The minimum amount of useful space: New results and new directions

Abuzer Yakaryılmaz
National Laboratory for Scientific Computing, Brazil

Klaus Reinhardt
University of Tubingen & Humboldt University of Berlin, Germany

DLT 2014 (Ekaterinburg)
August 29, 2014

Realtime deterministic finite automata (rtDFA) defines the class of REGULAR LANGUAGES!

And, even two-way alternation does not change the class!

A fundamental question is

What is the minimum amount of resource to add a rtDFA to recognize a non-regular language?

What is the minimum amount of time?

What is the minimum amount of <u>space</u>?

Three parameters:

Computation modes: deterministic, nondeterministic, alternating, probabilistic, and quantum.

Tape heads: realtime, one-way, and two-way.

Memory types: counters, stack, and tapes.

and

Unary versus general alphabet languages

Three ways of defining space

A language is said to be recognized by a machine in **X**-type s(|w|) space:

1) **Strong** space: The space used by the machine on each input w is at most s(|w|).
2) **Middle** space: The space used by the machine on each member w is at most s(|w|).
3) **Weak** space: The minimum amount of space used by the machine in a single accepting path (tree) for each member w is at most s(|w|).

Two-way Turing machines

We have a complete picture:

Any language recognized by a ATM in weak o(loglogn) space is regular.

There is a unary non-regular language recognizes by a DTM in weak O(loglogn) space.

One-way and realtime Turing machines

Table 1. Minimum space used by one-way TMs for recognizing nonregular languages.

	General input alphabet			Unary input alphabet		
	Strong	Middle	Weak	Strong	Middle	Weak
Deterministic TM	$\log n$	$\log n$	$\log n$	$\log n$	$\log n$	$\log n$
Nondeterministic TM	$\log n$	$\log n$	$\log\log n$	$\log n$	$\log n$	$\log\log n$
Alternating TM	$\log n$	$\log\log n$	$\log\log n$	$\log n$	$\log n$	$\log\log n$

Open Problem 1 *[26] Are the double logarithmic lower bounds for the recognition of the nonregular unary languages by real-time nondeterministic and alternating TMs tight?*

Pushdown automata

- No weak o(n)-space bounded one-way deterministic PDAs can recognize any non-regular language!
- Realtime deterministic PDAs can recognize the language having equal number of a's and b's.
- For one-way nondeterministic PDAs, a weak logarithmic space algorithm was given for a non-regular language.
  - We improve this bound to *loglog(n)* space also for realtime head.

REI composed by the non-prefixes of the following infinite word

$$bc_0ac_1^Rbc_1a\cdots bc_kac_{k+1}^Rbc_{k+1}ac_{k+2}^Rb\cdots ,$$

where

- $c_k = eb_1db_{k,1}db_1^Reb_2db_{k,2}db_2^Reb_3db_{k,3}db_3^Re\cdots eb_{k,\lceil logk\rceil}db_{\lceil logk\rceil}db^R_{k,\lceil logk\rceil}e$ is a counter representation for k augmented with subcounters,
- $b_i \in \{0,1\}^*$ is the binary representation of i, and
- $b_{k,i} \in \{0,1\}$ is the i-th last bit (value 2^i) in the binary representation of k.

Each b_i is associated with $b_{k,i}$. The length of $b_{k,\lceil logk\rceil}$ is $O(loglog(k))$.

Theorem 1. *Realtime nondeterministic PDAs can recognize nonregular language* REI *with weak* $\log\log n$ *space.*

This bound is also tight for one-way/realtime alternating PDAs since $o(\log\log n)$ weak-space alternating TMs cannot recognize any nonregular language [20].

Open Problem 2 *What are the tight strong/middle space bounds for one-way/realtime nondeterministic and alternating PDAs for the recognition of nonregular languages?*

Unary languages and PDAs

It is a well-known fact that one-way nondeterministic PDA cannot recognize any nonregular language.

One-way alternating PDAs are quite powerful and they can recognize any language in linear alternating space (space inefficient).

What about alternating counter-automaton (CA)?

$$\mathtt{UPOWER} = \{a^{2^n} \mid n \geq 0\} \text{ and } \mathtt{UPOWER+} = \{a^{2^n+2n} \mid n \geq 0\}$$

Theorem 2. *Realtime alternating CAs can recognize nonregular* UPOWER+ *in weak logarithmic space.*

Open Problem 3 *What are the tight space bounds for realtime/one-way alternating CAs for the recognition of nonregular unary and binary languages?*

A trade-off to alternation depth:

Our alternating algorithm has a linear alternation depth (for the members).

