The Effect of Performance Optimizations on Soft Error Reliability for Machine Learning Applications

Eda Nur Azın, Işıl Öz

Izmir Institute of Technology

January 22, 2020

Outline

Outline

GPUs: Accelerators for Machine Learning

- Developing ML applications starts with training models with large datasets
- GPUs offer throughput-optimized fast parallel processing
- Architectures powered by tensor cores provide faster training and greater performance (especially for neural networks)

GPU Reliability

- With many execution cores and complex memory structures, GPUs show high vulnerability to soft errors
- GPGPU applications do not tolerate errors like graphics applications

Performance/Reliability Tradeoff

- With many execution cores and low-latency memory structures, GPUs offer different performance optimizations
- How do the performance optimizations affect the reliability of the applications?

Outline

CUDA Performance Optimization Principles [1]

- Leverage zero-overhead thread scheduling to hide memory latency
- Optimize use of on-chip memory to reduce bandwidth usage and redundant execution
- Group threads to avoid SIMD penalties and memory port/bank conflicts
- Threads within a thread block can communicate via synchronization, but there is no built-in global communication mechanism for all threads

[1] *"Optimization Principles and Application Performance Evaluation of a Multithreaded GPU Using CUDA"*, S. Ryoo et al., *PPoPP*, 2008.

Outline

Reliability Evaluation on GPUs

- Fault injection experiments
- SASSIFI [2]
  - Does not work for our target platform since NVIDIA has removed the support in the modern architectures (after Maxwell)
- GPU-Qin [3]
  - Too slow due to assembly-level fault injection
- CAROL-FI [4]
  - Software-level open source fault injector tool based on GDB

[2] *"SASSIFI: An architecture-level fault injection tool for GPU application resilience evaluation"*, Hari et al., *ISPASS*, 2017.

[3] *"GPU-Qin: A methodology for evaluating the error resilience of GPGPU applications"*, Fang et al., *ISPASS*, 2014.

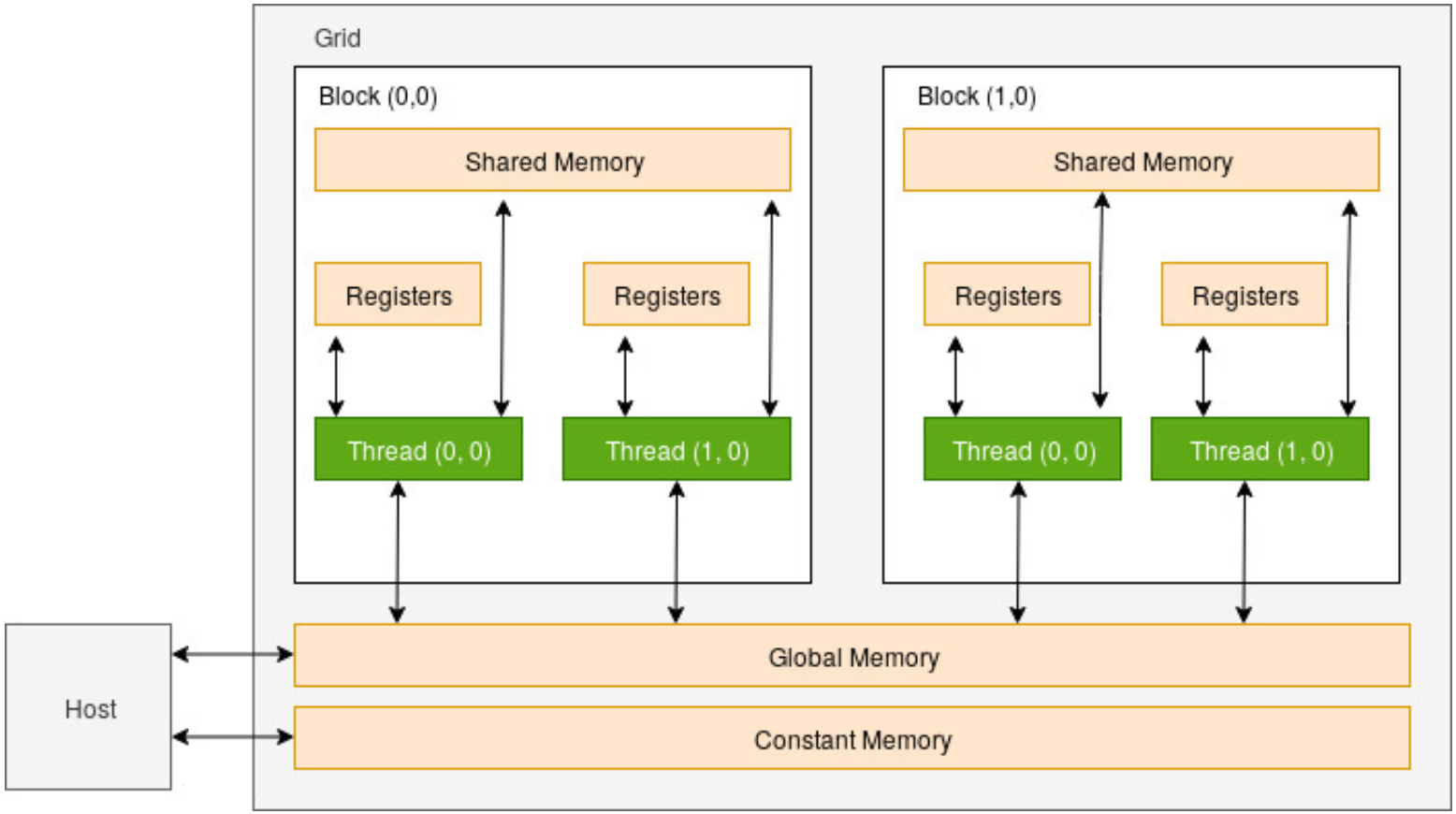
[4] *"Reliability Evaluation of Mixed-Precision Architectures"*, Fernandes dos Santos et al., *HPCA*, 2019.

