

# Performance Analysis of GPU-accelerated Applications with HPCToolkit

John Mellor-Crummey Professor, Rice University



extremecomputingtraining.anl.gov

## Outline

- Introduction to HPCToolkit performance tools
  - -Overview of HPCToolkit components and their workflow
  - -HPCToolkit's graphical user interfaces
- Analyzing the performance of GPU-accelerated codes with HPCToolkit
  - -GAMESS (OpenMP)
  - —Deepwave (Pytorch)
  - -Quicksilver (CUDA)
  - -PeleC (AMReX)
  - -LAMMPS at Exascale (Kokkos)
- Coming attractions
- Hands-on with HPCToolkit
  - Installing HPCToolkit's hpcviewer graphical user interface on your laptop
  - Analyzing executions with HPCToolkit's hpcviewer user interface
  - Collecting performance measurements for sample programs (or your own code)
- Troubleshooting





## Linux Foundation's HPCToolkit Performance Tools

### Measure and analyze performance of CPU and GPU-accelerated applications

- Easy: profile unmodified application binaries
- Fast: low-overhead measurement
- Informative: understand where an application spends its time and why
  - -call path profiles associate metrics with application source code contexts
  - -optional hierarchical traces to understand execution dynamics
- Broad audience
  - -application developers
  - -framework developers
  - -runtime and tool developers
- Measures complex programs on a broad range of platforms
  - -CPU: x86\_64, Power, ARM
  - -GPU: NVIDIA, AMD, Intel





## How does HPCToolkit Differ from NVIDIA's Tools?

- NVIDIA NSight Systems
  - -tracing of CPU and GPU streams
  - -analyze traces when you open them with the GUI
    - long running traces are huge and thus extremely slow to analyze, limiting scalability
  - -designed for measurement and analysis within a node
- NVIDIA NSight Compute
  - -detailed measurement of kernels with counters and execution replay
  - -very slow measurement
  - -flat display of measurements within GPU kernels
- HPCToolkit
  - ---supports more scalable tracing than Nsight Systems
    - measure exascale executions across many GPUs and nodes
  - ---scalable, parallel post-mortem analysis vs. non-scalable in-GUI analysis
  - -detailed reconstruction of estimates for calling context profiles within GPU kernels





## **HPCToolkit's Workflow for CPU Applications**

























## **Measurement of CPU and GPU-accelerated Applications**

- Sampling using Linux timers and hardware counter overflows on the CPU
- Callbacks when GPU operations are launched and (sometimes) completed
- Event stream for GPU operations; PC Samples (NVIDIA, AMD, Intel)
- Binary instrumentation of GPU kernels on Intel GPUs for fine-grain measurement





## **Call Stack Unwinding to Attribute Costs in Context**

- Unwind when timer or hardware counter overflows
  - -measurement overhead proportional to sampling frequency rather than call frequency
- Unwind to capture context for events such as GPU kernel launches



### Calling context tree







## hpcrun: Measure CPU and/or GPU activity

- GPU profiling
  - -hpcrun -e gpu=xxx <app> ....
- GPU PC sampling (NVIDIA GPU only) —hpcrun -e gpu=nvidia,pc <app>
- CPU and GPU Tracing (in addition to profiling)

   hpcrun -e CPUTIME -e gpu=xxx -t <app>
   hpcrun -e CPUTIME -e gpu=xxx -tt <app> # boosted resolution CPU traces for GPU tracing
- Use hpcrun with MPI on Polaris

--mpiexec -n <ranks> ... hpcrun -e gpu=xxx <app>



xxx ∈ {nvidia, amd, opencl, level0}



## Measure and Attribute Performance of Python-driven Codes

Challenge: Straightforward samplingbased approach attributes metrics to Python interpreter code rather than application-level Python source code

Approach: Develop approach to map implementation-level measurements back to Python source code











## hpcstruct: Analyze CPU and GPU Binaries Using Multiple Threads

### • Usage

```
hpcstruct [--gpucfg yes] <measurement-directory>
```

- What it does
  - Recover program structure information
    - Files, functions, inlined templates or functions, loops, source lines
  - In parallel, analyze all CPU and GPU binaries that were measured by HPCToolkit
    - -analyze large application binaries with 16 threads
    - -analyze multiple small application binaries concurrently with 2 threads each
  - Cache binary analysis results for reuse when analyzing other executions

NOTE: --gpucfg yes needed only for analysis of GPU binaries for interpreting PC samples





