

Fine-grained Reductions Around CFL Reachability

ALEKSANDRA ISTOMINA, Saint Petersburg State University, Russia and JetBrains Research, Russia
RESEARCH ADVISOR: SEMYON GRIGOREV, Saint Petersburg State University, Russia and JetBrains Research, Russia

1 INTRODUCTION

CFL Reachability problem finds application in different fields of research: static code analysis (e.g. type-based flow analysis [8] or points-to analysis [11, 12]), graph databases, bioinformatics. It finds valid paths between vertices in a graph.

There are several cubic [9, 14] and slightly subcubic [3] algorithms for CFL reachability. The big open question is whether truly subcubic algorithm exists.

maybe we can prove that no such algorithm exist under some hypothesis

1.1 main problem

fine-grained complexity has some results in the area

results are scattered, have no structure

maybe everything is already proven

1.2 main goals, overview

collect existing results into easy-to-read form

focus on static, dynamic problems are omitted from this research

state open problems

2 PRELIMINARIES

Context-free grammar (CFG) is a triple $G = (N, \Sigma, R)$, where N is a set of nonterminals, Σ is a set of terminals and R is a set of productions of the followings form: $A \rightarrow \alpha$, $\alpha \in (N \cup \Sigma)^*$. Denote a context-free language of words derived from the starting symbol $S \in N$ as $L_S(G)$.

CFG Recognition problem is to decide whether $w \in L(G)$ given a CFG G and a string $w \in \Sigma^*$. This problem is closely related to *CFG parsing problem* where we want a possible derivation sequence, if $w \in L(G)$. It is known [10] that CFG recognition is as hard as CFG parsing up to logarithmic factors.

Let $D = (V, E)$ be a directed graph which edges are labelled with symbols from R . We call a path from vertex v to vertex u an S -path if concatenation of labels on that path is a word from $L_S(G)$. *Context-free-language (CFL) Reachability* problem [7] is to determine if there exists an S -path between a given sets of vertices A, B . In *single source/single target (s-t)* CFL Reachability $A = \{s\}$, $B = \{t\}$, $s, t \in V(D)$. In *all-pairs* CFL Reachability $A = B = V(D)$.

Dyck-k reachability problem is a CFL reachability problem where G defines a Dyck language on k types of parentheses.

For problems P, Q and time bounds t_P, t_Q , a *fine-grained reduction* [1] from (P, t_P) to (Q, t_Q) is an algorithm that, given an instance I of P , computes an instance J of Q such that:

- I is a YES-instance of P if and only if J is a YES-instance of Q ,
- for any $\epsilon > 0$ there is a $\delta > 0$ such that $t_Q(|J|)^{1-\epsilon} = O(t_P(|I|)^{1-\delta})$,
- the running time of the reduction is $O(t_P(|I|)^{1-\gamma})$ for some $\gamma > 0$.

3 MAIN RESULTS

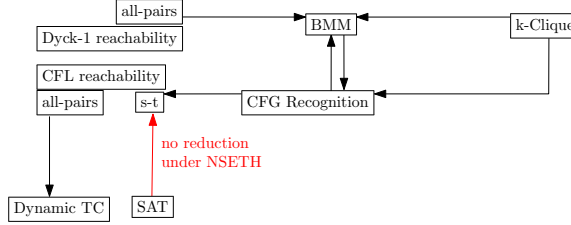


Fig. 1. Black arrow $a \rightarrow b$ represent existing reduction from a to b , directions of future work are marked with dashed arrows. Red and turquoise arrows analogously represent non-existence of the reduction and open problems respectively.

3.1 existing problems and hypotheses

There are several problems that are connected with CFL Recognition and Reachability.

Boolean satisfiability problem (SAT, k -SAT) is to determine if there exists an interpretation of variables that satisfies a given Boolean formula on n variables written in k -CNF, $k > 3$. The hypothesis about SAT, that we are interested about, is NSETH [2] which proposes that there is no $\epsilon > 0$ such that k -SAT can be solved co-nondeterministically in time $2^{(1-\epsilon)n}$ for any k .

In *Boolean Matrix Multiplication (BMM)* problem it is needed to calculate matrix product of the two given $n \times n$ matrices over (AND, OR). BMM hypothesis states that there is no $O(n^{3-\epsilon})$ combinatorial algorithm for that.

Orthogonal Vectors (OV) problem decides whether two sets X, Y of n boolean d -dimensional vectors contain a pair $x \in X, y \in Y$ which dot product $x \cdot y$ equals zero. Hypothesis states that OV problem can not be solved in $O(n^{2-\epsilon} \cdot \text{poly}(d))$ time.

Given a context free language $L(G)$ the *Language Edit Distance (LED)* problem seeks the minimum number of edits (insertions, deletions and substitutions) required to convert the given string s into a member of $L(G)$.

The incremental *dynamic transitive closure (DTC)* [4] problem asks to maintain reachability information in a directed graph between arbitrary pairs of vertices under insertions of edges.

3.2 existing reductions

CLFLR \leftrightarrow SetConstr

The problem has been shown to be 2NPDA-hard [5], which yields a conditional cubic lower bound in its complexity.

dynamic TC to all-pairs

BMM to Dyck-1

s-t \rightarrow cfg \rightarrow bmm \Rightarrow no combinatorial algorithm

short s-t certificates \Rightarrow no reduction from SAT

3.3 open problems

This paper is a part of research dedicated to determination of existence or non-existence of a truly subcubic algorithm for CFL Reachability. Apart from that there are several open problems in the area worth mentioning.

