Overview and Analysis of GPU Acceleration for Regular Expressions

Engineering Methods 2023/2024

Roman Gajdoš

Institute of Informatics, Information Systems and Software Engineering
Faculty of Informatics and Information Technologies
Slovak University of Technology in Bratislava

27. november 2023

GPU's or something else?

- Can we use GPU's to accelerate regular expressions?
- What are the alternatives?
- What are the pros and cons of each?
- What are the current state of the art measurements?
- What are the current limitations?
- What are the future directions?

Outline

Background

- Regular expressions
  - Finite state automata
  - DFA and NFA
- GPU
  - Large memory bandwidth and parallelism
  - CUDA and OpenCL

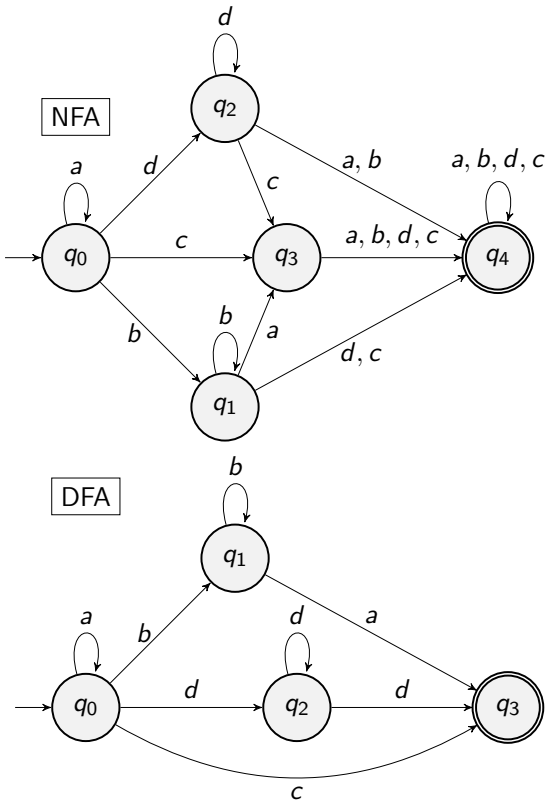

Figure: FSM diagram of `a*(b+a|d*c)` regular expression

Related work

Study	Year
FSA measurements	
GPU Acceleration of Regular Expression [6]	2013
Acceleration platforms	
Demystifying Automata Processing [3]	2017
Papers on speeding up FSA processing on GPUs	
DFA	
On-the-Fly Principled Speculation [7]	2015
Scaling Out Speculative Execution [5]	2020
GSpecPal: Speculation-Centric FSM Parallelization [4]	2022
NFA	
Why GPUs are Slow at Executing NFAs [1]	2020
Asynchronous Automata Processing on GPUs [2]	2023

Table: Table of Papers on Regular Expression Matching

Studies findings: FSA

Dataset	U-DFA	C-DFA	E-DFA
Backdoor	4.56	0.37	1.13
Spyware	3.63	0.25	0.69
E-M	8.63	8.63	8.63
Range.5	8.51	8.51	8.51
Range1	8.24	8.24	8.24
Dotstar.05	7.39	0.53	1.44
Dotstar.1	6.08	0.34	1.16
Dotstar.2	X	0.26	1.10

Table: Speedup of DFA's formats over NFA, data from [6]

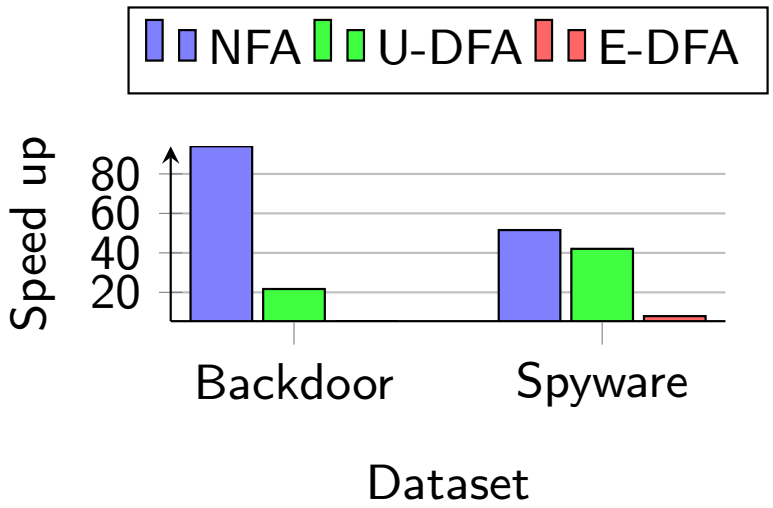

Figure: Speed up of GPU accelerated traversals over CPU, data from [6]

