Push Complexity

Optimal Bounds and Unary Inputs

Giovanni Pighizzini

Dipartimento di Informatica
Università degli Studi di Milano, Italy

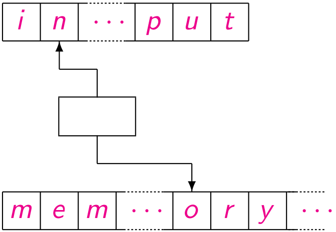
CIAA 2024 – Akita, Japan
September 4, 2024

UNIVERSITÀ DEGLI STUDI
DI MILANO

Computing with Very Restricted Resources

Turing machines

Space $s(n)$



$$s(n) = O(1)$$
$$\Downarrow$$
Regularity

Question ([Stearns&Hartmanis&Lewis II '65, Hopcroft&Ullman '69])

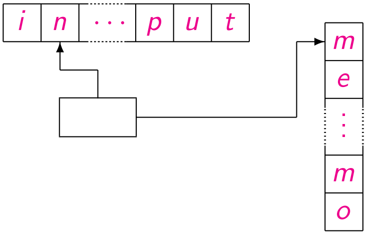
What is the minimal amount of space $s(n)$ which is necessary and sufficient for the recognition of nonregular languages?

For almost all the variants of Turing machines: $s(n) = \Theta(\log\log n)$

(one-way/two-way, deterministic/nondeterministic/alternating)

Computing with Very Restricted Resources

Pushdown automata



Pushdown Height
height(n)

$\text{height}(n) = O(1)$
$\Downarrow$
Regularity

Question ([P&Prigioniero '23])

How much should the height grow when it is non constant?

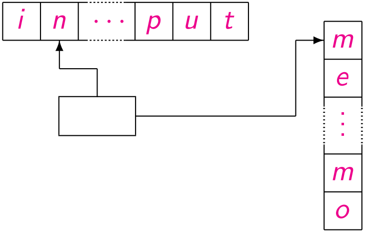
height(n) growing as:

- $\log \log n$, for input alphabets with at least 2 symbols
- $\log n$, for unary input alphabets

Another Measure for Pushdown Automata

Push complexity push(n)

number of *push operations* that are sufficient to accept inputs of length n [Bordhin&Mitrana '20]



$$\mathrm{push}(n) = O(1)$$
$$\Downarrow$$
$$\mathrm{height}(n) = O(1)$$
$$\Downarrow$$
$$\text{Regularity}$$

Question

What is the minimal push complexity for recognizing nonregular context-free languages?

Definitions

- PDA $\mathcal{M}$, input alphabet Σ
- $\mathcal{C}$ computation of $\mathcal{M}$:
$$\mathrm{push}_{\mathcal{M}}(\mathcal{C}) = \text{number of push operations executed in } \mathcal{C}$$
- $w \in \Sigma^*$:
$$\mathrm{push}_{\mathcal{M}}(w) = \begin{cases} \min\{\mathrm{push}_{\mathcal{M}}(\mathcal{C}) \mid \mathcal{C} \text{ accepting computation on } w\} & \text{if } w \in L(\mathcal{M}) \\ 0 & \text{otherwise} \end{cases}$$
- $n \in \mathbf{N}$:
$$\mathrm{push}_{\mathcal{M}}(n) = \max\{\mathrm{push}_{\mathcal{M}}(w) \mid |w| = n\}$$
- $L \subseteq \Sigma^*$:
$$\mathrm{push}_L = \min\{\mathrm{push}_{\mathcal{M}} \mid \mathcal{M} \text{ accepts } L\}$$

Observations

- $\mathrm{height}_{\mathcal{M}}(n) \leq \mathrm{push}_{\mathcal{M}}(n)$
- $\mathrm{height}_{\mathcal{M}}(n) = \Theta(\mathrm{push}_{\mathcal{M}}(n))$, for 1-turn PDAs
- $\mathrm{push}_{\mathcal{M}}(n) = O(1) \implies L(\mathcal{M})$ is regular
- $\mathrm{push}_{L}(n) = O(1) \iff \mathrm{height}_{L}(n) = O(1) \iff L$ is regular

Optimal Lower Bounds for Push Complexity

Questions

[Bordhin&Mitrana '20]:

- There exist languages with push complexity $O(\log n)$ and $O(\sqrt{n})$
- Does there exist some nonregular language with push complexity $O(f(n))$ for some other sublinear function f?

How small can such f be?

Question (languages)

Find the "smallest" function f s.t. $\mathrm{push}_L(n) = O(f)$ for some nonregular language L

Question (machines)

Find the "smallest" function f s.t. $\mathrm{push}_{\mathcal{M}}(n) = O(f)$ for some PDA $\mathcal{M}$ making a nonconstant number of push operations

Questions

Theorem ([Alberts '85])

If a Turing Machine works in space $s(n) = o(\log \log n)$ then it works in constant space.

