rewriting (Dassow, Păun 1989)
- ...

Devices and related structures

Not only devices themselves are of interest, but also structures related to them.

Examples:

- Turing machines and the set of valid computations (Hartmanis 1967)
- Quantum finite automata and control languages (Mereghetti, Palano 2006)
- Pushdown automata with context-dependent nondeterminism (Kutrib, Malcher 2006)
- Grammars and regulated rewriting (Dassow, Păun 1989)
- ...
- Finite automata and the size of their syntactic monoid (Holzer, König 2002)

Pushdown store languages

The pushdown store language of a PDA M is the set $P(M)$ of all words occurring on the pushdown store along accepting computations of M.

$$P(M) = \{u \in \Gamma^* \mid \exists x, y \in \Sigma^*,\ q \in Q,\ f \in F :$$
$$(q_0, xy, Z_0) \vdash^* (q, y, u) \vdash^* (f, \lambda, \gamma),\ \text{for some } \gamma \in \Gamma^*\}.$$

Pushdown store languages

The pushdown store language of a PDA M is the set $P(M)$ of all words occurring on the pushdown store along accepting computations of M.

$$P(M) = \{u \in \Gamma^* \mid \exists x, y \in \Sigma^*,\ q \in Q,\ f \in F : \\ (q_0, xy, Z_0) \vdash^* (q, y, u) \vdash^* (f, \lambda, \gamma),\ \text{for some } \gamma \in \Gamma^*\}.$$

Theorem (Greibach 1967)

Let M be a PDA. Then, $P(M)$ is a regular language.

Example

The language $\{\, a^n b^n \mid n \geq 1 \,\}$ is accepted by the following (deterministic) PDA

$$M = \langle \{q_0, q_1, q_2\}, \{a, b\}, \{Z, Z_0\}, \delta, q_0, Z_0, \{q_2\} \rangle$$

such that

$$\begin{aligned} \delta(q_0, a, Z_0) = \{(q_0, ZZ_0)\}, \;\; & \delta(q_0, a, Z) = \{(q_0, ZZ)\}, \\ \delta(q_0, b, Z) = \{(q_1, \lambda)\}, & \\ \delta(q_1, b, Z) = \{(q_1, \lambda)\}, \;\; & \delta(q_1, \lambda, Z_0) = \{(q_2, Z_0)\}. \end{aligned}$$

Example

The language $\{ a^n b^n \mid n \geq 1 \}$ is accepted by the following (deterministic) PDA

$$M = \langle \{q_0, q_1, q_2\}, \{a, b\}, \{Z, Z_0\}, \delta, q_0, Z_0, \{q_2\} \rangle$$

such that

$$\begin{aligned} \delta(q_0, a, Z_0) = \{(q_0, ZZ_0)\}, \;\; & \delta(q_0, a, Z) = \{(q_0, ZZ)\}, \\ \delta(q_0, b, Z) = \{(q_1, \lambda)\}, & \\ \delta(q_1, b, Z) = \{(q_1, \lambda)\}, \;\; & \delta(q_1, \lambda, Z_0) = \{(q_2, Z_0)\}. \end{aligned}$$

The pushdown store language is $P(M) = Z^* Z_0$.

Finite automata construction

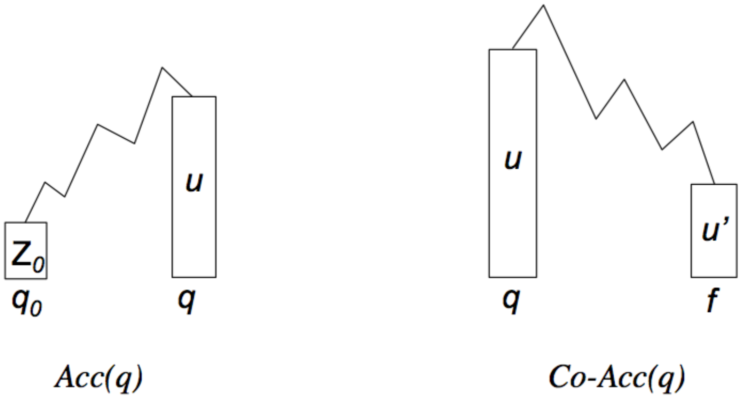
Autebert, Berstel, and Boasson (1997) propose the following construction:

