LSU Louisiana State University, PI: Lu Peng, lpeng@lsu.edu

CNS-1527318

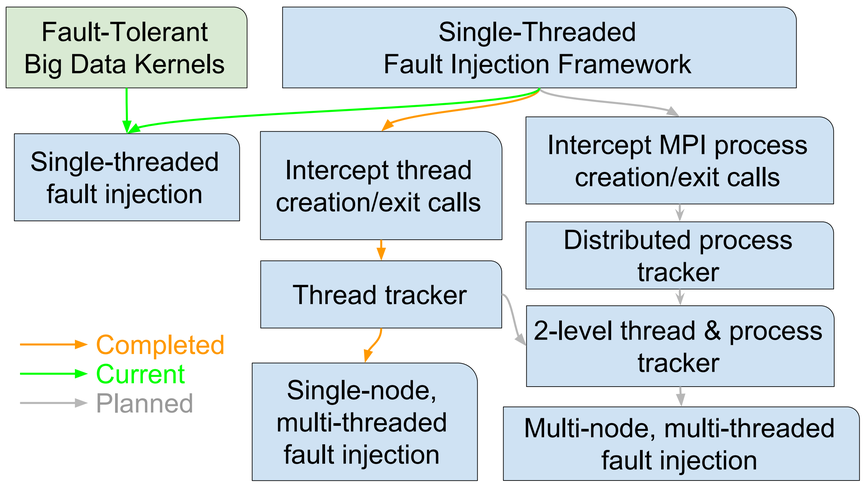
Collaborative Research: Comprehensive Algorithmic Resilience (CAR) for Big Data Analytics

Motivation

Big data analytics may be vulnerable to soft errors

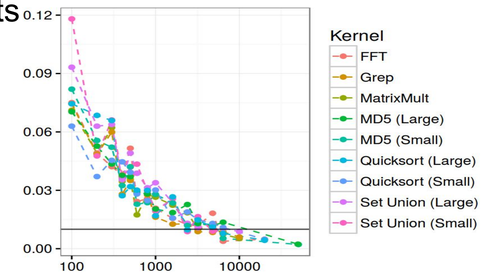
- Understand possible impacts due to soft errors
- Seek error resilience with two approaches
  - Low- and high-level algorithmic checks *(this project)*
  - Adaptive checkpointing *(collaborative project CNS-1527051)*

Overall Roadmap



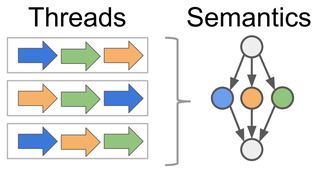
Single-Threaded Experiment Procedure

- Program is instrumented, allowing run-time bit-flips
  - Original & Fault-tolerant source code
- 1 bit flip error per program run
  - Select a set of dynamic instructions, uniform distribution
  - Inject error in each run and run to end
- Outcomes are categorized
- Output errors are measured with kernel-specific metric
- Determining number of experiments for statistically-sound results
  - Using a regression tree that predicts outcome as a proxy
  - Increase # experiments until standard deviation of accuracy falls below threshold



Issues for Multithreaded Programs

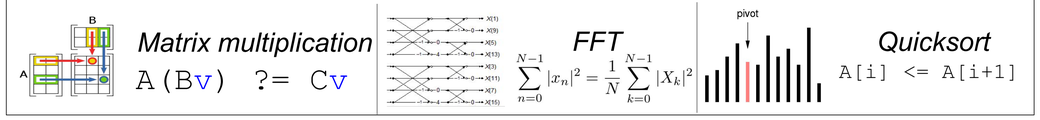
- Match thread execution to program semantics
  - Handle reused thread IDs
  - Handle nondeterministic thread order
- Fault site includes thread ID



Big Data Kernels and Invariant Enforcement

- 8 Kernels chosen from BigDataBench
- Kernels have high-level and/or low-level invariants
- When invariant is violated, redo part of the computation

High-level invariants



Low-level invariants

```
while (left < right) ...
if (left < right) ...
else ...
```

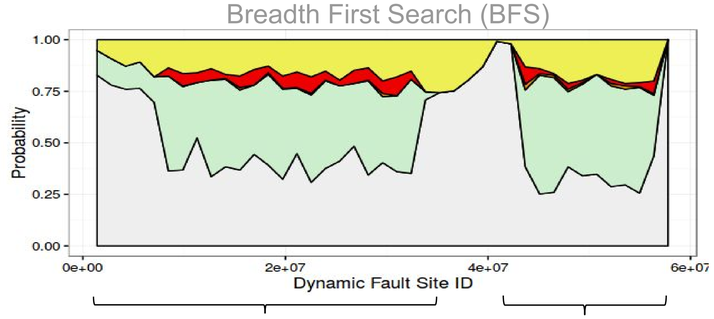
Conditions should hold in conditional statements

```
A[i] = B[i]
a = a + 1
b = foo()
```

