Title: Simulation-Informed Performance Tuning for Monte Carlo Proton Transport on GPUs

Author(s): Zieb, Kristofer James Ekhart

Intended for: Report

Issued: 2016-10-06

Disclaimer:
Los Alamos National Laboratory, an affirmative action/equal opportunity employer, is operated by the Los Alamos National Security, LLC for the National Nuclear Security Administration of the U.S. Department of Energy under contract DE-AC52-06NA25396. By approving this article, the publisher recognizes that the U.S. Government retains nonexclusive, royalty-free license to publish or reproduce the published form of this contribution, or to allow others to do so, for U.S. Government purposes. Los Alamos National Laboratory requests that the publisher identify this article as work performed under the auspices of the U.S. Department of Energy. Los Alamos National Laboratory strongly supports academic freedom and a researcher's right to publish; as an institution, however, the Laboratory does not endorse the viewpoint of a publication or guarantee its technical correctness.

Simulation-Informed Performance Tuning for Monte Carlo Proton Transport on GPUs

Presented at Rensselaer Polytechnic Institute

Kristofer Zieb

October 11, 2016

MCNP® Trademark

MCNP® and Monte Carlo N-Particle® are registered trademarks owned by Los Alamos National Security, LLC, manager and operator of Los Alamos National Laboratory. Any third party use of such registered marks should be properly attributed to Los Alamos National Security, LLC, including the use of the ® designation as appropriate.

- **Please note that trademarks are adjectives and should not be pluralized or used as a noun or a verb in any context for any reason.**
- **Any questions regarding licensing, proper use, and/or proper attribution of Los Alamos National Security, LLC marks should be directed to trademarks@lanl.gov.**

Overview

- Background
- Proton Transport
- GPUs
- Performance Tests
- Physics Test Problems
- Proposed Work
- Wrap-Up

- How the Work Began
- Exascale Initiative
- Call to Action
- Our Goal: The Path Forward
- Charged Particle Challenges

DOE Exascale (10^{18} FLOPS!) Initiative

- Started in 2011.
- Reach exascale by 2024.
- National labs are being equipped with the latest hardware accelerators.
- Must be forward thinking to accomplish this goal.

Los Alamos NATIONAL LABORATORY EST.1943

Argonne NATIONAL LABORATORY

OAK RIDGE National Laboratory

Lawrence Livermore National Laboratory

DOE Exascale (10^{18} FLOPS!) Initiative

National Lab	Los Alamos NATIONAL LABORATORY EST. 1943	Argonne NATIONAL LABORATORY	OAK RIDGE National Laboratory	Lawrence Livermore National Laboratory
HPC Machine				
Hardware Accelerator	*Intel's Knight's Landing*	*Intel's Knight's Hill*	*NVIDIA's Volta GPU*	*NVIDIA's Volta GPU*

This hardware is going to require us to think differently when programming!

Call to Action

- **Need applications to take advantage of these new machines!**
- **Need to understand how older applications function on new hardware!**
  - MCNP6 at LANL
  - ITS at Sandia
  - Mercury at LLNL
  - GEANT4 at CERN

Our Goals: The Path Forward

- **Examine novel and existing techniques for proton transport.**
- **Determine portability and performance on new hardware.**
- **Develop template GPU code that can be expanded.**

We need to explore what is possible with our new hardware!

Overview

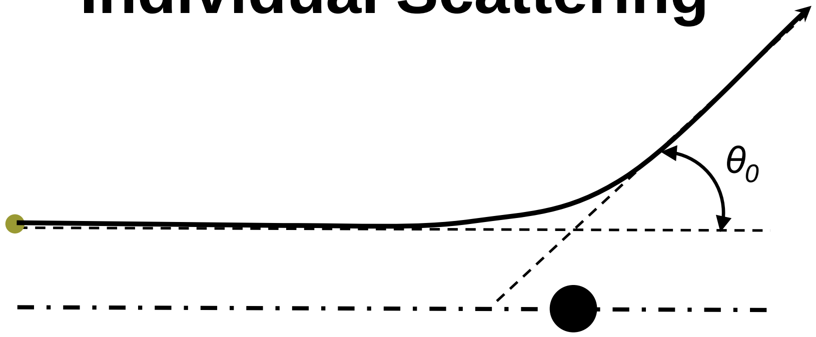
- Background
- **Proton Transport**
- GPUs
- Performance Tests
- Physics Test Problems
- Proposed Work
- Wrap-Up

- **Proton Transport 101**
- **Charged Particle Challenges**
- **Step-Based Physics**
- **Single-Event Physics**

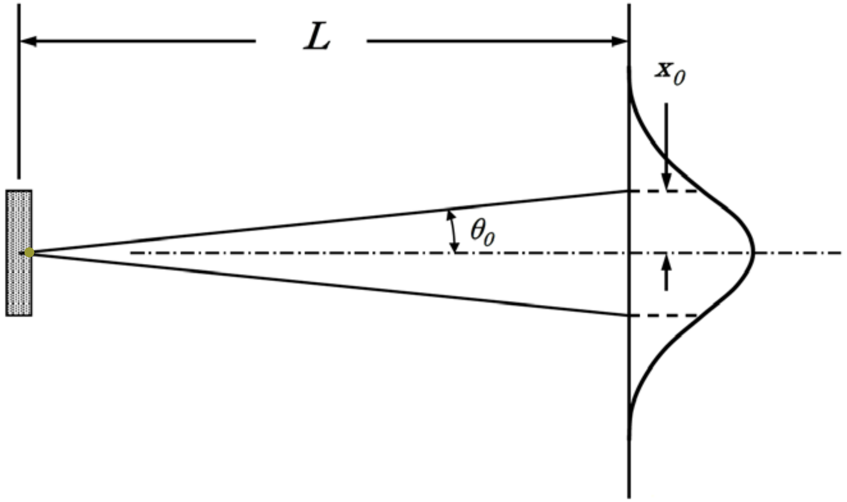
Proton Transport 101

Atomic Scale

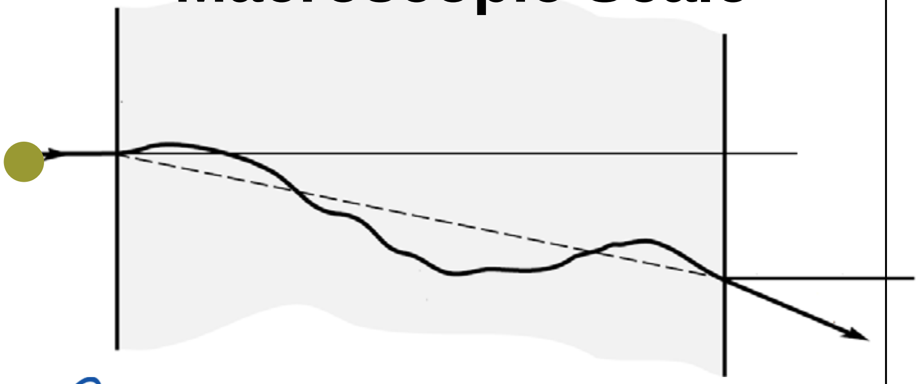
Individual Scattering



Average Behavior



Macroscopic Scale



Charged Particle Challenges

- **Charges act over infinite distances!**
- **Can't represent every interaction.**
  - *Analog Problem*
  - Useful for validation
  - Too slow for practical use
- **How can we work around this?**