Finite automata construction

Autebert, Berstel, and Boasson (1997) propose the following construction:

Let $M = \langle Q, \Sigma, \Gamma, \delta, q_0, Z_0, F \rangle$ be a PDA. For every $q \in Q$,

$$\mathit{Acc}(q) = \{u \in \Gamma^* \mid \exists x, y \in \Sigma^* : (q_0, xy, Z_0) \vdash^* (q, y, u)\},$$
$$\mathit{Co\text{-}Acc}(q) = \{u \in \Gamma^* \mid \exists y \in \Sigma^*, f \in F, u' \in \Gamma^* : (q, y, u) \vdash^* (f, \lambda, u')\}.$$



Finite automata construction

Then, the pushdown store language is

$$P(M) = \bigcup_{q \in Q} \mathit{Acc}(q) \cap \mathit{Co\text{-}Acc}(q).$$

Finite automata construction

Then, the pushdown store language is

$$P(M) = \bigcup_{q \in Q} \mathit{Acc}(q) \cap \mathit{Co\text{-}Acc}(q).$$

Finally, for every $q \in Q$, a left-linear grammar $G_{\mathit{Acc}(q)}$ for $\mathit{Acc}(q)$ and a right-linear grammar $G_{\mathit{Co\text{-}Acc}(q)}$ for $\mathit{Co\text{-}Acc}(q)$ is constructed.

Finite automata construction

Then, the pushdown store language is

$$P(M) = \bigcup_{q \in Q} \mathit{Acc}(q) \cap \mathit{Co\text{-}Acc}(q).$$

Finally, for every $q \in Q$, a left-linear grammar $G_{\mathit{Acc}(q)}$ for $\mathit{Acc}(q)$ and a right-linear grammar $G_{\mathit{Co\text{-}Acc}(q)}$ for $\mathit{Co\text{-}Acc}(q)$ is constructed.

Estimation of the size:

Finite automata construction

Then, the pushdown store language is

$$P(M) = \bigcup_{q \in Q} \mathit{Acc}(q) \cap \mathit{Co\text{-}Acc}(q).$$

Finally, for every $q \in Q$, a left-linear grammar $G_{\mathit{Acc}(q)}$ for $\mathit{Acc}(q)$ and a right-linear grammar $G_{\mathit{Co\text{-}Acc}(q)}$ for $\mathit{Co\text{-}Acc}(q)$ is constructed.

Estimation of the size:

- An NFA for $\mathit{Acc}(q)$ needs $|Q| \cdot |\Gamma| + 1$ states.

Finite automata construction

Then, the pushdown store language is

$$P(M) = \bigcup_{q \in Q} \mathit{Acc}(q) \cap \mathit{Co\text{-}Acc}(q).$$

Finally, for every $q \in Q$, a left-linear grammar $G_{\mathit{Acc}(q)}$ for $\mathit{Acc}(q)$ and a right-linear grammar $G_{\mathit{Co\text{-}Acc}(q)}$ for $\mathit{Co\text{-}Acc}(q)$ is constructed.

Estimation of the size:

- An NFA for $\mathit{Acc}(q)$ needs $|Q| \cdot |\Gamma| + 1$ states.
- An NFA for $\mathit{Co\text{-}Acc}(q)$ needs $|Q| + 1$ states.

Finite automata construction

Then, the pushdown store language is

$$P(M) = \bigcup_{q \in Q} \mathit{Acc}(q) \cap \mathit{Co\text{-}Acc}(q).$$

Finally, for every $q \in Q$, a left-linear grammar $G_{\mathit{Acc}(q)}$ for $\mathit{Acc}(q)$ and a right-linear grammar $G_{\mathit{Co\text{-}Acc}(q)}$ for $\mathit{Co\text{-}Acc}(q)$ is constructed.