$\Rightarrow$ If $\text{push}_L(n)$ and $\text{push}_{\mathcal{M}}(n)$ are not bounded by any constant, then they must grow at least as $\log \log n$

Can this $\log \log(n)$ bound be reached?

The Language REI
[Bednárová&Geffert&Reinhardt&Yakaryilmaz '16]

Set of strings that are not prefixes of the infinite word $bc_1ac_2^Rbc_2ac_3^R \cdots bc_kac_{k+1}^Rbc_{k+1}ac_{k+2}^R \cdots$, where

- $c_k = eb_0db_{k,0}db_0^R\,eb_1db_{k,1}db_1^R \cdots eb_{\lfloor \log k \rfloor}db_{k,\lfloor \log k \rfloor}db_{\lfloor \log k \rfloor}^R e$ is a counter representation for k, augmented with subcounters
- $b_{k,i} \in \{0,1\}$ is the ith bit in the binary representation of k, and $b_i \in \{0,1\}^*$ denotes the number i written in binary, for $i \in \{0, 1, \ldots, \lfloor \log k \rfloor\}$

Theorem ([Bednárová&Geffert&Reinhardt&Yakaryilmaz '16])

REI *is a nonregular language accepted by a* PDA $\mathcal{M}$ *using height* $O(\log \log n)$

An inspection to the definition of $\mathcal{M}$ shows that each accepting computations makes at most 1 turn

$\Rightarrow$ The language REI and the PDA $\mathcal{M}$ have minimal non-constant push complexity $O(\log \log n)$

Decidability

Decidability Questions

Problem (Languages)

Given a CFL L is $\mathrm{push}_L(n) = O(1)$*?*
is $\mathrm{height}_L(n) = O(1)$*?*

Undecidable!

Equivalent to the Regularity Problem for CFLs

Problem (Machines)

Given a PDA $\mathcal{M}$ *is* $\mathrm{push}_{\mathcal{M}}(n) = O(1)$*?*
is $\mathrm{height}_{\mathcal{M}}(n) = O(1)$*?*
($L(\mathcal{M})$ could be regular)

Undecidable!

[Bordhin&Mitrana '20, P&Prigioniero '23]

All these undecidability results are proved using an input alphabet of at least 2 symbols!

Decidability Questions in the Unary Case

Problem (Languages)

Given a unary CFL L is $\text{push}_L(n) = O(1)$*?*
is $\text{height}_L(n) = O(1)$*?*

Always true!

Unary CFLs are regular [Ginsburg&Rice '62]

Problem (Machines)

Given a unary PDA $\mathcal{M}$*:*
is $\text{push}_{\mathcal{M}}(n) = O(1)$*?*
is $\text{height}_{\mathcal{M}}(n) = O(1)$*?*
($L(\mathcal{M})$ is regular)

$\text{height}_{\mathcal{M}}$: decidable! [P&Prigioniero '23]

$\text{push}_{\mathcal{M}}$: ???

Decidability in the Unary Case

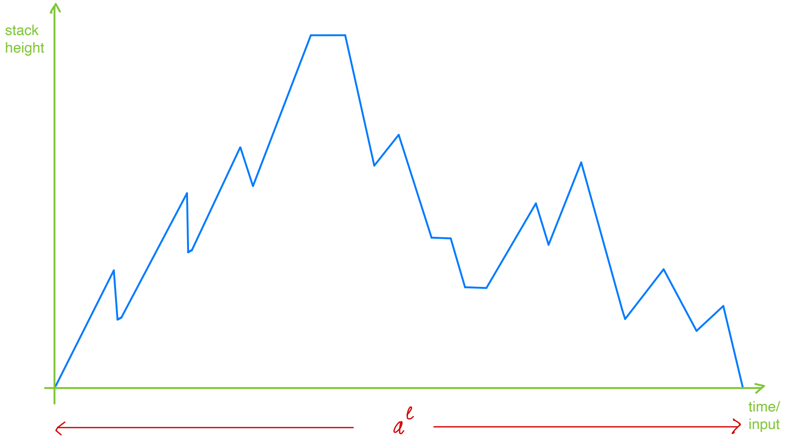
Theorem

Given a unary PDA $\mathcal{M}$, *it is decidable whether* $\mathrm{push}_{\mathcal{M}}(n)$ *is bounded by some constant.*