Coulomb's Law

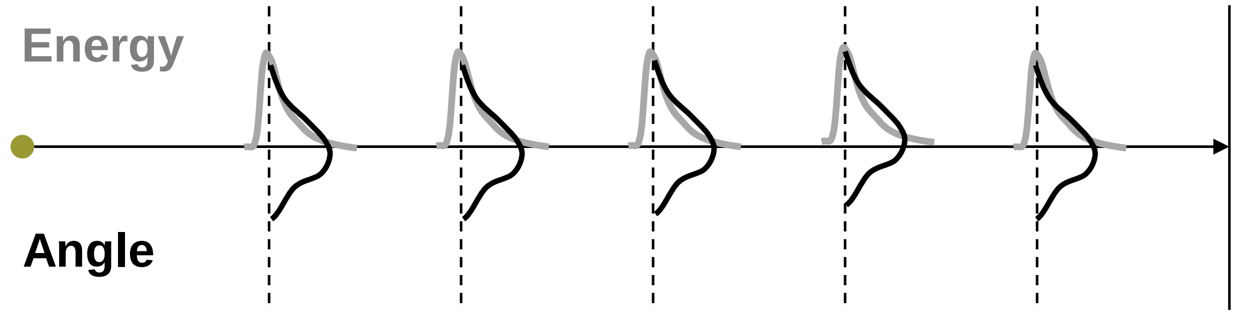
$$F = k_e \frac{q_1 q_2}{r^2}$$

but…

$$\lim_{r \to \infty} \frac{1}{r^2} \approx 0$$

Step-Based Method (Just Scattering)

- Traditional method for representing many scatters.
- Sample distance to a “hard” collision.
- Create substep distance from CSDA tables.
- Sample from Vavilov to determine energy loss about step.
- Sample from Moliere to determine scatter angle about step.



Single-Event Method (Just Scattering)

- **Many options to represent single-event scattering.**
  - Hybrid Methods
  - Moment Preservation
  - Analog Simulation
- **Differences are in how the scattering cross-section is determined.**
- **Step length is sampled from an exponential distribution of mean free path.**

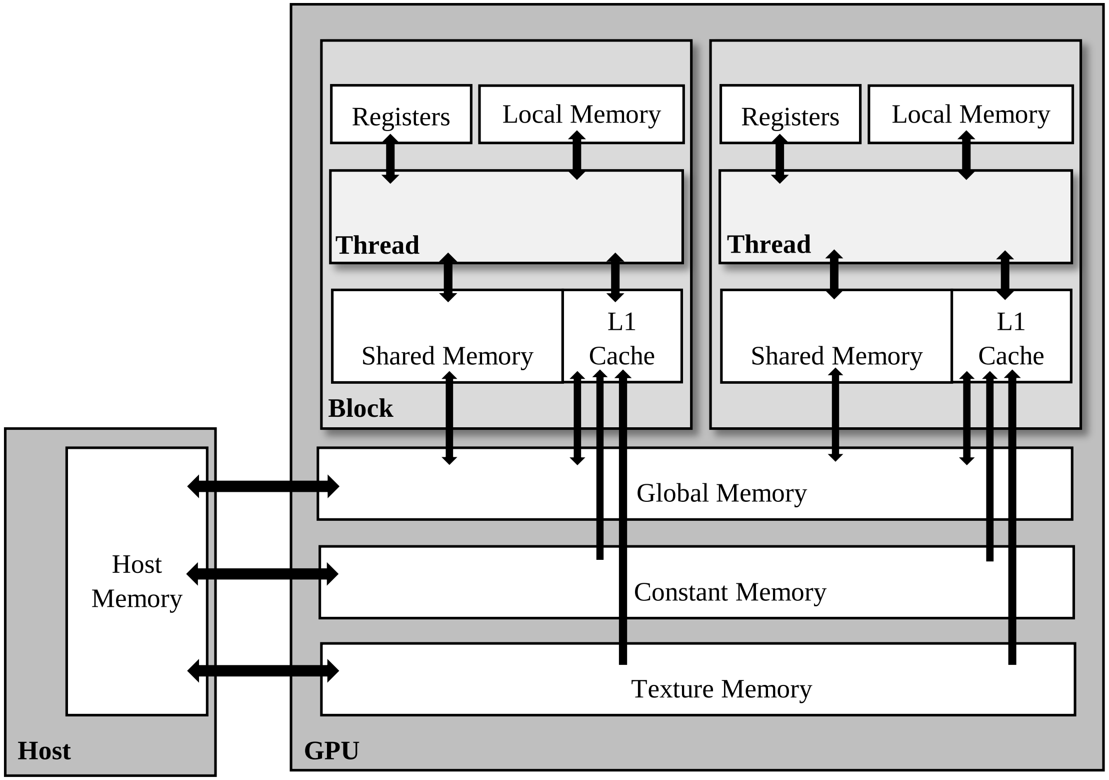
Overview

- Background
- Proton Transport
- **GPUs**
- Performance Tests
- Physics Test Problems
- Proposed Work
- Wrap-Up

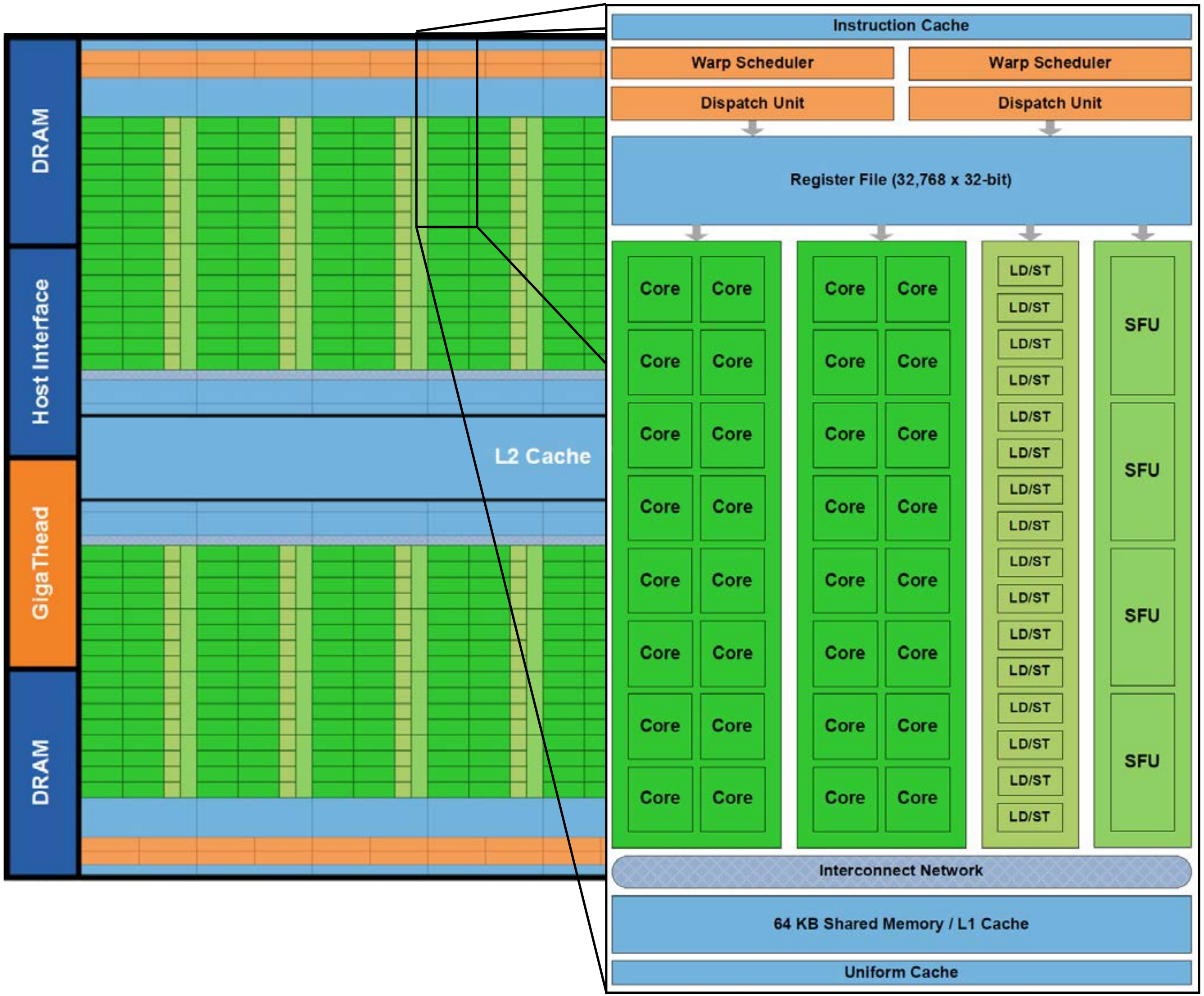
- **Features**
  - Memory Hierarchy
  - Hardware Architecture
- **Issues**
  - Memory Coalescence
  - Bank Conflicts
  - Thread Divergence

Features: Memory Hierarchy

- Users have explicit control over multiple memory types.
- Several memory types are controlled by the device.
- Benefits and limitations to each.