Estimation of the size:

- ➜ An NFA for $\mathit{Acc}(q)$ needs $|Q| \cdot |\Gamma| + 1$ states.
- ➜ An NFA for $\mathit{Co\text{-}Acc}(q)$ needs $|Q| + 1$ states.
- ➜ An NFA for the intersection $\mathit{Acc}(q) \cap \mathit{Co\text{-}Acc}(q)$ needs $(|Q| \cdot |\Gamma| + 1)(|Q| + 1)$ states.

Finite automata construction

Then, the pushdown store language is

$$P(M) = \bigcup_{q \in Q} \mathit{Acc}(q) \cap \mathit{Co\text{-}Acc}(q).$$

Finally, for every $q \in Q$, a left-linear grammar $G_{\mathit{Acc}(q)}$ for $\mathit{Acc}(q)$ and a right-linear grammar $G_{\mathit{Co\text{-}Acc}(q)}$ for $\mathit{Co\text{-}Acc}(q)$ is constructed.

Estimation of the size:

- ➜ An NFA for $\mathit{Acc}(q)$ needs $|Q| \cdot |\Gamma| + 1$ states.
- ➜ An NFA for $\mathit{Co\text{-}Acc}(q)$ needs $|Q| + 1$ states.
- ➜ An NFA for the intersection $\mathit{Acc}(q) \cap \mathit{Co\text{-}Acc}(q)$ needs $(|Q| \cdot |\Gamma| + 1)(|Q| + 1)$ states.
- ➜ The union over all $q \in Q$ gives a factor $|Q|$.

Finite automata construction

Then, the pushdown store language is

$$P(M) = \bigcup_{q \in Q} \mathit{Acc}(q) \cap \mathit{Co\text{-}Acc}(q).$$

Finally, for every $q \in Q$, a left-linear grammar $G_{\mathit{Acc}(q)}$ for $\mathit{Acc}(q)$ and a right-linear grammar $G_{\mathit{Co\text{-}Acc}(q)}$ for $\mathit{Co\text{-}Acc}(q)$ is constructed.

Estimation of the size:

- An NFA for $\mathit{Acc}(q)$ needs $|Q| \cdot |\Gamma| + 1$ states.
- An NFA for $\mathit{Co\text{-}Acc}(q)$ needs $|Q| + 1$ states.
- An NFA for the intersection $\mathit{Acc}(q) \cap \mathit{Co\text{-}Acc}(q)$ needs $(|Q| \cdot |\Gamma| + 1)(|Q| + 1)$ states.
- The union over all $q \in Q$ gives a factor $|Q|$.
- Altogether, we need $|Q|^3|\Gamma| + |Q|^2(|\Gamma| + 1) + |Q| + 1$ states.

Finite automata construction improved

Avoid the union (factor $|Q|$) by considering

$$\mathit{Acc}(Q) = \{[q]u \in [Q]\Gamma^* \mid u \in \mathit{Acc}(q)\},$$
$$\mathit{Co\text{-}Acc}(Q) = \{[q]u \in [Q]\Gamma^* \mid u \in \mathit{Co\text{-}Acc}(q)\}.$$

Finite automata construction improved

Avoid the union (factor $|Q|$) by considering

$$\mathit{Acc}(Q) = \{[q]u \in [Q]\Gamma^* \mid u \in \mathit{Acc}(q)\},$$
$$\mathit{Co\text{-}Acc}(Q) = \{[q]u \in [Q]\Gamma^* \mid u \in \mathit{Co\text{-}Acc}(q)\}.$$

Then, $P(M)$ is $\mathit{Acc}(Q) \cap \mathit{Co\text{-}Acc}(Q)$ where the first symbol is removed.

Finite automata construction improved

Avoid the union (factor $|Q|$) by considering

$$\mathit{Acc}(Q) = \{[q]u \in [Q]\Gamma^* \mid u \in \mathit{Acc}(q)\},$$
$$\mathit{Co\text{-}Acc}(Q) = \{[q]u \in [Q]\Gamma^* \mid u \in \mathit{Co\text{-}Acc}(q)\}.$$

Then, $P(M)$ is $\mathit{Acc}(Q) \cap \mathit{Co\text{-}Acc}(Q)$ where the first symbol is removed.

