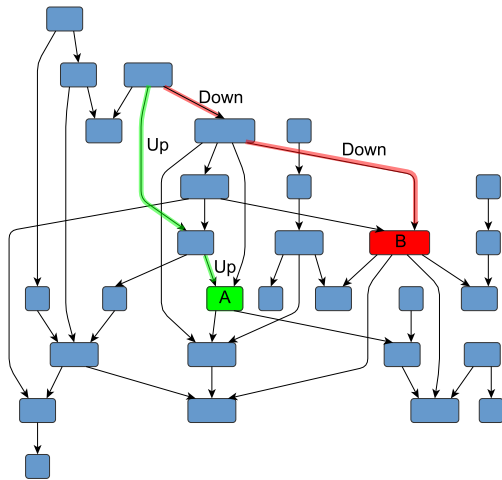# Multiple-Source Context-Free Path Querying in Terms of Linear Algebra

Arseniy Terekhov, Vlada Pogozhelskaya, Vadim Abzalov, Timur Zinnatulin, **Semyon Grigorev**

JetBrains Research, Programming Languages and Tools Lab
Saint Petersburg University

March 24, 2021

Navigation through a graph

- Are nodes A and B on the same level of hierarchy?
- Is there a path of form $\mathbf{Up}^n \mathbf{Down}^n$?
- Find all paths of form $\mathbf{Up}^n \mathbf{Down}^n$ which start from the node A

Context-Free Path Querying (CFPQ)

- Applications
  - Static code analysis
  - Graph segmentation
- Theory
  - !!!
  - !!!

Context-Free Path Querying (CFPQ)

- Applications
  - Static code analysis
  - Graph segmentation
- Theory
  - !!!
  - !!!
- Integration with real-world systems
  - !!!!
  - Kuijpers for Neo4j: too slow to be practical

Context-Free Path Querying (CFPQ)

- Applications
  - Static code analysis
  - Graph segmentation
- Theory
  - !!!
  - !!!
- Integration with real-world systems
  - !!!!
  - Kuijpers for Neo4j: too slow to be practical
  - RedisGraph

Linear Algebra Based CFPQ Algorithm

- Definition
  - ► Special DSL which can be specialized and compiled
  - ► Ahead-of-time specialization
- Impractical memory consumption
  - ► Naïve multiple substring matching
  - ► 2D convolution
- Context-Free grammars are too hard to be used by end-users
  - ► **GTX-1070**: Pascal architecture, 8GB GDDR5, 1920 CUDA cores
  - ► **Tesla T4**: Turing architecture, 16GB GDDR6, 2560 CUDA cores

Proposed Solution

- RedisGraph
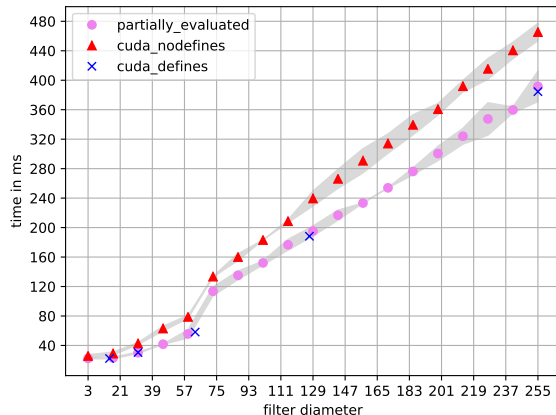- Cypher[1]
- Multiple-Soource CFPQ
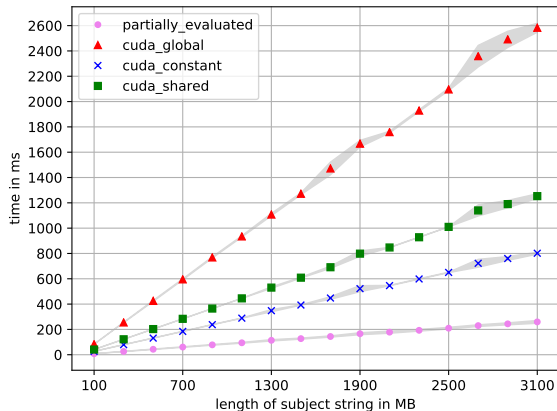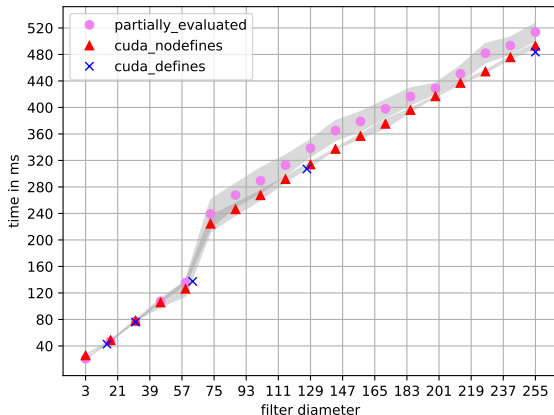