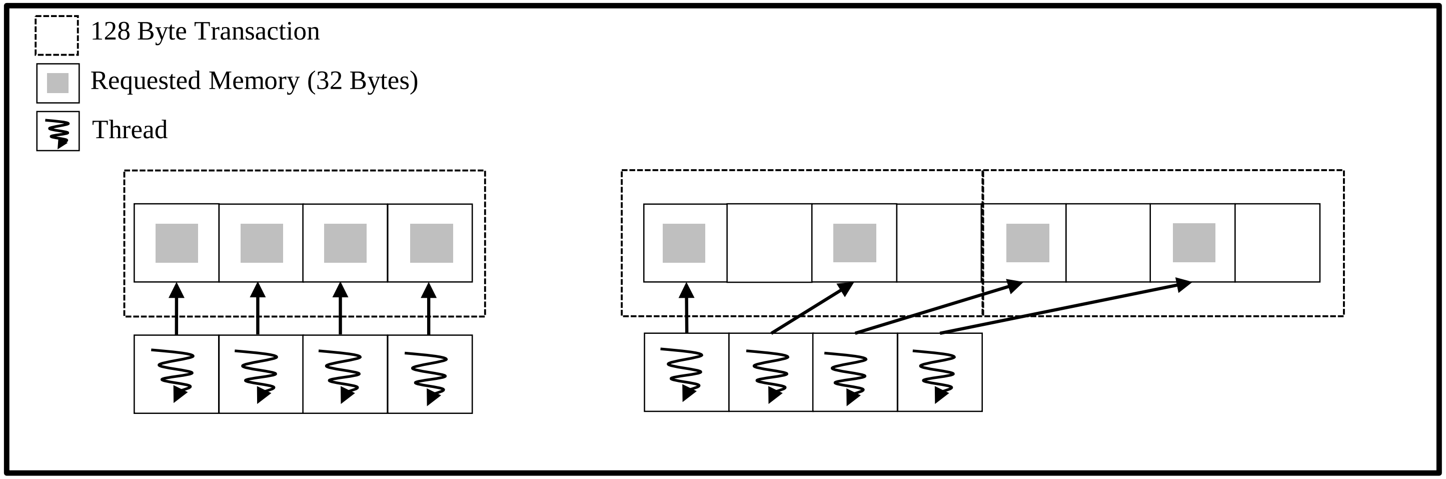
Features: Hardware Architecture



UNCLASSIFIED

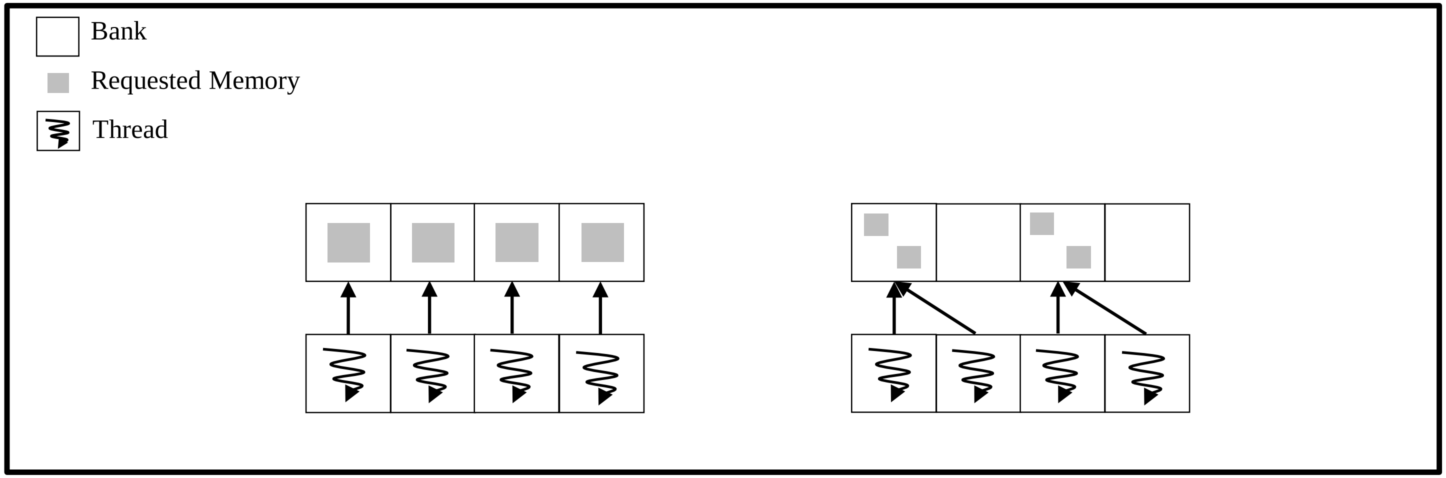
Issues: Memory Coalescence

- GPU reads from global memory in fixed, sequential pieces.
- Device tries to *coalesce* requests into as few reads as possible.
- Performance impacts can be substantial from this.



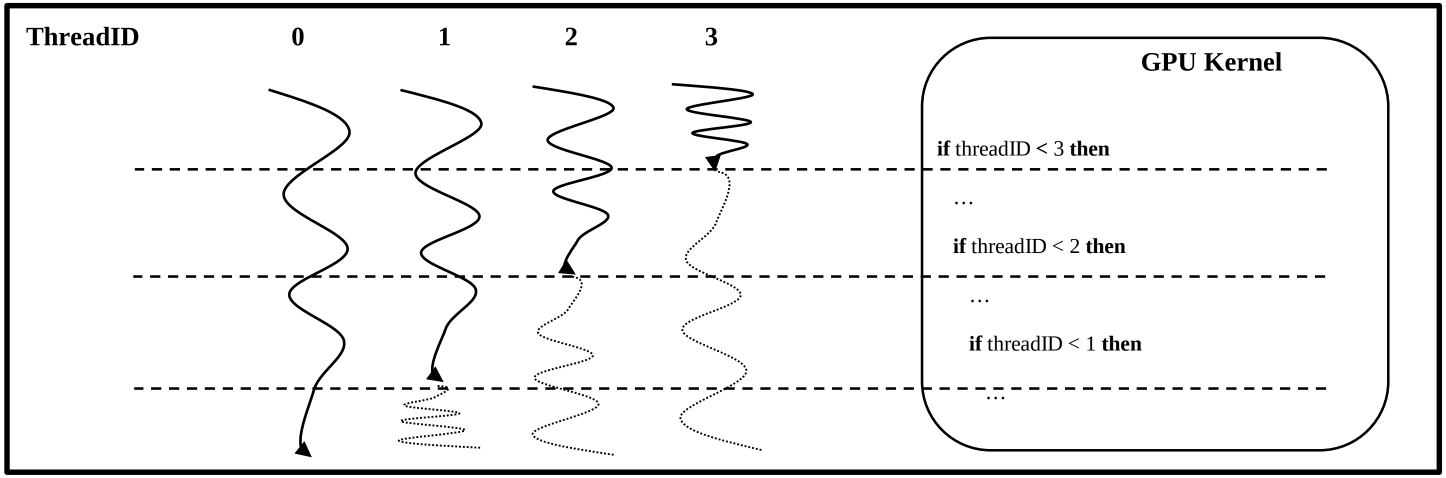
Issues: Bank Conflicts

- **Shared memory is divided into banks.**
- **Reads can occur simultaneously from separate banks.**
- **Requests from the same bank result in a *conflict*, are resolved in serial.**



Issues: Thread Divergence

- GPU instructions are executed across 32 threads at a time.
- *If* and *If-else* statements cause branching.
- Some threads remain idle while others continue branched operations.