Estimation of the size:

Finite automata construction improved

Avoid the union (factor $|Q|$) by considering

$$\mathit{Acc}(Q) = \{[q]u \in [Q]\Gamma^* \mid u \in \mathit{Acc}(q)\},$$
$$\mathit{Co\text{-}Acc}(Q) = \{[q]u \in [Q]\Gamma^* \mid u \in \mathit{Co\text{-}Acc}(q)\}.$$

Then, $P(M)$ is $\mathit{Acc}(Q) \cap \mathit{Co\text{-}Acc}(Q)$ where the first symbol is removed.

Estimation of the size:

- An NFA for $\mathit{Acc}(Q)$ needs $|Q|(|\Gamma| + 1) + 1$ states.

Finite automata construction improved

Avoid the union (factor $|Q|$) by considering

$$\mathit{Acc}(Q) = \{[q]u \in [Q]\Gamma^* \mid u \in \mathit{Acc}(q)\},$$
$$\mathit{Co\text{-}Acc}(Q) = \{[q]u \in [Q]\Gamma^* \mid u \in \mathit{Co\text{-}Acc}(q)\}.$$

Then, $P(M)$ is $\mathit{Acc}(Q) \cap \mathit{Co\text{-}Acc}(Q)$ where the first symbol is removed.

Estimation of the size:

- An NFA for $\mathit{Acc}(Q)$ needs $|Q|(|\Gamma| + 1) + 1$ states.
- An NFA for $\mathit{Co\text{-}Acc}(Q)$ needs $|Q| + 2$ states.

Finite automata construction improved

Avoid the union (factor $|Q|$) by considering

$$\mathit{Acc}(Q) = \{[q]u \in [Q]\Gamma^* \mid u \in \mathit{Acc}(q)\},$$
$$\mathit{Co\text{-}Acc}(Q) = \{[q]u \in [Q]\Gamma^* \mid u \in \mathit{Co\text{-}Acc}(q)\}.$$

Then, $P(M)$ is $\mathit{Acc}(Q) \cap \mathit{Co\text{-}Acc}(Q)$ where the first symbol is removed.

Estimation of the size:

- An NFA for $\mathit{Acc}(Q)$ needs $|Q|(|\Gamma| + 1) + 1$ states.
- An NFA for $\mathit{Co\text{-}Acc}(Q)$ needs $|Q| + 2$ states.
- An NFA for the intersection $\mathit{Acc}(Q) \cap \mathit{Co\text{-}Acc}(Q)$ needs $(|Q|(|\Gamma| + 1) + 1)(|Q| + 2)$ states.

Finite automata construction improved

Avoid the union (factor $|Q|$) by considering

$$\mathit{Acc}(Q) = \{[q]u \in [Q]\Gamma^* \mid u \in \mathit{Acc}(q)\},$$
$$\mathit{Co\text{-}Acc}(Q) = \{[q]u \in [Q]\Gamma^* \mid u \in \mathit{Co\text{-}Acc}(q)\}.$$

Then, $P(M)$ is $\mathit{Acc}(Q) \cap \mathit{Co\text{-}Acc}(Q)$ where the first symbol is removed.

Estimation of the size:

- An NFA for $\mathit{Acc}(Q)$ needs $|Q|(|\Gamma| + 1) + 1$ states.
- An NFA for $\mathit{Co\text{-}Acc}(Q)$ needs $|Q| + 2$ states.
- An NFA for the intersection $\mathit{Acc}(Q) \cap \mathit{Co\text{-}Acc}(Q)$ needs $(|Q|(|\Gamma| + 1) + 1)(|Q| + 2)$ states.
- The removal of the first symbol is for free.

Finite automata construction improved

Avoid the union (factor $|Q|$) by considering

$$\mathit{Acc}(Q) = \{[q]u \in [Q]\Gamma^* \mid u \in \mathit{Acc}(q)\},$$
$$\mathit{Co\text{-}Acc}(Q) = \{[q]u \in [Q]\Gamma^* \mid u \in \mathit{Co\text{-}Acc}(q)\}.$$

Then, $P(M)$ is $\mathit{Acc}(Q) \cap \mathit{Co\text{-}Acc}(Q)$ where the first symbol is removed.

