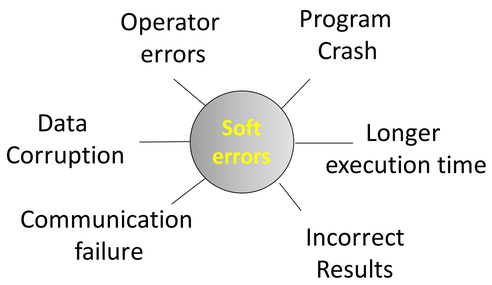
Understanding Ineffectiveness of the Application-Level Fault Injection

Luanzheng Guo, Jing Liang, Dong Li
Parallel Architecture, System, and Algorithm Lab@Department of EECS
University of California, Merced

Motivation and Introduction

- Soft errors are a big threat to extreme-scale applications
- The application-level fault injection is one of the most common methods to study the application resilience to soft errors [2].

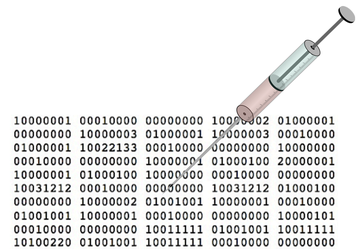
- **What is the application-level fault injection?**
  - Artifcial faults are randomly injected into application variables or computation logic
  - A large amount of random fault injection tests



Research questions:

- The application-level fault injection has some inherent uncertainties due to its random nature
  - What is the correlation between the fault injection result and the number of fault injection tests?
  - Does the application vulnerability vary with the input problem?

Those uncertainties can make fault injection ineffective for accurately modeling application vulnerability.



Methodology

- Use a LLVM-based fault injection tool [1]
  - Randomly select an instruction and and then randomly flip a bit in the output operand.
  - We inject faults into the critical functions of the benchmarks
  - Always single-bit flip

- We inject faults into the critical functions of NAS parallel benchmarks

Benchmark	Critical functions	Input Problem Size
CG	conj_grad	S, W, A, B, C
MG	mg3P	S, W, A, B, C
FT	fftXYZ	S, W, A, B, C
BT	x_solve	S, W, A, B, C
SP	x_solve	S, W, A, B, C
LU	ssor	S, W, A, B, C
IS	randlc	S, W, A, B, C

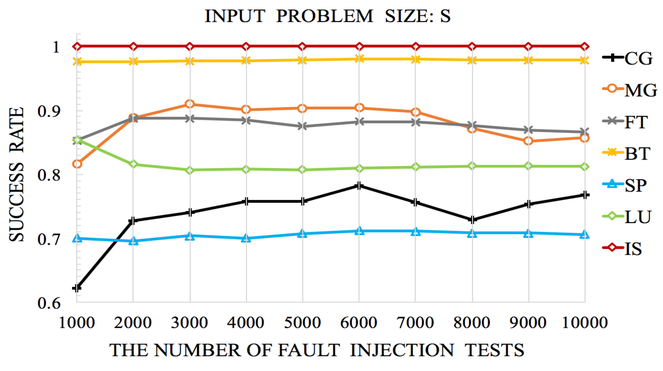
- 55,000 fault injection tests in total per benchmark

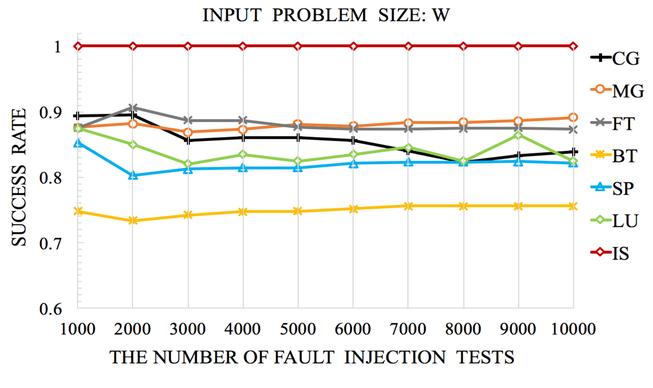
References

[1] J. Calhoun, L. Olson, and M. Snir. Flipit: An LLVM based fault injector for HPC. In Euro-Par Parallel Processing Workshops , 2014.

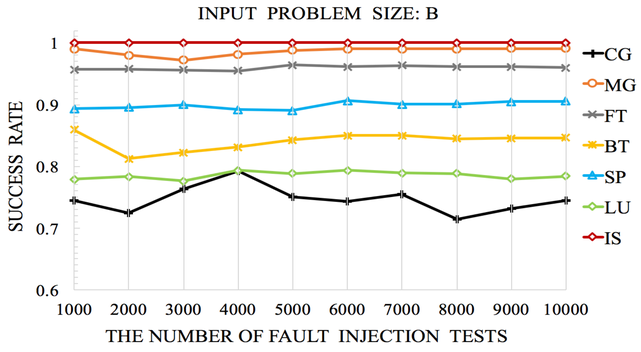
[2] Dong Li, Jeffrey S. Vetter, and Weikuan Yu. "Classifying Soft Error Vulnerabilities in Extreme-Scale Scientific Applications Using a Binary Instrumentation Tool". In 24th ACM/IEEE International Conference for High Performance Computing, Networking, Storage and Analysis (**SC**), 2012

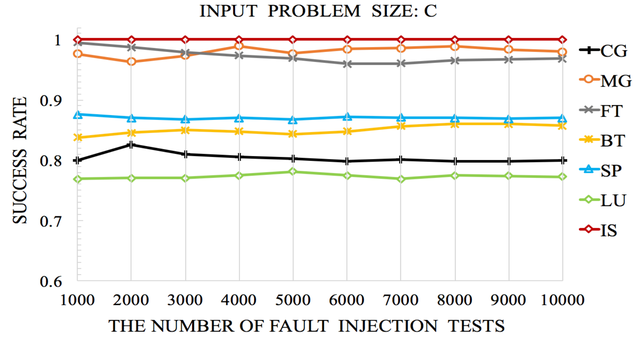
Evaluation Results

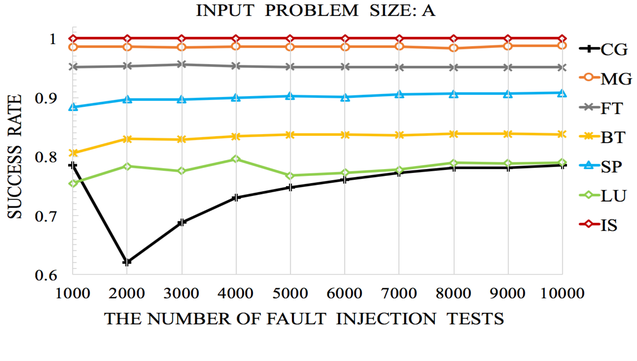




- **Conclusions**
  - The fault injection results of some applications are sensitive to the number of fault injection tests, while the fault injection results of some applications are not.
  - The application resilience depends on the input problem size







Observation 1:
For BT and SP, the fault injection results vary across input problem sizes. There is more than 25% variance.

Observation 2:
The fault injection result for CG and MG vary across different numbers of fault injection tests, while the fault injection results for other benchmarks are relatively stable.

Observation 3:
Based on the results of 1000 fault injection tests, we find CG is more reliable than MG. But based on the results of 2000, we make an opposite conclusion.