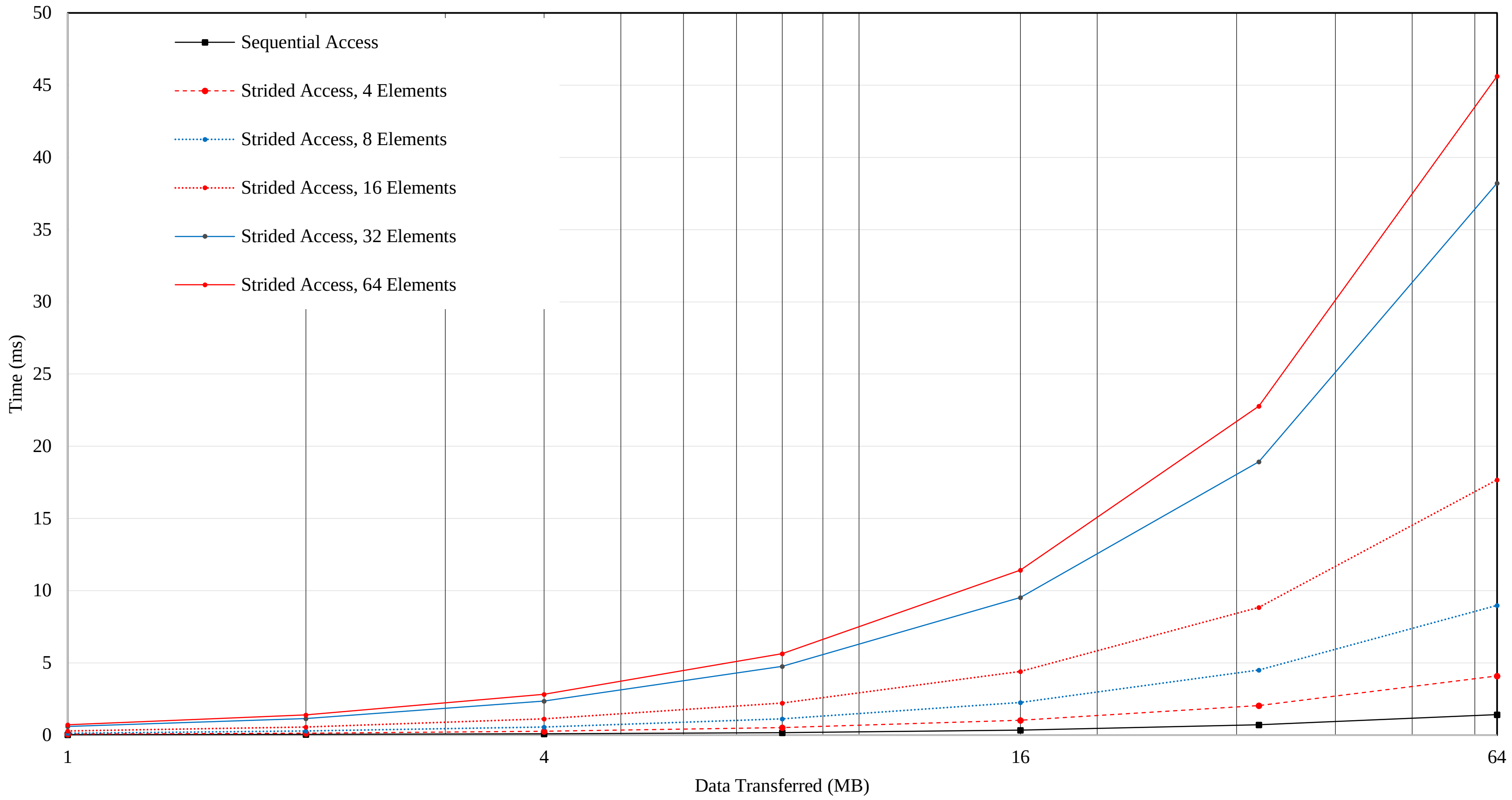


Overview

- Background
- Proton Transport
- GPUs
- **Performance Tests**
- Physics Test Problems
- Proposed Work
- Wrap-Up

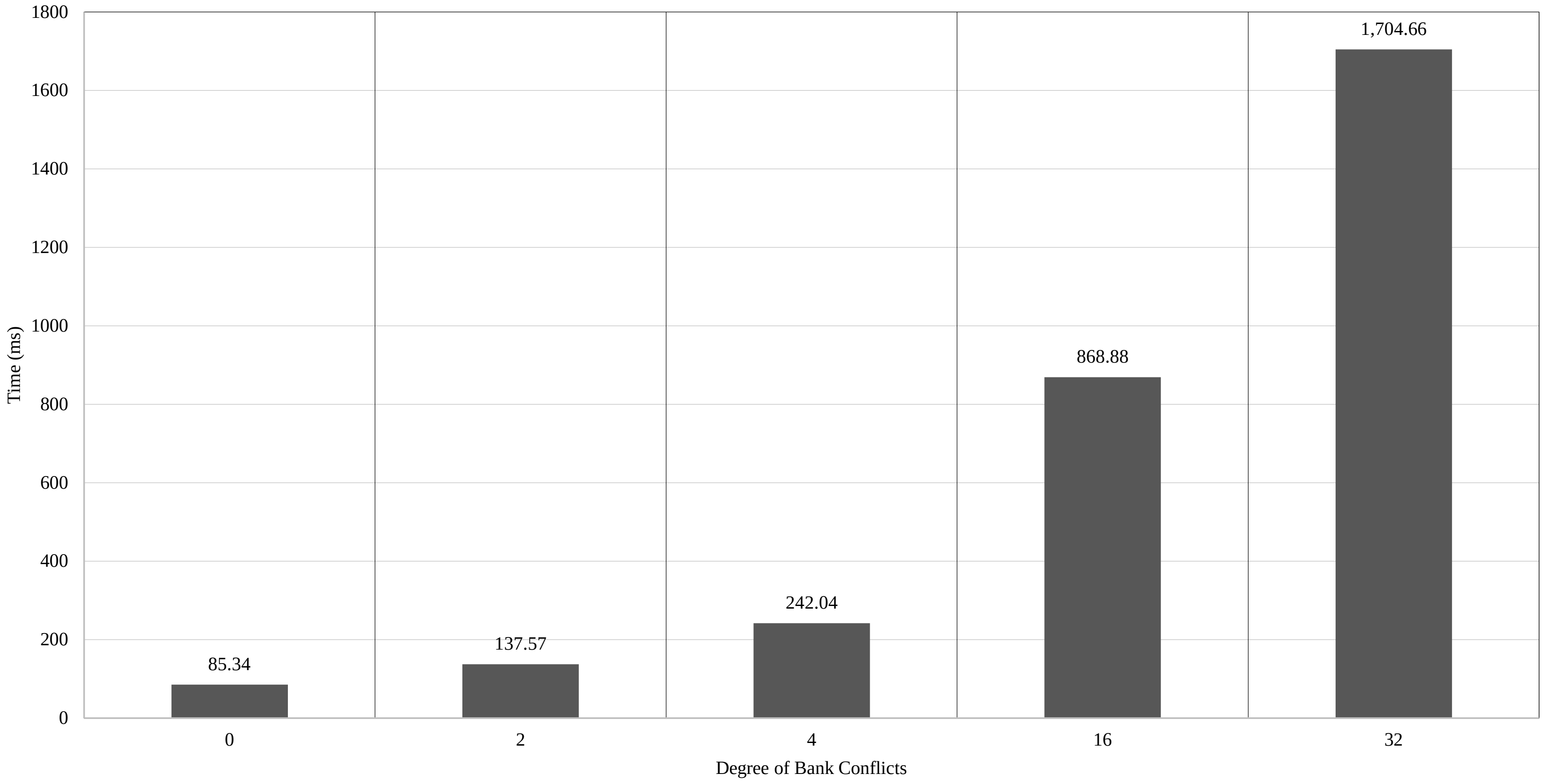
- **Memory Coalescence**
- **Bank Conflicts**
- **Single Value Access**