Estimation of the size:

- An NFA for $\mathit{Acc}(Q)$ needs $|Q|(|\Gamma| + 1) + 1$ states.
- An NFA for $\mathit{Co\text{-}Acc}(Q)$ needs $|Q| + 2$ states.
- An NFA for the intersection $\mathit{Acc}(Q) \cap \mathit{Co\text{-}Acc}(Q)$ needs $(|Q|(|\Gamma| + 1) + 1)(|Q| + 2)$ states.
- The removal of the first symbol is for free.
- Altogether, we need $|Q|^2(|\Gamma| + 1) + |Q|(2|\Gamma| + 3) + 2$ states.

Lower bounds

Consider the language family $L_{m,k}$ for $m \geq 2$ and $k \geq 1$:

$$L_{m,k} = \{(a^{m^2} b^{m^2})^{(k-1)/2} a^{m^2} c\}, \text{ for odd } k,$$
$$L_{m,k} = \{(a^{m^2} b^{m^2})^{k/2} c\}, \text{ for even } k.$$

Lower bounds

Consider the language family $L_{m,k}$ for $m \geq 2$ and $k \geq 1$:

$$L_{m,k} = \{(a^{m^2} b^{m^2})^{(k-1)/2} a^{m^2} c\}, \text{ for odd } k,$$
$$L_{m,k} = \{(a^{m^2} b^{m^2})^{k/2} c\}, \text{ for even } k.$$

$L_{m,k}$ can be accepted by a PDA with $O(m)$ states and $O(k)$ pushdown symbols whereas every NFA for $P(L_{m,k})$ needs at least $\Omega(m^2 \cdot k)$ states.

Lower bounds

Consider the language family $L_{m,k}$ for $m \geq 2$ and $k \geq 1$:

$$L_{m,k} = \{(a^{m^2} b^{m^2})^{(k-1)/2} a^{m^2} c\}, \text{ for odd } k,$$
$$L_{m,k} = \{(a^{m^2} b^{m^2})^{k/2} c\}, \text{ for even } k.$$

$L_{m,k}$ can be accepted by a PDA with $O(m)$ states and $O(k)$ pushdown symbols whereas every NFA for $P(L_{m,k})$ needs at least $\Omega(m^2 \cdot k)$ states.

Theorem

Let $M = \langle Q, \Sigma, \Gamma, \delta, q_0, Z_0, F \rangle$ be a PDA. Then, an NFA for $P(M)$ exists with $O(|Q|^2|\Gamma|)$ states. On the other hand, there exist infinitely many PDA $M_{Q,\Gamma}$ of size $O(|Q| \cdot |\Gamma|)$ such that every NFA accepting $P(M_{Q,\Gamma})$ needs $\Omega(|Q|^2|\Gamma|)$ states.

Special case (1): PDA that never pop

- Observe that $P(M) = \{u \in \Gamma^* \mid u \in Acc(q) \text{ and } q \in F\}$.

Special case (1): PDA that never pop

- Observe that $P(M) = \{u \in \Gamma^* \mid u \in Acc(q)$ and $q \in F\}$.
- An NFA for $P(M)$ then needs at most $|Q| \cdot |\Gamma| + 1$ states.

Special case (1): PDA that never pop

- Observe that $P(M) = \{u \in \Gamma^* \mid u \in Acc(q) \text{ and } q \in F\}$.
- An NFA for $P(M)$ then needs at most $|Q| \cdot |\Gamma| + 1$ states.
- It is here possible to find a tight lower bound:

Special case (1): PDA that never pop

- Observe that $P(M) = \{u \in \Gamma^* \mid u \in Acc(q) \text{ and } q \in F\}$.
- An NFA for $P(M)$ then needs at most $|Q| \cdot |\Gamma| + 1$ states.
- It is here possible to find a tight lower bound:

Lemma

For $m, k \geq 2$, there exist PDA $M_{m,k}$ which can never pop having m states and k pushdown symbols, for which every NFA for $P(M_{m,k})$ needs at least $k \cdot m + 1$ states.

