THEORIE-TAG 2023
A. Lin, G. Zetzsche (Eds.): Theorietag 2023, Kaiserslautern, 4. – 6.10.2023
RPTU & MPI-SWS Kaiserslautern, Technical Report, pp. 1–4.

Checking Directedness of Regular and Context-free Languages

Moses Ganardi[(A)] Irmak Sağlam*[(A)] Georg Zetzsche[(A)]

[(A)]Max Planck Institute for Software Systems (MPI-SWS), Germany
{mganardi,isaglam,georg}@mpi-sws.org

1. Abstract

In this work we focus on the following problem: Given a downward closed language L, what is the complexity of deciding L's upward directedness? We study the problem on regular L and context-free L given via their grammars. We show that if L is a regular language on a fixed alphabet (alphabet is a part of the input), the problem is in NL, whereas if L is on an arbitrary alphabet it is in AC^1. On the other hand, if L is given as a CFG, we show the problem to be PSPACE-complete.

2. Preliminaries

Subword ordering. Let w_1 and w_2 be two finite words on the finite alphabet Σ. Then w_1 is said to be a *subword* of w_2, if we can get w_1 by deleting some letters of w_2.
Downward closedness and upward directedness A language L over the finite alphabet Σ is called *downward closed*, if for every word $w \in L$, all subwords of w are also in L. Similarly, L is called *upward directed* (or for short, *directed*) if for any two words $w_1, w_2 \in L$ there exists a word $w_3 \in L$ such that both w_1 and w_2 are subwords of w_3.
Ideals. Ideals I over Σ are downward closed and directed subsets of Σ^*, and the languages they accept can be represented as a concatenation of languages accepted by *atoms*, as follows:

$$I = L(A_1 A_2 \dots A_n)$$

$A_1, \dots, A_n$ are called *atoms* over Σ and are languages either of shape $\{a, \epsilon\}$ for some $a \in \Sigma$ or of shape Δ^* for some $\Delta \subseteq \Sigma^*$. $A_1 A_2 \dots A_n$ is called a representation of ideal I.

Clearly, one ideal can have different representations, e.g. $I = \{a, \epsilon\}\{a\}^* = \{a\}^*$.

It is known that all downward closed languages over Σ can be written as a finite union of their ideals. That is, for a downward language L, there exists a finite set of ideals $\mathcal{I}$ such that

$$L = \bigcup_{I \in \mathcal{I}} I \tag{1}$$

$\mathcal{I}$ is called a *ideal decomposition* of L.
Reduced ideals. We call an ideal representation $A_1A_2\ldots A_n$ *reduced* if for all $i \in [1, n-1]$, neither the language of A_i contains the language of A_{i+1}, or vice versa.

We show that all ideals have a reduced representation. Therefore, a downward closed language can also be written as a finite union of their reduced ideals, as in (1).
Weight function. Next, we define a function μ_k that assigns a weight to each ideal representation, and call μ_k the *weight function*.
Formally, $\mu_k(A_1 \ldots A_n) = \sum_{i=1}^{n} \mu_k(A_i)$ where

$$\mu_k(A_i) = \begin{cases} 1, & \text{if } A_i = \{a, \epsilon\} \text{ for some } a \in \Sigma, \\ (k+1)^{|\Delta|}, & \text{if } A_i = \Delta^* \text{ for some } \Delta \subseteq \Sigma \end{cases}$$

We show that for two reduced ideal representations $A_1 \ldots A_n$ and $B_1 \ldots B_m$ representing ideals I and J, respectively,

$$\text{if } I \subseteq J, \text{ then } \mu_k(A_1 \ldots A_n) \leq \mu_k(B_1 \ldots B_m) \text{ for any } k \geq \max(n, m) \tag{2}$$

Moreover, if $I \subsetneq J$, then the inequality is strict.

3. Main approach

Our main approach to tackling the problem, both for regular and context-free L is to efficiently manipulate the finite abstraction the language is given by, to obtain a similar model that accepts a reduced ideal decomposition of L. That is, in the case of regular L we will get a DFA that accepts a finite language of reduced ideals of L; and in the case of context-free L we will get a CFG that accepts the same. In both of these models, each accepted word of the model gives one reduced ideal in the decomposition of L. The union of all these ideals gives L. Then, we efficiently check the weights of all accepted words in the respective model, and obtain an ideal with the maximum weight $I_{\max}$ (according to μ_k where k is the length of the longest accepted word).

Since L is downward closed, checking its directedness correspond to checking whether L is an ideal itself. Due to the inclusion result we have on the weight function (2), if L is an ideal itself, then it should be contained by $I_{\max}$ (Note that $I_{\max} \subseteq L$ trivially holds). Then the result of this inclusion check, gives us the answer to the directedness of L.

To efficiently transform the abstractions that accepts L into models that accept the reduced ideal decompositions of L, we use transducers. We show that this translation can be achieved in NL for regular languages and P for context-free languages.

Then we show that the maximum weighted-path in the model can be obtained in NL for regular languages with fixed alphabets, AC^1 for regular languages with arbitrary alphabets and in P for context-free languages. In the regular case, we calculate the maximum-weighted path in the DFA my using max-plus semiring; and in the context-free case we use dynamic programming to obtain the path of the CFG that has the maximum weight.

Then we use the existing result [1] to check for the inclusion $L \subseteq I_{\text{max}}$ for regular languages in NL, and we show the inclusion can be checked in PSPACE for context-free languages by simply guessing an ideal representation of $I \in \mathcal{I}$ that does not embed in I_{max} and checking the whether I embeds in I_{max} atom by atom.

Lastly, we give a matching lowerbound for the context-free case. In particular, we reduce a known PSPACE-hard membership problem in straight-line programs to the problem of checking whether a language given by a CFG is contained in an ideal.

Literatur

[1] G. ZETZSCHE, The Complexity of Downward Closure Comparisons. In: I. CHATZIGIANNAKIS, M. MITZENMACHER, Y. RABANI, D. SANGIORGI (eds.), *43rd International Colloquium on Automata, Languages, and Programming, ICALP 2016, July 11-15, 2016, Rome, Italy*. LIPIcs 55, Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2016, 123:1–123:14. `https://doi.org/10.4230/LIPIcs.ICALP.2016.123`