Fault Site Pruning for Practical Reliability Analysis of GPGPU Applications

Bin Nie, Lishan Yang, Adwait Jog, Evgenia Smirni
College of William & Mary
Email: {bnie, lyang11, esmirni}@cs.wm.edu, ajog@wm.edu

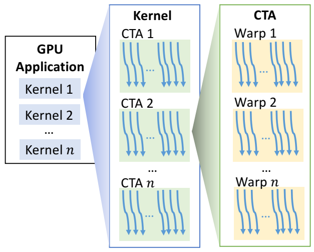
Motivation

- Challenge in GPU reliability research: huge unreachable exhaustive fault sites for fault injection
  - Benchmark GEMM:

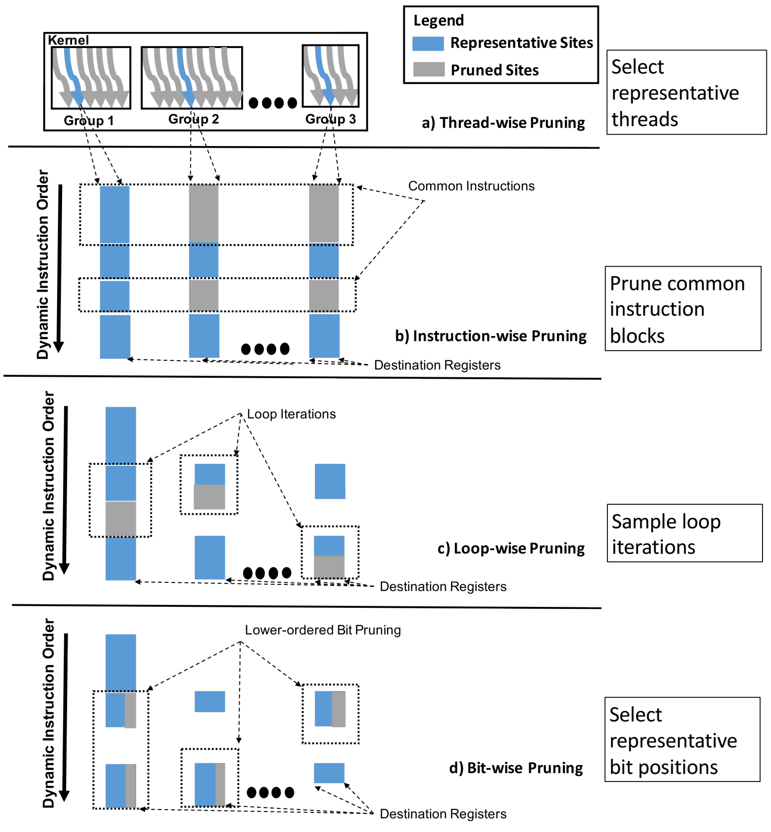
  16384 threads → 6.23×10^8 fault sites!
- Baseline solution: Random sampling based on statistics

  Confidence Interval:99.8% } 60K fault sites
  Error Margin: 1.26%
- Our goal: accurate & effective fault injection methodology

Background



Progressive Error Sites Pruning



a) Thread-wise Pruning

Select representative threads

b) Instruction-wise Pruning

Prune common instruction blocks

c) Loop-wise Pruning

Sample loop iterations

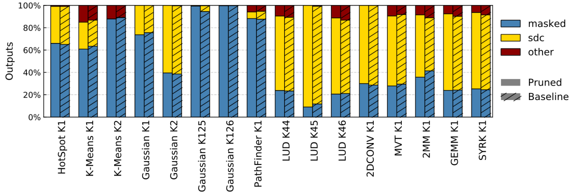
d) Bit-wise Pruning

Select representative bit positions

Evaluation

Accuracy

Our pruning method gives excellent error resilience estimations for most of benchmark kernels. On average, the differences between our pruning technique and baseline regarding masked, SDC (Silent Data Corruption), other outputs are 1.68%, 1.90%, and 1.04%, respectively.



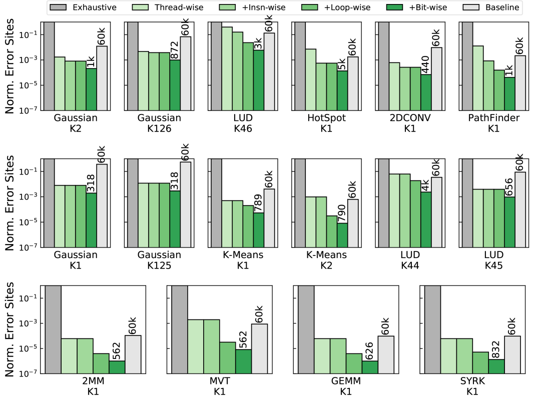
❖Baseline: 60K random experiments, with 99.8% confidence intervals and 1.26% error margin.

Effectiveness

✓ Thread-wise pruning is most effective, as it reduces the magnitude of the number of error steps by up to 5 orders of magnitude.

✓ Instruction-wise pruning is most effective for HotSpot and PathFinder, because these are a lot of threads left after thread-wise pruning.

✓ Loop-wise pruning and bit-wise pruning progressively contributes to the reduction of the error sites for each benchmark kernel.



❖“+” indicates that each pruning technique is progressively built upon the pruned sites delivered by the previous one.

❖The number of pruned fault sites is normalized by the original exhaustive error sites for each benchmark kernel.

❖We use log scale with a base of 10 for the y-axis.

❖Only kernels in the first row are applicable to instruction-wise pruning.

Reference

Nie, Bin, Lishan Yang, Adwait Jog, and Evgenia Smirni. "Fault Site Pruning for Practical Reliability Analysis of GPGPU Applications.“ In *Proceedings of the 51th Annual IEEE/ACM International Symposium on Microarchitecture*. ACM, 2018.

Conclusion

- GPGPU applications have huge unreachable exhaustive fault sites
- We propose our progressive fault site pruning methodology leveraging GPGPU-specific features.
- Our pruning technique gets accurate GPU reliability assessment and achieves significant reduction in the number of fault injection experiments.

Acknowledgement

This material is based upon work supported by the National Science Foundation (NSF) grants (#1717532 and #1750667) and a summer grant from the College of William and Mary. This work was performed in part using computing facilities at the College of William and Mary which were provided by contributions from the NSF, the Commonwealth of Virginia Equipment Trust Fund and the Office of Naval Research.