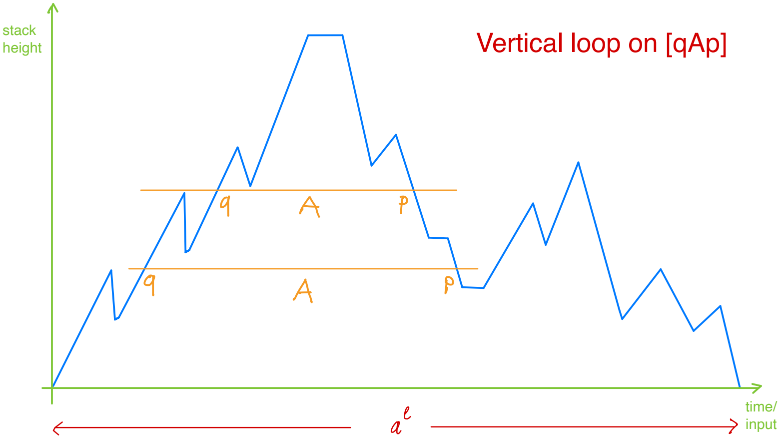
Proof Idea:

- Each accepting computation on a sufficiently long input a^{ℓ} should contain some repetitions
- When possible, replace parts between repetitions by loops that do not use any push
- $\mathrm{push}_{\mathcal{M}}(n) = O(1)$ iff the replacement is possible for each $a^{\ell} \in L(\mathcal{M})$, with finitely many exceptions

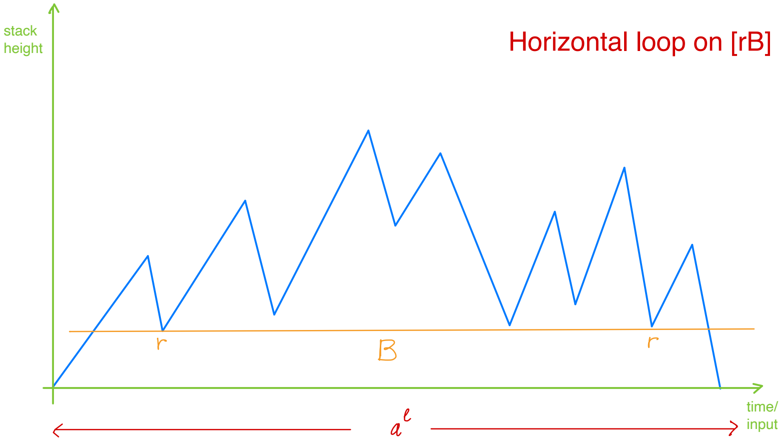
Computations of PDAs



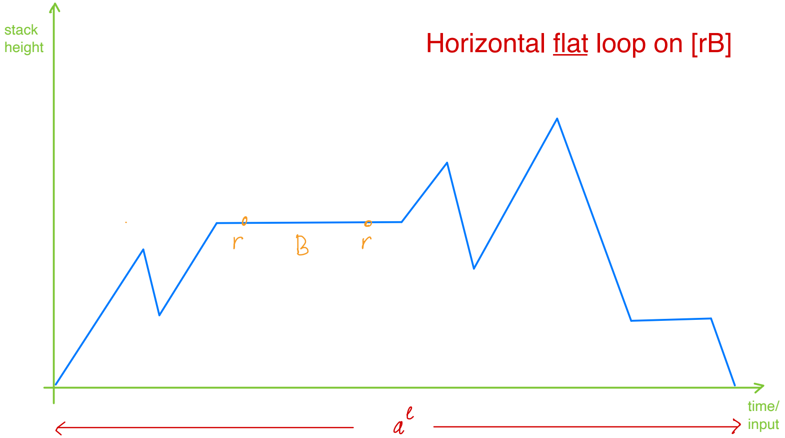
Loops in Computations



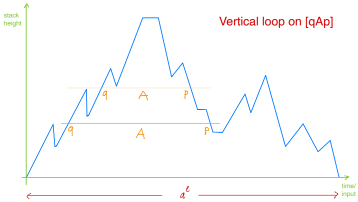
Loops in Computations

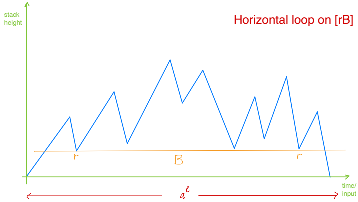


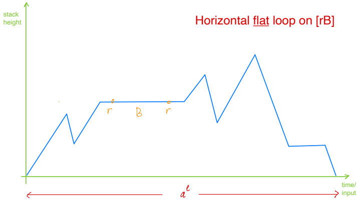
Loops in Computations



Loops in Computations







Flat loops do not make any push

$\Downarrow$

When possible, use them to simulate vertical and horizontal nonflat loops

Using Flat Loops

Lemma

If a^ℓ has an accepting computation $\mathcal{C}$ visiting a pair $[rB]$, where $[rB]$ has a flat loop, then a^ℓ has also an accepting computation $\mathcal{C}'$ with $\mathrm{push}(\mathcal{C}') \leq H$ (H is a constant depending on $\mathcal{M}$).