Special case (2): stateless PDA

- General construction gives an upper bound of $3|\Gamma| + 6$.

Special case (2): stateless PDA

- General construction gives an upper bound of $3|\Gamma| + 6$.
- Improved construction gives an upper bound of $|\Gamma| + 1$.

Special case (2): stateless PDA

- General construction gives an upper bound of $3|\Gamma| + 6$.
- Improved construction gives an upper bound of $|\Gamma| + 1$.
- It is also possible to find a tight lower bound:

Special case (2): stateless PDA

- General construction gives an upper bound of $3|\Gamma| + 6$.
- Improved construction gives an upper bound of $|\Gamma| + 1$.
- It is also possible to find a tight lower bound:

Lemma

For any $k \geq 0$, there exists a stateless PDA M_k having $|\Gamma_k| = k + 1$ pushdown symbols, for which every NFA for $P(M_k)$ needs at least $k + 2 = |\Gamma_k| + 1$ states.

Special case (3): counter PDA

- For a counter PDA M, $P(M)$ is either $Z^* Z_0$ or $Z^{\leq h} Z_0$ for some fixed $h \geq 0$.

Special case (3): counter PDA

- For a counter PDA M, $P(M)$ is either $Z^* Z_0$ or $Z^{\leq h} Z_0$ for some fixed $h \geq 0$.
- It can be shown via pumping arguments that h is bounded by the number of states $|Q|$, if $P(M) = Z^{\leq h} Z_0$.

Special case (3): counter PDA

- For a counter PDA M, $P(M)$ is either $Z^* Z_0$ or $Z^{\leq h} Z_0$ for some fixed $h \geq 0$.
- It can be shown via pumping arguments that h is bounded by the number of states $|Q|$, if $P(M) = Z^{\leq h} Z_0$.
- Then, $|Q| + 2$ is an upper bound.

Special case (3): counter PDA

- For a counter PDA M, $P(M)$ is either Z^*Z_0 or $Z^{\leq h}Z_0$ for some fixed $h \geq 0$.
- It can be shown via pumping arguments that h is bounded by the number of states $|Q|$, if $P(M) = Z^{\leq h}Z_0$.
- Then, $|Q| + 2$ is an upper bound.
- Language $L_m = \{\lambda, a^m\}$ for $m \geq 1$ gives a tight lower bound.

Special case (3): counter PDA

- For a counter PDA M, $P(M)$ is either Z^*Z_0 or $Z^{\leq h}Z_0$ for some fixed $h \geq 0$.
- It can be shown via pumping arguments that h is bounded by the number of states $|Q|$, if $P(M) = Z^{\leq h}Z_0$.
- Then, $|Q| + 2$ is an upper bound.
- Language $L_m = \{\lambda, a^m\}$ for $m \geq 1$ gives a tight lower bound.

Lemma

Let M be a counter PDA with state set Q. Then, $P(M)$ is accepted by some NFA with size bounded by $|Q| + 2$. Moreover, this size bound is optimal.

Applications: complexity of decidability questions

Lemma

Let M be a PDA. Then, an NFA for $P(M)$ can be constructed in deterministic polynomial time.

Applications: complexity of decidability questions

Lemma

Let M be a PDA. Then, an NFA for $P(M)$ can be constructed in deterministic polynomial time.

➜ For the construction of the set $Acc(Q)$, the reachability of $O(|Q|^2|\Gamma|^2)$ pairs has to be tested.

Applications: complexity of decidability questions

Lemma

Let M be a PDA. Then, an NFA for $P(M)$ can be constructed in deterministic polynomial time.

- For the construction of the set $Acc(Q)$, the reachability of $O(|Q|^2|\Gamma|^2)$ pairs has to be tested.
- Each test can be seen as an instance of the emptiness problem for context-free languages which is in P.

Applications: complexity of decidability questions

Lemma

Let M be a PDA. Then, an NFA for $P(M)$ can be constructed in deterministic polynomial time.

