Complexity of context-free language (CFL) reachability problem: new challenges and opportunities

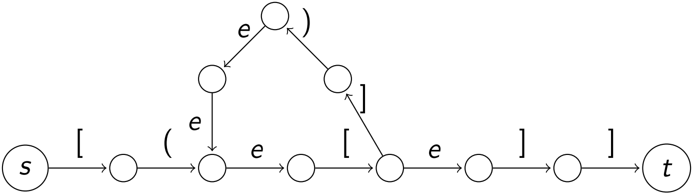
Ekaterina Shemetova

JetBrains Research, Programming Languages and Tools Lab
SPBAU, SPBSU

17.12.2021

CFL-reachability

graph reachability + constraints in the terms of context-free language



[(e[])eee[e]]

- Static code analysis: interprocedural dataflow analysis, points-to, ..., (equivalence to the subclass of set constraints)
- Datalog chain queries
- Graph databases

Motivation

There is no algorithm for the CFL-reachability problem faster than $O(n^3)$ ($O(n^3/\log n)$).

Nevin Heintze and David McAllester, 1997

On the Cubic Bottleneck in Subtyping and Flow Analysis

- It is an open problem whether truly subcubic $O(n^{3-\varepsilon})$ algorithm exists
- Under the **combinatorial BMM-hypothesis** such an algorithm is unlikely to exist...
- Moreover, CFL-reachability problem is inherently sequential (is hard to parallelize effectively)

Plan

While it is hard to obtain a good solution in the general case, we can go deeper into:

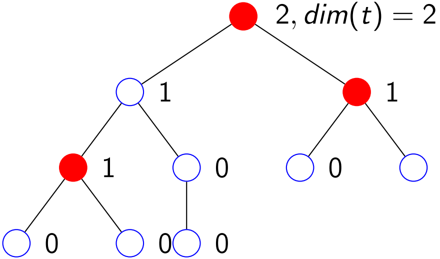
- Parametrized complexity: fix some parameter (specific language or graph) and try to improve this variant of the problem
- Fine-grained complexity: no improvement, just better understanding why problem is hard
- Dynamic complexity: how fast can we update the output when the input is slightly changed

Parametrized complexity: languages of bounded dimension

Advantages and disadvantages of Linear Datalog

👍 Linear Datalog=Linear languages can be evaluated effectively
👍 Linear Datalog=Linear languages is effectively parallelizable
👎 Linear Datalog=Linear languages is not expressive enough (i.e. for static code analysis)

In order to increase expressivity, we considered the **languages of bounded dimension** — generalization of linear languages



Parametrized complexity: languages of bounded dimension

Our results:

- CFL-reachability where input is the language of bounded dimension can be effectively evaluated and parallelized
- The rational index (complexity measure of language in FL theory) of the languages of bounded dimension d is no more than $O(n^{2d})$
- Will be submitted to DLT/SCR/journal

Future work:

- Generalize our techniques to estimate the complexity of the reachability in terms of MCFG (multiple CFG)
- Languages of bounded dimension can be more precise and still effective approximation of analysis in compare with the linear ones — estimate what variants of CFL-reachability based static code/data analyses can be effectively approximated in such a way

Dynamic complexity: motivation

Dynamic complexity

How fast can we update the output when the input is slightly changed?

Why we are interested in dynamic complexity of the CFL-reachability problem?

- CFL-reachability is a dynamic problem
- When code changes, how fast can we update the result of the CFL-reachability based code analysis?
- New algorithmic approaches: expressing an updating in first-order logic, usage of the regular Datalog for updating, ...
- Recently we have obtained reduction from the CFL-reachability problem to incremental transitive closure

Grants and students

- Participating in research project “Logical and algebraic methods in formal language theory” (under supervision of Alexander Okhotin) funded by the Russian Scientific Foundation
- Students:
    - Alexandra Olemskaya (HSE, now SPBSU) — had her bachelor degree in HSE with thesis “On some special cases of the CFL-reachability problem”
    - Alexandra Istomina (SPBSU) — working on her master thesis “Fine-grained reductions around the CFL-reachability problem”
- Teaching: “Dynamic graph algorithms” block (Spring 2021) in the “Graph theory” course (SPSU)

Current & Future work

- Dynamic complexity of the CFL-reachability in static and dynamic setting
- Fine-grained complexity: add to the current reduction's zoo some new reductions
- Generalization of obtained results on the CFL-reachability to more complicated language-constraint reachability problems (MCFG, Interleaved Dyck, ...)