Results for GTX-1070
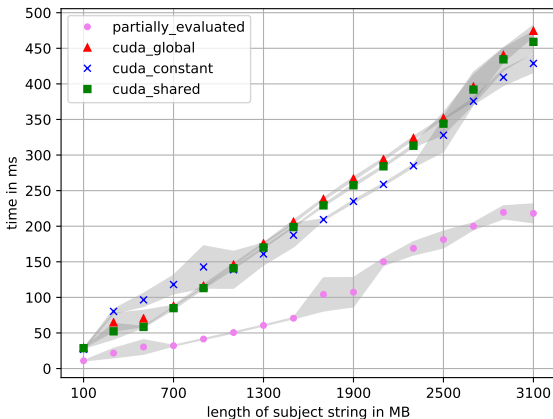
Results for Tesla T4

Multiple-Source CFPQ

- Application: image processing
- Subject image: random image of size 1GB
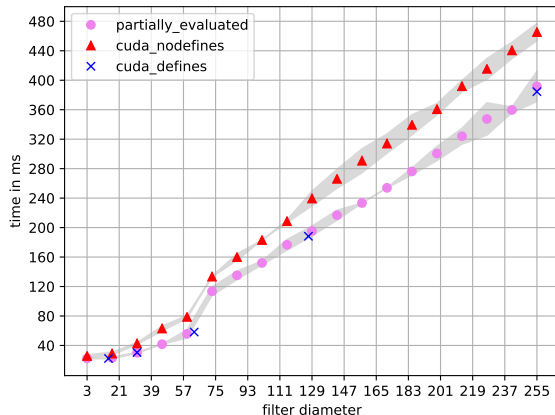- Filters: random square filters with diameter 3 to 255

Results for GTX-1070

Results for Tesla T4

Implementation Details

- Application: image processing
- Subject image: random image of size 1GB
- Filters: random square filters with diameter 3 to 255
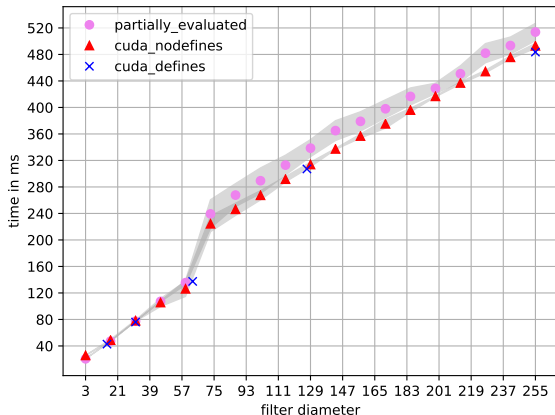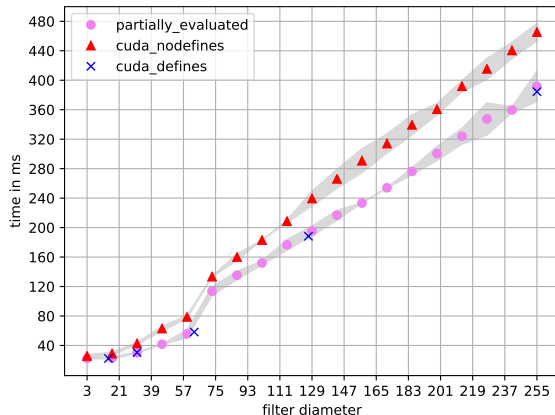

Results for GTX-1070

Results for Tesla T4

Cypher Extension

- Application: image processing
- Subject image: random image of size 1GB
- Filters: random square filters with diameter 3 to 255

Results for GTX-1070
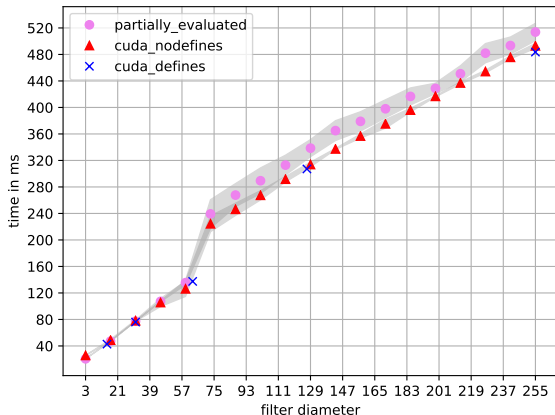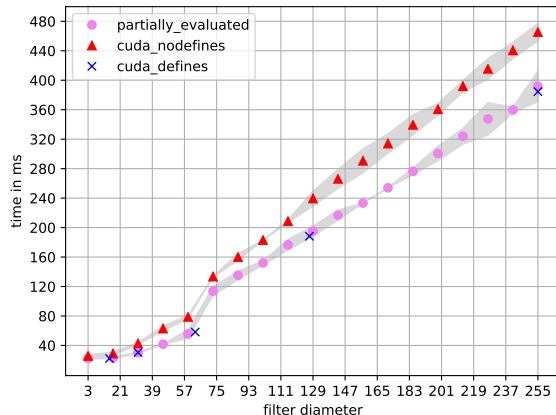
Results for Tesla T4