- For the construction of the set $Acc(Q)$, the reachability of $O(|Q|^2|\Gamma|^2)$ pairs has to be tested.
- Each test can be seen as an instance of the emptiness problem for context-free languages which is in P.
- An NFA for $Acc(Q)$ can be constructed in deterministic polynomial time.

Applications: complexity of decidability questions

Lemma

Let M be a PDA. Then, an NFA for $P(M)$ can be constructed in deterministic polynomial time.

- For the construction of the set $Acc(Q)$, the reachability of $O(|Q|^2|\Gamma|^2)$ pairs has to be tested.
- Each test can be seen as an instance of the emptiness problem for context-free languages which is in P.
- An NFA for $Acc(Q)$ can be constructed in deterministic polynomial time.
- Similarly, an NFA for $Co\text{-}Acc(Q)$ can be constructed in deterministic polynomial time as well as for the intersection of both and the removal of the first symbol.

Applications: complexity of decidability questions

Lemma

Given a PDA M, it is P-complete to decide whether:
(i) $P(M)$ is a finite set. (ii) $P(M)$ is a finite set of words having at most length k, for a given $k \geq 1$.

Applications: complexity of decidability questions

Lemma

Given a PDA M, it is P-complete to decide whether:
(i) $P(M)$ is a finite set. (ii) $P(M)$ is a finite set of words having at most length k, for a given $k \geq 1$.

A PDA M is of constant height whenever there exists a constant $k \geq 1$ such that, for any word in $L(M)$, there exists an accepting computation along which the pushdown store never contains more than k symbols.

Applications: complexity of decidability questions

Lemma

Given a PDA M, it is P-complete to decide whether:
(i) $P(M)$ is a finite set. (ii) $P(M)$ is a finite set of words having at most length k, for a given $k \geq 1$.

A PDA M is of constant height whenever there exists a constant $k \geq 1$ such that, for any word in $L(M)$, there exists an accepting computation along which the pushdown store never contains more than k symbols.

Corollary

Given an unambiguous PDA M, it is P-complete to decide whether: (i) M is a constant height PDA. (ii) M is a PDA of constant height k, for a given $k \geq 1$.

Applications: complexity of decidability questions

Lemma

Given a PDA M, it is P-complete to decide whether $P(M)$ is a subset of Z^*Z_0.

Applications: complexity of decidability questions

Lemma

Given a PDA M, it is P-complete to decide whether $P(M)$ is a subset of Z^*Z_0.

Corollary

Given a PDA M, it is P-complete to decide whether M is essentially a counter machine.

Summary and open questions

- Tight bounds of $\Theta(|Q|^2|\Gamma|)$ in the general case.

Summary and open questions

- Tight bounds of $\Theta(|Q|^2|\Gamma|)$ in the general case.
- Better and also tight bounds for special cases.

Summary and open questions

- Tight bounds of $\Theta(|Q|^2|\Gamma|)$ in the general case.
- Better and also tight bounds for special cases.
- Some decidability questions are solvable in P and P-hard.

Summary and open questions

- Tight bounds of $\Theta(|Q|^2|\Gamma|)$ in the general case.
- Better and also tight bounds for special cases.
- Some decidability questions are solvable in P and P-hard.

- Consider other special cases, e.g., m-counter PDA or turn-bounded PDA.

Summary and open questions

- Tight bounds of $\Theta(|Q|^2|\Gamma|)$ in the general case.
- Better and also tight bounds for special cases.
- Some decidability questions are solvable in P and P-hard.

- Consider other special cases, e.g., m-counter PDA or turn-bounded PDA.
- Investigate trade-offs occurring when determinizing the NFA for $P(M)$.

Summary and open questions

- Tight bounds of $\Theta(|Q|^2|\Gamma|)$ in the general case.
- Better and also tight bounds for special cases.
- Some decidability questions are solvable in P and P-hard.

- Consider other special cases, e.g., m-counter PDA or turn-bounded PDA.
- Investigate trade-offs occurring when determinizing the NFA for $P(M)$.
- Extend the decidability of being a constant height PDA to arbitrary PDA.