We also present a realtime algorithm for UPOWER with logarithmic alternation depth but it uses a linear counter.

$$\begin{array}{|c|c|}\hline b_{k,j} & b_{k,j-1} \\ \hline b_{k-1,j} & b_{k-1,j-1} \\ \hline \end{array}$$

$b_{k,j}$ can be determined by $b_{k,j-1}$, $b_{k-1,j}$, and $b_{k-1,j-1}$.

$$\begin{array}{|c|c|}\hline 0&0\\ \hline 0&0\\ \hline\end{array}\quad
\begin{array}{|c|c|}\hline 1&0\\ \hline 0&1\\ \hline\end{array}\quad
\begin{array}{|c|c|}\hline 1&0\\ \hline 1&0\\ \hline\end{array}\quad
\begin{array}{|c|c|}\hline 0&0\\ \hline 1&1\\ \hline\end{array}\quad
\begin{array}{|c|c|}\hline 0&1\\ \hline 0&0\\ \hline\end{array}\quad
\begin{array}{|c|c|}\hline 0&1\\ \hline 0&1\\ \hline\end{array}\quad
\begin{array}{|c|c|}\hline 1&1\\ \hline 1&0\\ \hline\end{array}\quad
\begin{array}{|c|c|}\hline 1&1\\ \hline 1&1\\ \hline\end{array}$$

$b_{k,1}$ can be determined by $b_{k-1,1}$.

Let a^m be the input. The head position and the value of the counter represent a binary number (b_k, j):

- the head is on $(m-k)$th symbol and
- the value of the counter is j.

The automaton nondeterministically picks a value of k at the beginning and then universally enters:

$$(b_{m,1}, 1), (b_{m,2}, 0), \ldots, (b_{m,k}, 0).$$

The automaton represents 2^k at this point. By reading the input, the automaton decrements the counter. If it hits zero, then $2^k = m$.

Two-way PDAs

In the case of two-way PDAs, we have tight bounds since a two-way deterministic PDAs can recognize REI with strong $\log \log n$-space.

Theorem 3. *Two-way deterministic PDAs can recognize* REI *in strong* $\log \log n$*-space.*

Unary languages:

Any unary language recognized by a two-way deterministic PDA using sublinear space on its stack is regular.

Two-way deterministic CAs can recognize nonregular unary language UPOWER in linear space.

Therefore, linear space is a tight bound for both two-way deterministic PDAs and CAs.

Currently, we do not know whether nondeterminism or using random choices can help for unary languages.

Multi counter/pushdown automata

Another interesting direction is to identify the tight space bounds for one-way/realtime multi-counter/pushdown automata.

We only know that realtime deterministic automata with *k* counters can recognize some non-regular languages in middle $O(n^{\frac{1}{k}})$ space, where *k>1.*

Probabilistic and quantum machines

- Probabilistic models are special cases of their quantum counterparts.
- Realtime probabilistic finite automata (PFAs) can recognize unary non-regular languages with unbounded-error.
- Bounded-error:
  - One-way PFAs recognize only regular languages.
  - Two-way PFAs can recognize some non-regular languages but only with exponential-expected time.
  - Two-way probabilistic TMs can recognize some non-regular languages in polynomial time with an arbitrary small space.
  - One-way probabilistic TMs cannot recognize any non-regular language in space $o(\log\log(n))$.

Probabilistic and quantum machines - 2

- Bounded-error:
  - Realtime quantum finite automata (QFAs) can recognize only regular languages.
  - Two-way QFAs can recognize some non-regular languages in polynomial expected time.
  - If the head is quantum, then one-way QFAs can recognize some non-regular languages in linear time.

Probabilistic and quantum machines **Unary** languages

- Bounded-error:
  - Two-way PFAs can recognize only regular languages.
  - The question is open for two-way QFAs.
  - We show that if two-way QFA (with classical head) has a classical counter, then they can recognize UPOWER by using logarithmic space for the members.

Quantum algorithm for UPOWER

- We know that two-way QFAs can recognize POWER

  language. $POWER = \{ a^n b^{2^n} | n > 0 \}$

  Let M be such an machine:
  - The members are accepted exactly.
  - The non-members are rejected with probability at least 0.8.
  - So, when M rejects, it is certain that the input is not in the language.
- By using a counter, we can iteratively mark 1,2,3,...,i,... symbols.
  - In each iteration, we execute M on $a^i a^n$.
    - If n is not a power of i, then M rejects. So, as long as getting “reject”, we continue the iteration.
      - For members of UPOWER, i never takes the value of n.
      - By carefully selecting a small accepting probability, we obtain our algorithm.

Bounded-error probabilistic pushdown automata

One-way unary probabilistic PDAs can recognize only regular languages.

- The question is open for one-way quantum PDAs.

- Realtime probabilistic PDAs can recognize the following binary language by using middle logarithmic space:

$$\{b_1 a b_2^R a b_3 a b_4^R a \cdots a b_{2k-1} a b_{2k}^R \mid k > 0\}$$

THANK YOU!

QUESTIONS?