Memory Coalescence



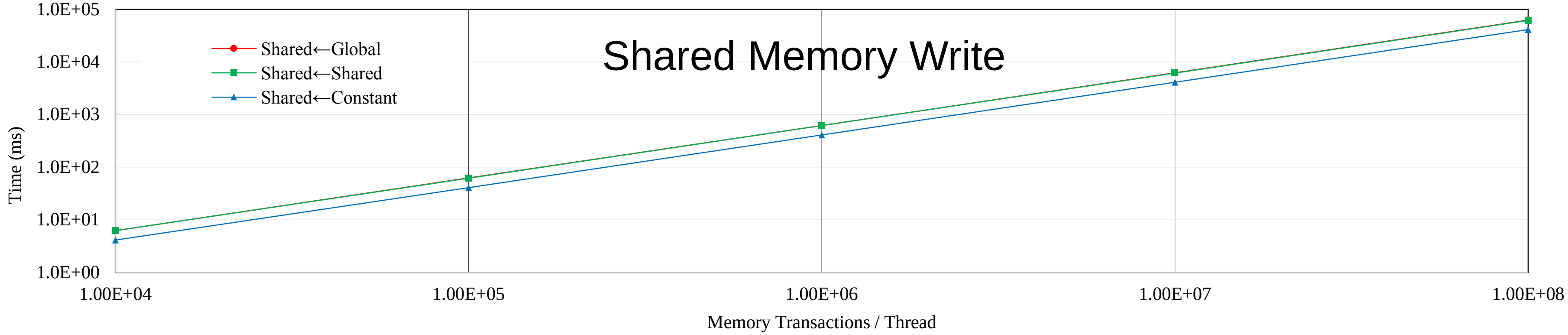
UNCLASSIFIED

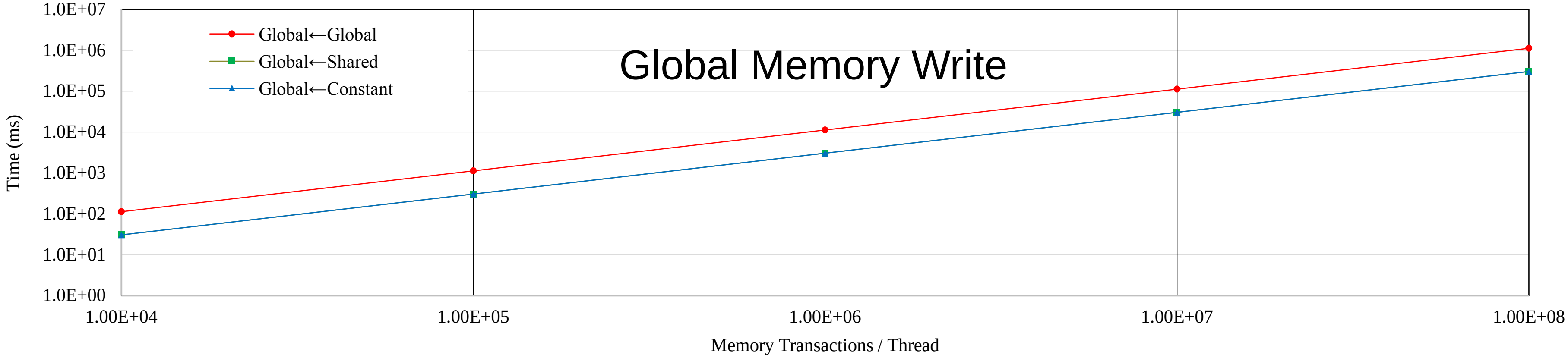
Bank Conflicts



UNCLASSIFIED

Single Value Access





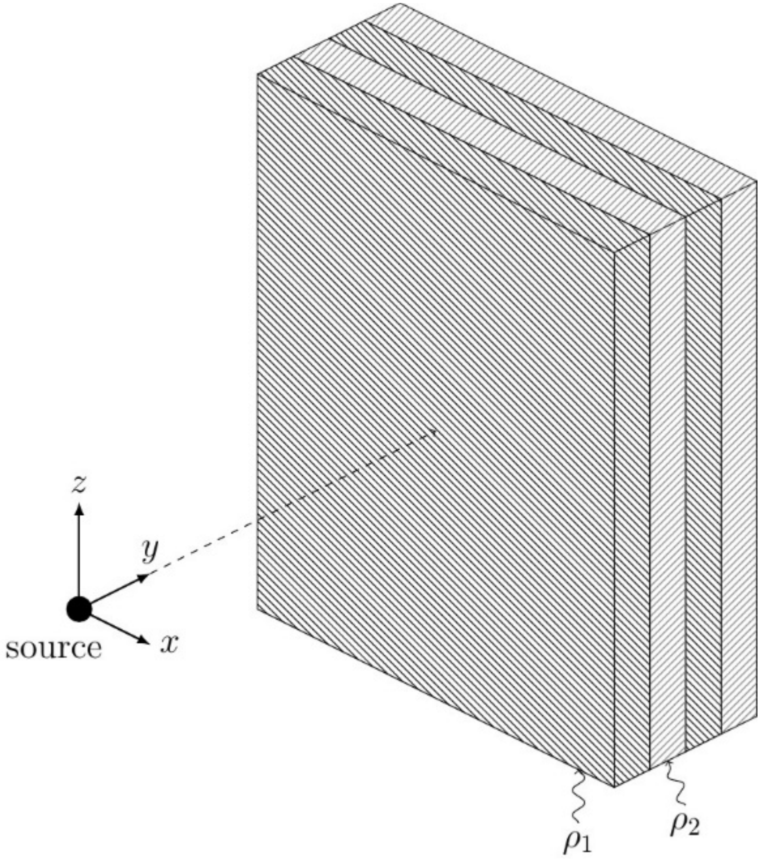
Overview

- Background
- Proton Transport
- GPUs
- Performance Tests
- **Physics Test Problems**
- Proposed Work
- Wrap-Up

- **Boundary Crossing**
- **Limbing**
- **Large Angle Scatter**

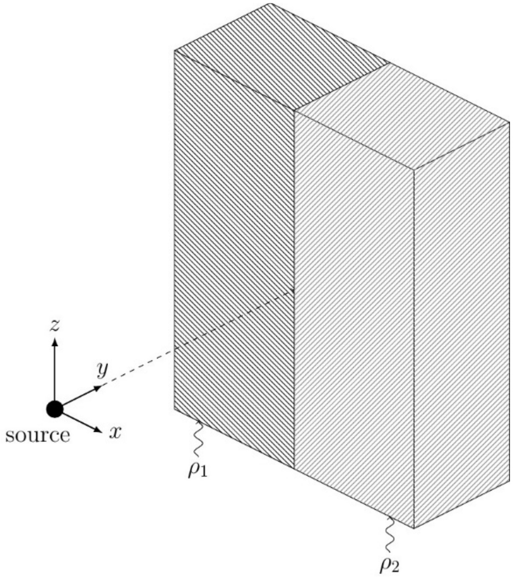
Boundary Crossing

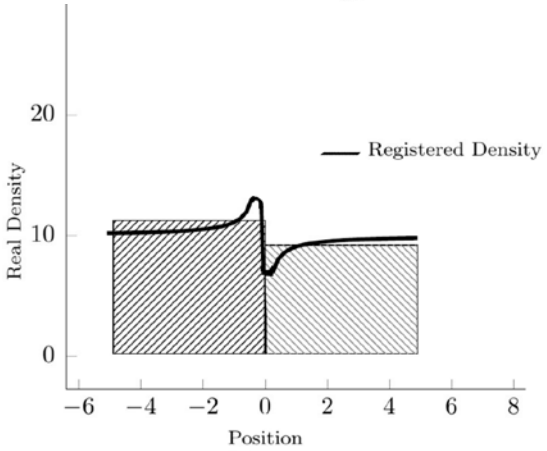
- **Multiple thin slabs of high density and low density material.**
- **Step-based methods make boundary crossing approximations.**
- **Single-event should show better agreement with analog models.**