Studies findings: Acceleration platforms

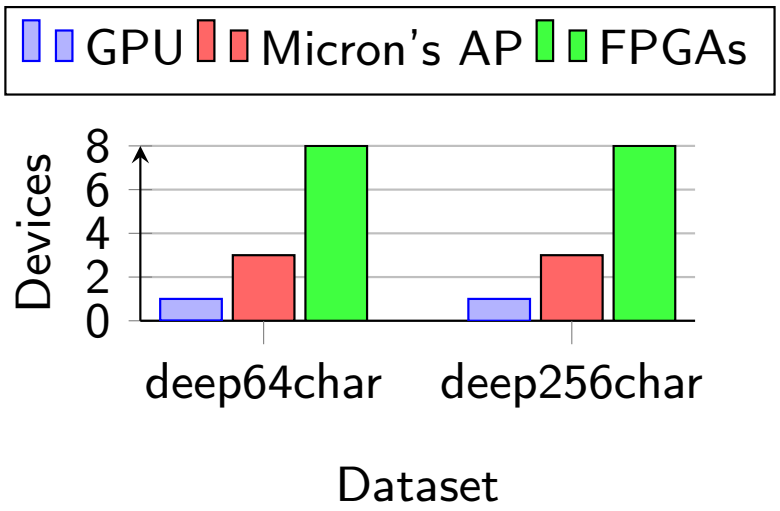

Figure: Number of devices, part of figure 5 from [3]

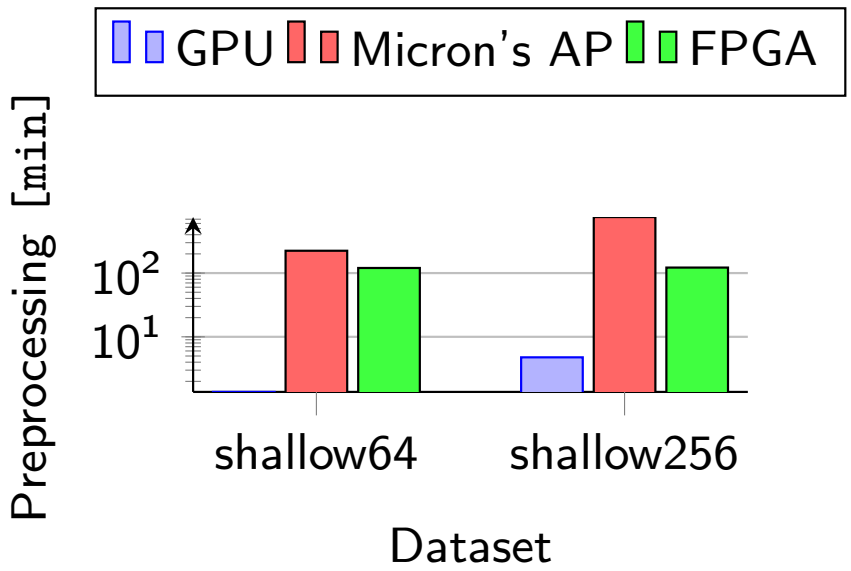

Figure: Preprocessing time, part of figure 5 from [3]

Conclusion and future work

- Conclusion
  - GPU acceleration of regular expressions is possible and viable option
  - DFA faster but larger memory footprint
  - NFA slower but smaller memory footprint
  - FPGA and AP are more efficient, but less flexible
- Future work
  - Up to date measurements
  - Heteregeneous acceleration
  - CUDA and OpenCL

Bibliography

Hongyuan Liu, Sreepathi Pai, and Adwait Jog.
Why gpus are slow at executing nfas and how to make them faster.
In *Proceedings of the Twenty-Fifth International Conference on Architectural Support for Programming Languages and Operating Systems*, ASPLOS '20, page 251–265, New York, NY, USA, 2020. Association for Computing Machinery.

Hongyuan Liu, Sreepathi Pai, and Adwait Jog.
Asynchronous automata processing on gpus.
Proc. ACM Meas. Anal. Comput. Syst., 7(1), mar 2023.

Marziyeh Nourian, Xiang Wang, Xiaodong Yu, Wu-chun Feng, and Michela Becchi.
Demystifying automata processing: Gpus, fpgas or micron's ap?
In *Proceedings of the International Conference on Supercomputing*, ICS '17, New York, NY, USA, 2017. Association for Computing Machinery.

Yuguang Wang, Robbie Watling, Junqiao Qiu, and Zhenlin Wang.
Gspecpal: Speculation-centric finite state machine parallelization on gpus.
In *2022 IEEE International Parallel and Distributed Processing Symposium (IPDPS)*, pages 481–491. IEEE, 2022.

Yang Xia, Peng Jiang, and Gagan Agrawal.
Scaling out speculative execution of finite-state machines with parallel merge.
In *Proceedings of the 25th ACM SIGPLAN Symposium on Principles and Practice of Parallel Programming*, PPoPP '20, page 160–172, New York, NY, USA, 2020. Association for Computing Machinery.

Xiaodong Yu and Michela Becchi.
Gpu acceleration of regular expression matching for large datasets: Exploring the implementation space.
In *Proceedings of the ACM International Conference on Computing Frontiers*, CF '13, New York, NY, USA, 2013. Association for Computing Machinery.

Zhijia Zhao and Xipeng Shen.
On-the-fly principled speculation for fsm parallelization.
ACM SIGPLAN Notices, 50(4):619–630, 2015.