Assigned values should match expectation

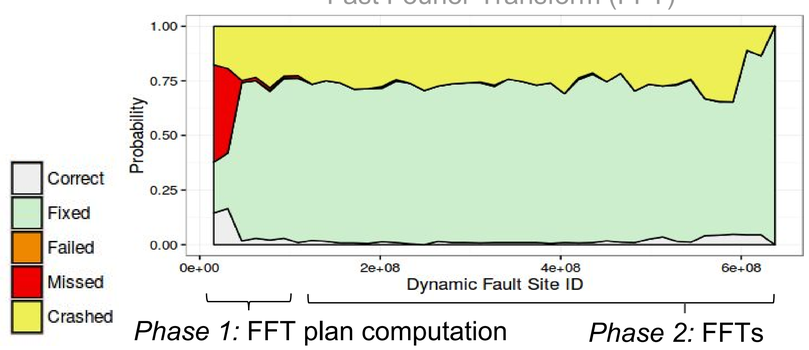
Details -- Phase Behavior

Breadth First Search (BFS)



Phase 1: Input preproccessing *Phase 2:* Iterations

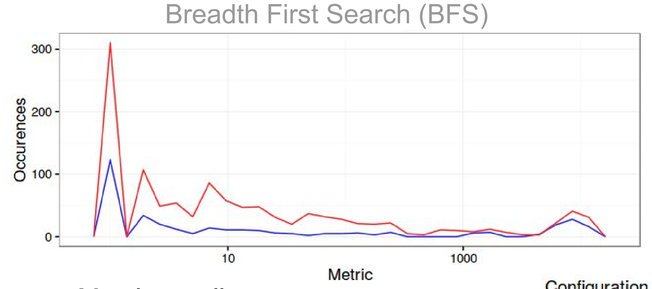
Fast Fourier Transform (FFT)



Phase 1: FFT plan computation *Phase 2:* FFTs

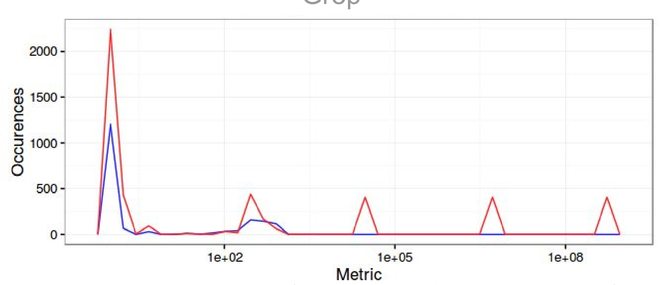
Details -- Output Error Magnitude

Breadth First Search (BFS)



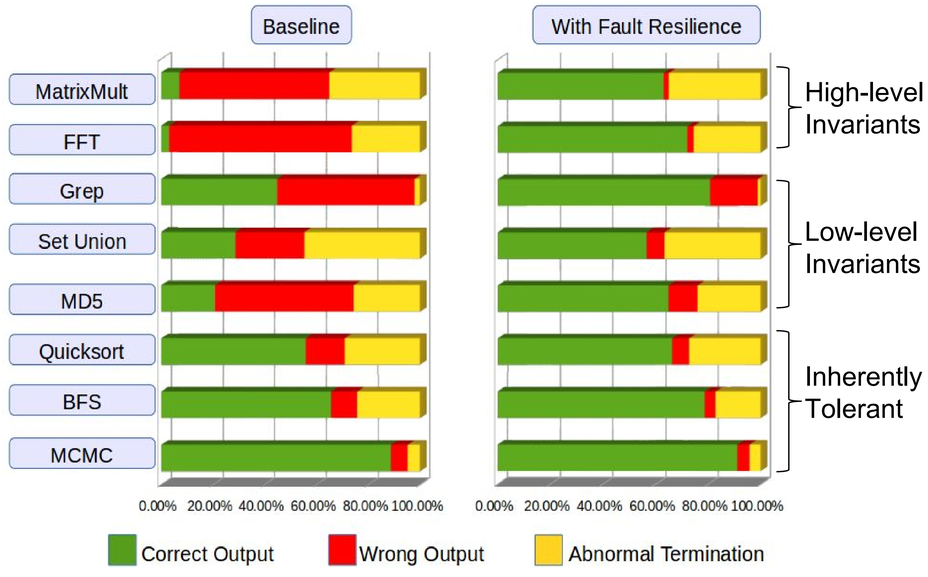
- Mostly small errors
- Inherently fault resilient

Grep

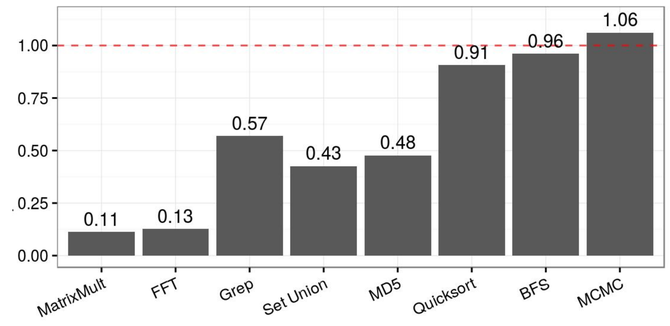


- Has large errors (due to injection in high bits)
- Error checkers effective in removing large errors

Overall Fault Resilience Improvement



Improved Expected Fault-Free Run Time



- Fault resilience overhead justified by faults corrected = shorter run time for a fault-free run in a faulty environment
- Focus on kernels that are not inherently fault-tolerant