Limbing

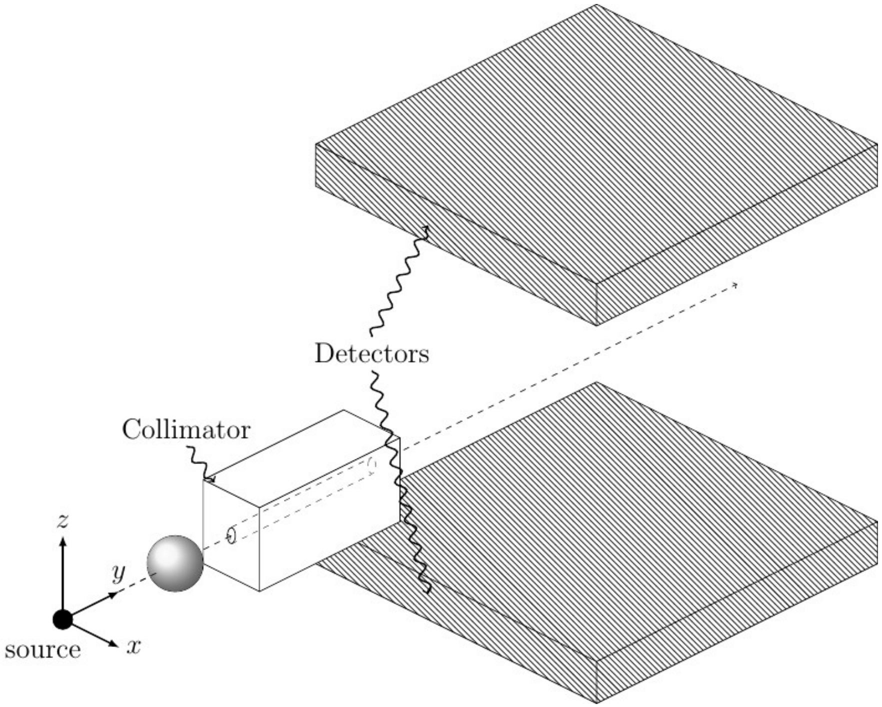
- ***Limbing* occurs between high density and low density material.**
- **Radiographic problem.**
- **Overestimate of transmission at high density region.**
- **Good applied test.**
  - Medical Imaging
  - Proton Radiography





Large Angle Scatter

- **Step-based methods typically rely on a small-angle approximation:**
- **Detector values will be inaccurate compared to analog models**
- **Single-event methods should be more accurate.**



UNCLASSIFIED

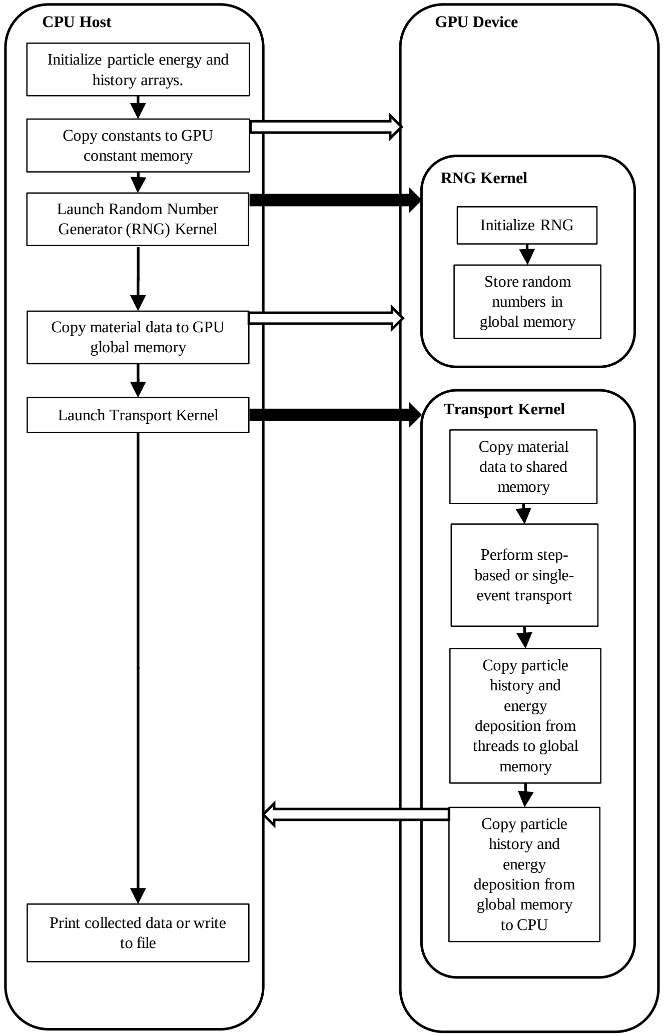
Overview

- Background
- Proton Transport
- GPUs
- Performance Tests
- Physics Test Problems
- **Proposed Work**
- Wrap-Up

- **General Transport Algorithm**
- **Tuning**
  - Global Memory
  - Shared Memory
  - Single Value Access
  - Thread Divergence
- **Tuned Algorithm**

General Transport Algorithm

- **Naïve implementation.**
- **Provide baseline for tuning validation.**
- **Noticeable Issues**
  - No memory management
  - No special thought for transport algorithm
  - Maximum history is limited by hardware.


CPU Host
Initialize particle energy and history arrays.
Copy constants to GPU constant memory
Launch Random Number Generator (RNG) Kernel
Copy material data to GPU global memory
Launch Transport Kernel
Print collected data or write to file
GPU Device
RNG Kernel
Initialize RNG
Store random numbers in global memory
Transport Kernel
Copy material data to shared memory
Perform step-based or single-event transport
Copy particle history and energy deposition from threads to global memory
Copy particle history and energy deposition from global memory to CPU

UNCLASSIFIED

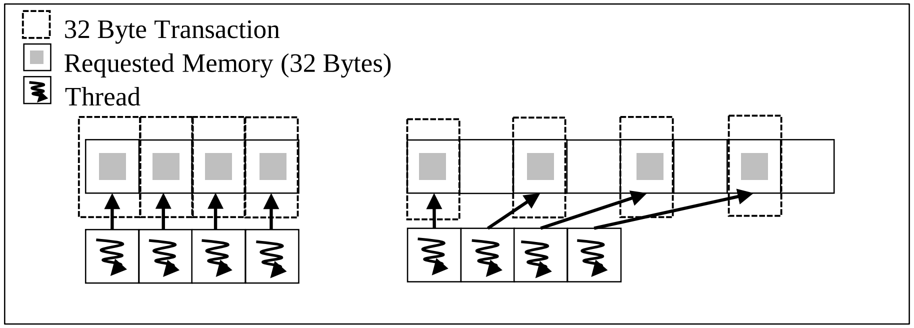
Tuning: Global Memory

Disabling L1 Cache

- Uses compiler flag, simple!
- Results in smaller transaction sizes.
- Good for scattered data access.
- May result in slower reads

```
nvcc  -Xptxas  -dlcm=cg  -o test test.cu
```

- Call to the NVIDIA compiler
- Specify options for PTX assembler
- Change default load cache modifier
- Business as usual!