Queries Examples

- Application: image processing
- Subject image: random image of size 1GB
- Filters: random square filters with diameter 3 to 255

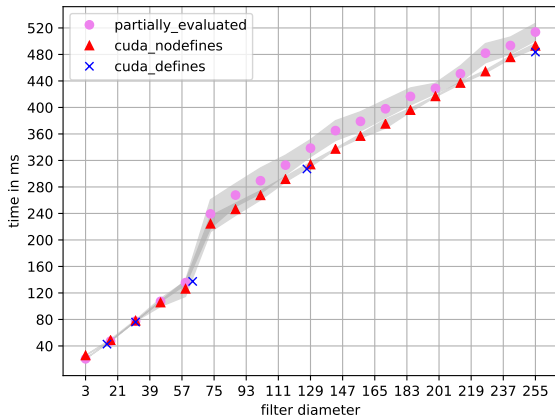Results for GTX-1070

Results for Tesla T4

Evaluation Setup

- Ubuntu 18.04, Intel Core i7-6700 CPU, 3.4GHz, DDR4 64Gb RAM
- Graphs stored in RedisGraph with our extensions
- Queries are generated with template for given size of start set
- The union of all start sets is a V

Graph	#V	#E	Q
core	1323	4342	g_1
pathways	6238	18 598	g_1
gohierarchy	45 007	980 218	g_1
enzyme	48 815	109 695	g_1
eclass_514en	239 111	523 727	g_1
geospecies	450 609	2 311 461	geo
go	272 770	534 311	g_1

```
PATH PATTERN S =
  ()-/ [<:SubClassOf [~S | ()] :SubClassOf] | [<:Type [~S | ()] :Type] /->()
MATCH (src)-/ ~S /->()
WHERE {id_from} <= src.id and src.id <= {id_to}
RETURN count(*)
```

eclass_514en

Summary
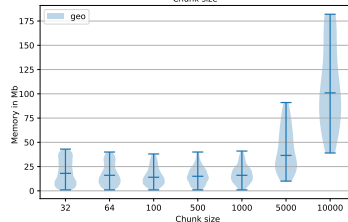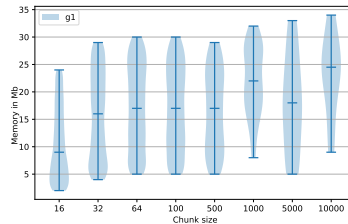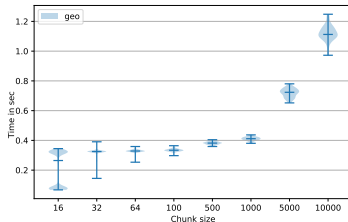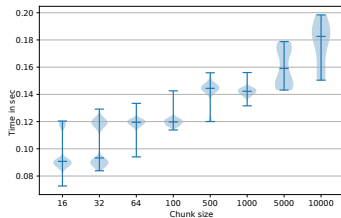
- Full-stack support for CFPQ in real-world graph query languages on the top of real-world graph database
  - No more ugly context-free grammars
  - No more custom graph formats and storages
- Reasonable performance of context-free path queries
  - Multiple-source scenario
  - Space-time ratio can be tuned
- Context-free path queries can be used in applications with well-established tools

Future Research

- Cypher semantics mechanization in Coq
  - For new extension
  - Correctness of translation to linear algebra
- Tensor-based CFPQ algorithm
  - All paths
  - Multiple-source
- Evaluation
  - More data
  - More algorithms

Contact Information

- Semyon Grigorev:
  - s.v.grigoriev@spbu.ru
  - Semen.Grigorev@jetbrains.com
- Arseniy Terekhov: simpletondl@yandex.ru
- Vlada Pogozhelskaya: pogozhelskaya@gmail.com
- Vadim Abzalov: vadim.i.abzalov@gmail.com
- Timur Zinnatulin: teemychteemych@gmail.com

- Try it out (Docker image with all included): !!!
- Sources of RedisGraph extended with CFPQ: !!!
- Sources of Cypher parser extended with path patterns: !!!

Thanks!