Introduction to Graduate Study in CSE

CSE 591
Lecture 3
Prof. Patrick Crowley
Plan for Today

• Questions
• Finish Wednesday’s discussion
• Today’s discussion
  – Anatomy of a CS conference paper
  – Introduction to Rotations
• Next lecture
  – Bill Smart, Robotics
Today’s Discussion

• See “Guide to Finding CSE Literature”
  – On-line at the CSE 591 web page

• Anatomy of a CS conference paper
Abstract

Regular expression matching is a crucial task in several networking applications. Current implementations are based on one of two types of finite state machines. Non-deterministic finite automata (NFAs) have minimal storage demand but have high memory bandwidth requirements. Deterministic finite automata (DFAs) exhibit low and deterministic memory bandwidth requirements at the cost of increased memory space. It has already been shown how the presence of wildcards and repetitions of large character classes can render DFAs and NFAs impractical. Additionally, recent security-oriented rule-sets include patterns with advanced features, namely back-references, which add to the expressive power of traditional regular expressions and cannot therefore be supported through classical finite automata.

In this work, we propose and evaluate an extended finite automata designed to address these shortcomings. First, the automaton provides an alternative approach to handle character repetitions that limits memory space and bandwidth requirements. Second, it supports back-references without the need for backtracking in the input string. In our discussion of this proposal, we address practical implementation issues and evaluate the automaton on real-world rule-sets. To our knowledge, this is the first high-speed automata that can accommodate all Perl-compatible regular expressions present in the Snort network intrusion and detection system.

1. Introduction

Finding patterns of interest within large datasets is a central task in many applications and has been a well-studied area.
Where was it published?
Introduction

• Background
  – what is the problem?
  – why is it important?
  – to whom is it important? CS people? Specific non-CS people?

• Related work (may also be later)
  – who has worked on this stuff before?
  – what is novel about the present work?
  – does the work have any direct competitors?

• Statement of contribution
  – Novelty
  – Method
  – result summary
Methods/Proposal/Design

- Starts with fundamental definitions
- How they did what they did
- Does the paper describe new methods, or does it explore/integrate/compare existing ones?
- How much is explained vs elided to save space?
- Are there formal claims made? Are they proven?
- Is there supplementary information online? E.g., SIGGRAPH papers always have videos
Results/Evaluation

- High-level methods (for those who skip methods sections)
- Validation
  - what are the standards for this discipline?
    - does it have to be proven correct?
    - does it have to have an analytical performance estimate/model?
    - does it have to be coded up?
    - does it have to be put into hardware?
    - how thoroughly should it be tested?
      - on simulated data? how realistic?
      - on real data? how much/what kind?
    - must there be user studies?
  - what metrics must be reported?
  - are there be error bars? why or why not?
  - is there a comparison to competing tools?
Discussion/Conclusion and Future Work

• Why is the world better off for knowing the results of this paper?
  – new methods
  – new explanations
  – comparison among existing things
  – new empirical evidence of something

• do they make qualitative statements about their work?
  – "This method results in fewer dropped packets under realistic loads“
  – "This method predicts significantly more regulatory motifs than...“
  – "We have developed a new algorithm to substantially increase the accuracy
    of branch prediction in out-of-order, superscalar processors“
  – If so, what is the quantitative or formal support for these statements? Do
    you believe it?

• What was left to future work? Is the work self-contained? Are there
  "obvious" next steps?
operating semantics while retaining efficient support for repeated substrings and counting constraints. Also, we showed how standard compression techniques can be applied to an extended-DFA. As a practical consideration, we analyzed the problem of compiling several regular expressions with problematic sub-patterns into a single automaton, and proposed a hybrid, comprehensive solution.

To the best of our knowledge, we have described the first high-speed automaton that can accommodate all the Perl-Compatible Regular Expressions present in the Snort network intrusion and detection system.

ACKNOWLEDGMENTS
The authors wish to thank Chris Clark from Computer Resources, Inc. for his useful feedback. This work has been supported by National Science Foundation grants CCF-0430012 and CCF-0427794 and by Intel’s gifts.

REFERENCES
**Review 1**


**Contributions (What are the major issues addressed in the paper? Do you consider them important? Comment on the degree of novelty, creativity and technical depth in the paper.)**

The paper presents a latency study which compares PCI-X and PCIe buses. The main result of the paper is that PCIe's latency is substantially higher than that of PCI-X. On the basis of this observation, the authors argue for low-latency protocol designs (such as those that do not force "store-and-forward" style of switching) for the future I/O bus architectures (and standards).

**Strengths (What are the major reasons to accept the paper? [Be brief.])**

- The measurement methodology is interesting and has been validated.
- The study itself seems fairly complete and includes a good explanations for most of the measured numbers.
- While the basic observation that PCIe has worse latency than PCI-X is not that surprising (probably in hindsight), it's quite interesting to see how worse it actually is. And the implications for the protocol designs for the next generation I/O bus can be significant.
Weakenesses (What are the most important reasons NOT to accept the paper? [Be brief.])

- The paper doesn't really propose any new or original solutions. It merely observes that latency suffered in PCI-E and that may be an obvious thing for many in the field. This paper seems like a collection of interesting tidbits that together lead to nothing particularly new.

- While the methodology is adequate, some of the setup seems to favor PCI-X (see "Experiments" below)

- This reads like an expanded motivation section for a paper on more forward-looking research such as photonics.

Detailed comments (Please provide detailed comments that will be helpful to the TPC for assessing the paper, as well as feedback to the authors.)

- I think the overall theme of investigating I/O bus designs and latency problems is interesting and probably important, but this paper seems to be somewhat not ready for prime-time. The main contribution seems to be the conclusion that PCI-E have higher latency than PCI-X and newer non store-and-forward protocols are required, and that does not seem too exciting.

- Bulk of the paper focuses on the latencies of PCI-E ** links ** themselves. And I can argue that the actual numbers are not as important because, unlike latencies within the PCI-E core, technology CAN help link latencies. PCI-E 3.0 might cut an 8x latency