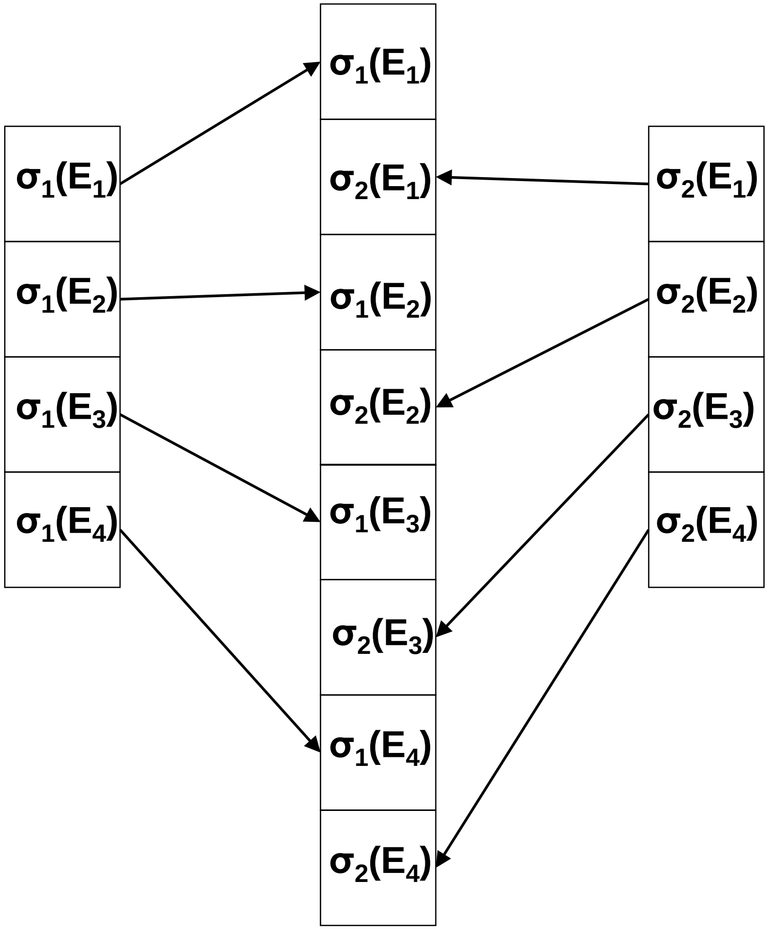


UNCLASSIFIED

Tuning: Global Memory

Restructuring Data

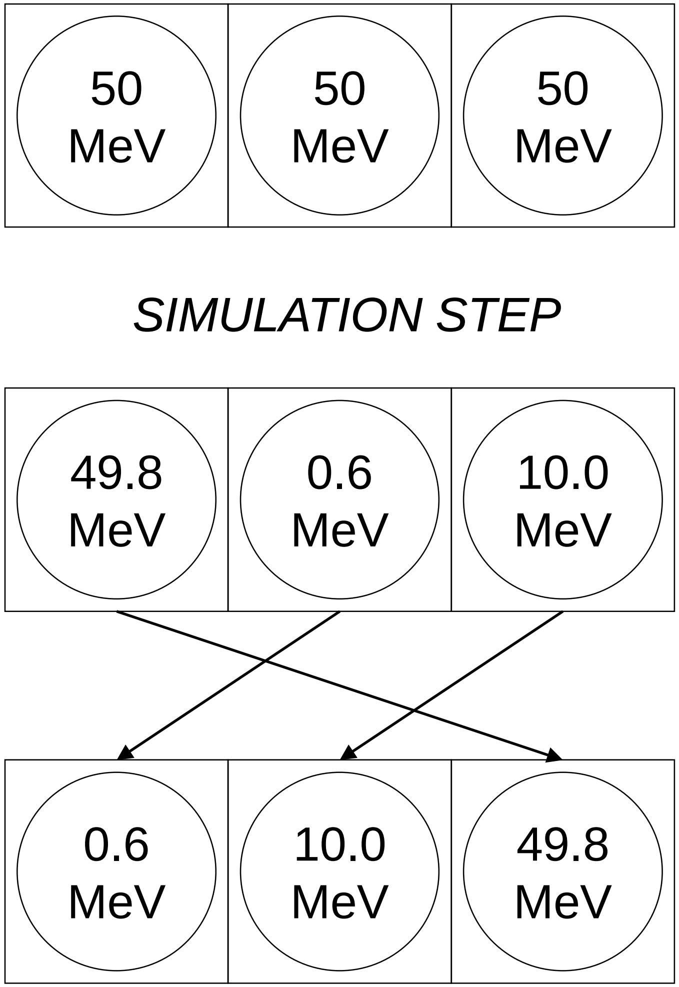
- **Must know existing access pattern through profiling.**
- **Beneficial if data already has some structure to it.**
- **Have the option to *interleave* frequently accessed data.**



Tuning: Global Memory

Sorting Particles

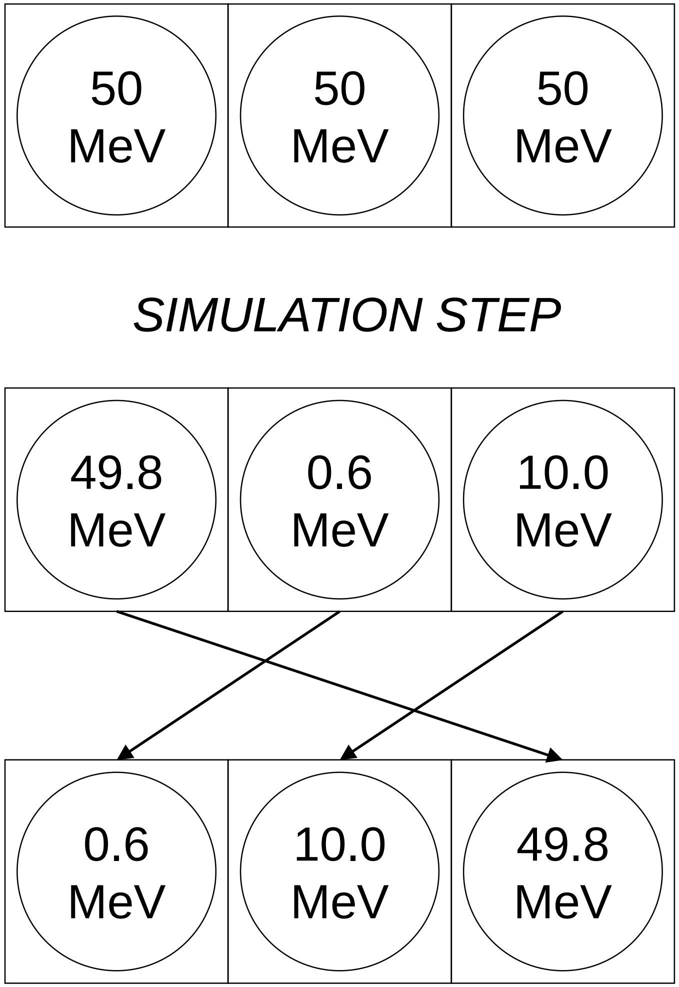
- Introduce memory locality
- Energies that are close will be processed on threads that are close.
- Can hash particles based on location as well.