Outline

K-means

- Commonly-used clustering algorithm
- Initializing the centroid points for the given cluster
- Until convergence
  - Calculate the distance from each point to each cluster centroid
  - Assign each point to the cluster which is the closest
  - Recalculate each centroid point such that the mean of all points assigned to it

CUDA Memory Model



Application Versions

Simple K-means

The assignment of the points, the computation of the distance, and the update of the centroid points are distributed into CUDA threads

K-means with Thrust Library

Thrust data structures

K-means with SM

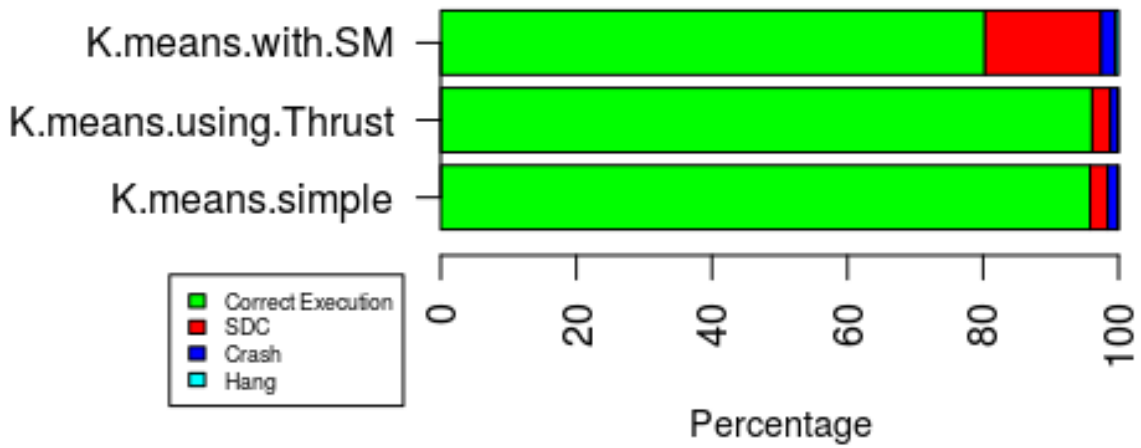
Loading data from global memory to shared memory once for all computation without separate references to the global memory for repeating accesses

Outline

Performance Results

	CPU	GPU	CPU + GPU
Simple approach	4.484s	2.777s	7.034s
Using Thrust	8.664s	2.771s	11.342s
With shared memory	4.440s	2.652s	6.916s

Fault Injection Results



Outline

Research Questions

- Do all performance optimizations impact the ML applications' soft error reliability?
- Do performance optimizations applied for ML applications impact soft error reliability for all hardware structures?
- What kind of performance optimizations do impact the ML applications' soft error reliability?
- Can we generalize the effect of the optimizations?
- Is there a tradeoff between performance and reliability for different versions of ML applications?
- What are the possible suggestions for the software developers aiming faster code to develop also more reliable software?
- What kind of fault tolerance techniques can be developed by considering fast but reliable GPU computation?

Conclusion

- Conduct more systematic and complete work for the analysis of soft error reliability for ML applications
- Extend the analysis for neural network applications

The Effect of Performance Optimizations on Soft Error Reliability for Machine Learning Applications

Eda Nur Azın, Işıl Öz

Izmir Institute of Technology

January 22, 2020