We consider languages:

- L_{f}: strings accepted by computations of $\mathcal{M}$ which visit at least one pair $[rB]$ having a flat loop
- L_{nf}: strings accepted by the computations of $\mathcal{M}$ which visit only pairs that do not have flat loops

Then:

$$\mathrm{push}_{\mathcal{M}}(n) = O(1) \iff L_{\mathrm{nf}} \setminus L_{\mathrm{f}} \text{ is finite}$$

L_{f} and L_{nf} are regular languages effectively constructible from $\mathcal{M}$

$\Longrightarrow$ "$L_{\mathrm{nf}} \setminus L_{\mathrm{f}}$ is finite" is decidable!

Optimal Lower Bounds in the Unary Case

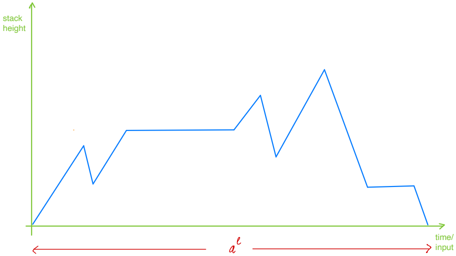
Lower Bound on $\mathrm{push}_{\mathcal{M}}(n)$

- We have seen that $\mathrm{push}_{\mathcal{M}}(n) \notin o(\log \log n)$ when it is not bounded by any constant
- There is a PDA $\mathcal{M}$ matching such a bound (language $\mathtt{REI}$)

What happens if the input alphabet is unary?

Lower Bounds on $\text{push}_{\mathcal{M}}(n)$

Unary case



Each sequence of m moves that do not change the stack, with $m \geq \#states$, contains a horizontal flat loop!

Theorem

Let $\mathcal{M}$ be a unary PDA*.*
If $\text{push}_{\mathcal{M}}(n) \notin O(1)$ *then* $\text{push}_{\mathcal{M}}(n) \notin o(n)$*,*
namely it must grow at least linearly in n.

Bounds

	general input	unary input
$\text{height}_{\mathcal{M}}(n)$	$\log \log n$	$\log n$
l.b.	[Alberts '85]	[P&Prigioniero '23]
u.b.	[P&Prigioniero '23]	*ibid*
$\text{push}_{\mathcal{M}}(n)$	$\log \log n$	n
l.b.	[Alberts '85]	[This work]
u.b.	[This work]	easy

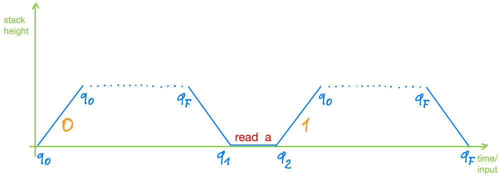
"Simultaneous" Optimal Bounds in the Unary Case

Theorem

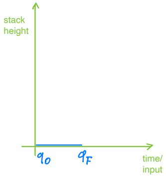
There exists a unary PDA $\mathcal{M}$ *accepting in nonconstant height s.t.* $\text{height}_{\mathcal{M}}(n) = O(\log n)$ *and* $\text{push}_{\mathcal{M}}(n) = O(n)$.

$\mathcal{M}$ accepts a^* (in a complicate way...):

- Two recursive calls with one read in between



- Base of the recursion: accept ε



- On input a^ℓ there exists an accepting computation $\mathcal{C}$ with:
  - $\text{push}_{\mathcal{M}}(\mathcal{C}) = 2\ell$
  - $\text{height}_{\mathcal{M}}(\mathcal{C}) = \lfloor \log_2 \ell \rfloor + 1$
- The other computations are more expensive

Conclusion

Summary and Problems

- For a PDA $\mathcal{M}$, both $\text{push}_{\mathcal{M}}(n)$ and $\text{height}_{\mathcal{M}}(n)$, if non constant, must grow at least as $\log\log n$
- These bounds are reachable in the case of binary alphabets

Problem

Can these $\log\log n$ bounds be reached if $\mathcal{M}$ accepts a regular language?

Open!

- For unary alphabets the optimal bounds grow as n and $\log n$, resp.
- In the general case, given a PDA $\mathcal{M}$ it is not decidable whether $\text{push}_{\mathcal{M}}(n)$ (resp., $\text{height}_{\mathcal{M}}(n)$) is bounded by a constant.
  In the unary case these two questions are decidable

Problem

Are these questions decidable, if we know that $L(\mathcal{M})$ is regular?

Still undecidable!

Thank you for your attention!