Tuning: Shared Memory

Sorting Particles

- **Shared memory access also will benefit.**
- **More likely to access data sequentially.**
- **Bank conflicts less likely to occur.**



Tuning: Single Value Access

Dynamic Memory Selection

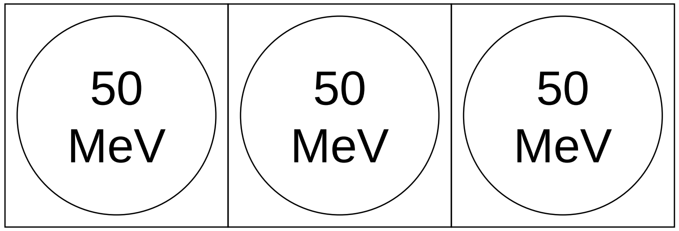
- **Constant memory should be utilized as frequently as possible.**
- **Develop a flag for input.**
- **When $N_{material}$ < *threshold***
  - Use Constant Memory
  - Use Different Access Pattern

Write Location	Banks	Threads per Bank	Read Operations per Thread	Global Read (ms)	Shared Read (ms)	Constant Read (ms)
Global	256	256	10000	112.17	30.40	30.47
	256	256	100000	1118.55	303.59	303.46
	256	256	1000000	11184.64	3035.65	3035.31
	256	256	10000000	111837.59	30355.34	30353.09
	256	256	100000000	1118413.13	303573.44	303499.28
Shared	256	256	10000	6.25	6.20	4.13
	256	256	100000	62.14	61.63	41.02
	256	256	1000000	621.10	616.01	409.93
	256	256	10000000	6215.21	6159.77	4098.74
	256	256	100000000	62107.87	61597.08	40986.92

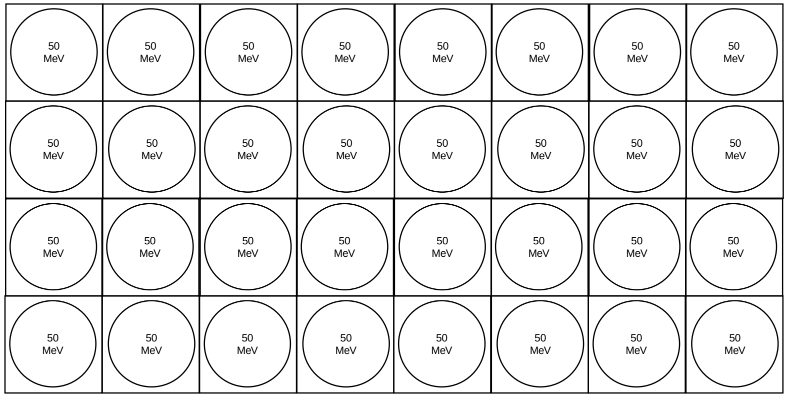
Tuning: Thread Divergence

Parallelism through Variance Reduction

- **Longer lived particles can stall a warp.**
- **Split particle across new warp (32 threads)**
- **Ensure more work for the GPU**



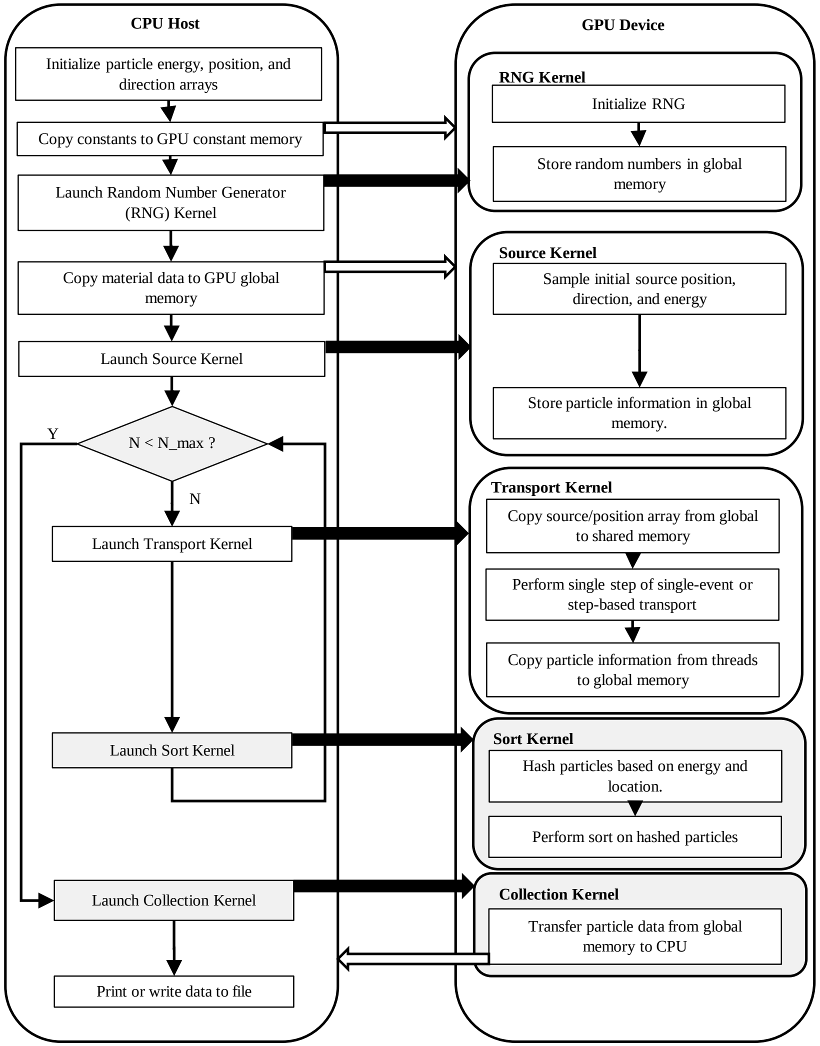
50 MeV	50 MeV	50 MeV



50 MeV	50 MeV	50 MeV	50 MeV	50 MeV	50 MeV	50 MeV	50 MeV
50 MeV	50 MeV	50 MeV	50 MeV	50 MeV	50 MeV	50 MeV	50 MeV
50 MeV	50 MeV	50 MeV	50 MeV	50 MeV	50 MeV	50 MeV	50 MeV
50 MeV	50 MeV	50 MeV	50 MeV	50 MeV	50 MeV	50 MeV	50 MeV

Tuning: Thread Divergence

- **Example of what will change.**
- **Significant difference from naïve implementation.**
- **More thought is given to limitations of the GPU devices**
- **Plays to GPU strengths**



CPU Host

Initialize particle energy, position, and direction arrays

Copy constants to GPU constant memory

Launch Random Number Generator (RNG) Kernel

Copy material data to GPU global memory

Launch Source Kernel

N < N_max ?

Y

N

Launch Transport Kernel

Launch Sort Kernel

Launch Collection Kernel

Print or write data to file

GPU Device

RNG Kernel

Initialize RNG

Store random numbers in global memory

Source Kernel

Sample initial source position, direction, and energy

Store particle information in global memory.

Transport Kernel

Copy source/position array from global to shared memory

Perform single step of single-event or step-based transport

Copy particle information from threads to global memory

Sort Kernel

Hash particles based on energy and location.

Perform sort on hashed particles

Collection Kernel

Transfer particle data from global memory to CPU

Overview

- Background
- Proton Transport
- GPUs
- Performance Tests
- Physics Test Problems
- Proposed Work
- **Wrap-Up**
- **Timeline**
- **Possible Future Goals**

Timeline

- **End of Calendar Year 2016**
  - Functioning CPU code with performance profiling
  - Comparison of single-event and step-based method on CPU
- **Beginning of Spring 2017**
  - Functioning GPU code with performance profiling
  - Comparison of single-event and step-based method on GPU
- **End of Spring 2017**
  - Tuning strategies validation
- **End of Summer 2017**
  - Complete comparison report of CPU & GPU codes
  - Include report on viable tuning strategies

Possible Future Goals: The Path Forward

- Include heavy ions
- Include electrons
  - Physics models are surprisingly different
  - May have to rethink transport algorithm
- Include neutral particles
- Maintain independent tuning suite
  - Allows use between different particle types

Adapt the work for a production code!