It is known [13] that detecting one triangle of a total negative edge weight in a weighted graph and up to $n^{3-\epsilon}$, $\epsilon > 0$ such triangles are subcubic equivalent tasks. Analogous question arises for s-t and all-pairs CFL reachability.

There exists a reduction from Andersen pointer analysis [6] to OV problem where as a part of reduction appears slightly modified Dyck-1 reachability with additional if-condition for edge existence in a graph. The question is if similar reduction exists to pure Dyck-1 problem. Existence of such a reduction would provide a conditional quadratic lower bound on Dyck-1 reachability problem, now there is no non-trivial one.

All-pairs shortest paths (APSP) problem and its subcubic equivalent analogues [13] is connected to problems on paths. Finding a fine-grained reduction from APSP to CFL reachability would give a conditional lower bound on a complexity of a problem.

4 CONCLUSION AND FUTURE WORK

CFL reachability is a popular problem strongly connected with many areas. It has several cubic algorithms and no truly subcubic combinatorial one under BMM hypothesis. It has been proved that we can't get cubic lower bound on CFL reachability using reduction from SAT under NSETH.

Still other reductions may be possible. For example, OMV problem is connected to dynamic problems as is CFL reachability and APSP to problems on paths. Getting cubic lower bound through them is a possible way of future work.

This overview was focused on static problems and reductions between them. There is a big field of dynamic problems where are many reductions of our interest. We plan to extend our research there.

5 ACKNOWLEDGMENTS

REFERENCES

- [1] Karl Bringmann. 2019. Fine-Grained Complexity Theory. In *36th International Symposium on Theoretical Aspects of Computer Science*. 1.
- [2] Marco L. Carmosino, Jiawei Gao, Russell Impagliazzo, Ivan Mihajlin, Ramamohan Paturi, and Stefan Schneider. 2016. Nondeterministic Extensions of the Strong Exponential Time Hypothesis and Consequences for Non-Reducibility. In *Proceedings of the 2016 ACM Conference on Innovations in Theoretical Computer Science (Cambridge, Massachusetts, USA) (ITCS '16)*. Association for Computing Machinery, New York, NY, USA, 261–270. <https://doi.org/10.1145/2840728.2840746>
- [3] Swarat Chaudhuri. 2008. Subcubic Algorithms for Recursive State Machines. In *Proceedings of the 35th Annual ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages (San Francisco, California, USA) (POPL '08)*. Association for Computing Machinery, New York, NY, USA, 159–169. <https://doi.org/10.1145/1328438.1328460>
- [4] Kathrin Hanauer, Monika Henzinger, and Christian Schulz. 2020. Faster Fully Dynamic Transitive Closure in Practice. *ArXiv abs/2002.00813* (2020).
- [5] Nevin Heintze and David McAllester. 1997. On the Cubic Bottleneck in Subtyping and Flow Analysis. In *Proceedings of the 12th Annual IEEE Symposium on Logic in Computer Science (LICS '97)*. IEEE Computer Society, USA, 342.
- [6] Anders Alnor Mathiasen and Andreas Pavlogiannis. 2021. The Fine-Grained and Parallel Complexity of Andersen's Pointer Analysis. *Proc. ACM Program. Lang.* 5, POPL, Article 34 (Jan. 2021), 29 pages. <https://doi.org/10.1145/3434315>
- [7] David Melski and Thomas Reps. 1997. Interconvertibility of Set Constraints and Context-Free Language Reachability. *SIGPLAN Not.* 32, 12 (Dec. 1997), 74–89. <https://doi.org/10.1145/258994.259006>
- [8] Jakob Rehof and Manuel Fähndrich. 2001. Type-Base Flow Analysis: From Polymorphic Subtyping to CFL-Reachability. *SIGPLAN Not.* 36, 3 (Jan. 2001), 54–66. <https://doi.org/10.1145/373243.360208>
- [9] Thomas Reps, Susan Horwitz, and Mooly Sagiv. 1995. Precise Interprocedural Dataflow Analysis via Graph Reachability. In *Proceedings of the 22nd ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages (San Francisco, California, USA) (POPL '95)*. Association for Computing Machinery, New York, NY, USA, 49–61. <https://doi.org/10.1145/199448.199462>
- [10] Walter L. Ruzzo. 1979. On the Complexity of General Context-Free Language Parsing and Recognition (Extended Abstract). In *Proceedings of the 6th Colloquium, on Automata, Languages and Programming*. Springer-Verlag, Berlin, Heidelberg, 489–497.

- [11] Manu Sridharan and Rastislav Bodik. 2006. Refinement-Based Context-Sensitive Points-to Analysis for Java. *SIGPLAN Not.* 41, 6 (June 2006), 387–400. <https://doi.org/10.1145/1133255.1134027>
- [12] Manu Sridharan, Denis Gopan, Lexin Shan, and Rastislav Bodik. 2005. Demand-Driven Points-to Analysis for Java. *SIGPLAN Not.* 40, 10 (Oct. 2005), 59–76. <https://doi.org/10.1145/1103845.1094817>
- [13] Virginia Vassilevska Williams and R. Ryan Williams. 2018. Subcubic Equivalences Between Path, Matrix, and Triangle Problems. *J. ACM* 65, 5, Article 27 (Aug. 2018), 38 pages. <https://doi.org/10.1145/3186893>
- [14] Mihalis Yannakakis. 1990. Graph-Theoretic Methods in Database Theory. In *Proceedings of the Ninth ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems* (Nashville, Tennessee, USA) (*PODS '90*). Association for Computing Machinery, New York, NY, USA, 230–242. <https://doi.org/10.1145/298514.298576>