#### Step 4:

•







### hpcprof/hpcprof-mpi: Associate Measurements with Program Structure

- Analyze data from modest executions with multithreading (moderate scale) hpcprof <measurement-directory>
- Analyze data from large executions with distributed-memory parallelism + multithreading (large scale)

mpiexec -n \${NODES} --ppn 1 -depth=64 \
 hpcprof-mpi <measurement-directory>





Step 4:







## **Code-centric Analysis with hpcviewer**







## **Understanding Temporal Behavior**

- Profiling compresses out the temporal dimension
  - -Temporal patterns, e.g. serial sections and dynamic load imbalance are invisible in profiles
- What can we do? Trace call path samples
  - -N times per second, take a call path sample of each thread
  - -Organize the samples for each thread along a time line
  - -View how the execution evolves left to right
  - -What do we view? assign each procedure a color; view a depth slice of an execution







## **Time-centric Analysis with hpcviewer**



ATPESC 2024 EXTREME - SCALE COMPUTING

A multi-level call stack based view of execution over time

Minimap indicates part of execution trace shown

## What Can you Measure?

Parallel programming models

```
Across nodes: MPI, SHMEM, UPC++,
```

```
Within nodes: OpenMP, Kokkos, RAJA, HIP, DPC++, Sycl, CUDA, OpenACC
```

Languages

C, C++, Fortran, Python

Frameworks

Pytorch, Tensorflow (maybe)

Hardware

CPU cores and GPUs within a node

All of the nodes in Polaris





### hpcstruct Example: Analyze 7.7GB TensorFlow library (170MB text) in 77s

hpcviewer	
File View Filter Help	
🔄 Profile: hpcstruct-bin 📑 Trace: hpcstruct-bin	🖞 🛠 영 🎶 🖓 수 위 🔶 이 🕶 💾 💆 🤊 🗗
Main view	Denth: 496 - +
Time Range: [0s, 77s] Rank Range: [Thread 0, Thread 31] Cross Hair: (53s, Thread	0)
	Image: Startistics       main
	Densit::there are a provided and provided and prov
	uction const)_ (libinstructionAPI.so.12.0.1)       ser::LaunchWork(LockFrd         (1) LockFreequeue-uynInst::ParseAPI::ParseFrd       (1) LockFreequeue-uynInst::ParseAPI::ParseFrd         (1) LockFreequeue-uynInst::ParseAPI::ParseFrd       (1) LockFreequeue-uynInst::ParseFrd         (1) LockFreequeue-uynInst::ParseFrd       (1) LockFreequeue-uynInst::ParseFrd         (1) LockFreequeue-uynInst::ParseFrd       (1) LockFrd         (2) LockFreequeue-uynInst::ParseFrd       (1) LockFreequeue-uynInst::ParseFrd         (2) LockFreequeue-uynInst::ParseFrd       (1) LockFrd         (2) LockFreequeue-uynInst::ParseFrd       (1) LockFrd         (2) LockFrd       (2) LockFrd         (2) LockFrd       (2) LockFrd         (3) LockFrd       (3) LockFrd         (3) LockFrd
	Oylinst::ParseAFI:rBaseF1:rBase
	III       Upinist::FaiserFire[Uni_uata::Inductor         III       Dyninst::dyn_c_hash_map <unsigned dr<="" long.="" td="">         III       tbb::interface5::concurrent_hash_map<uns.< td=""></uns.<></unsigned>
0s 2s 4s 6s 8s 10s 12s 14s 16s 18s 20s 22s 24s 26s 28s 30s 32s 34s	36s 38s 40s 42s 44s 46s 48s 50s 52s 54s 56s 58s 60s 62s 64s 66s 68s 70s 72s 74s 76s
Depth view Summary view	8+ 8-

### Analyze 38.1GB data for 2K MPI ranks + 2K GPUs using 1K threads in 41s





Slide credit: Jonathon Anderson

extremecomputingtraining.anl.gov



## **Case Studies**

- GAMESS (OpenMP)
- Quicksilver (CUDA)
- PeleC (AMReX)
- LAMMPS (Kokkos) at exascale





## **Case Study: GAMESS**

General Atomic and Molecular Electronic Structure System (GAMESS)

—general *ab initio* quantum chemistry package

- Calculates the energies, structures, and properties of a wide range of chemical systems
- Experiments
  - GPU-accelerated nodes at a prior Perlmutter hackathon
    - Single node with 4 GPUs
    - Five nodes with 20 GPUs

### Perlmutter node at a glance

AMD Milan CPU 4 NVIDIA A100 GPUs 256 GB memory





			10 IV	
JTILe: gamess.00.x 🗮 Trace: gamess.00.x	G 🔻 🗠	1. P. B. B. B. A.	(a) • (a • [H]	
view		Depth:	40 - +	
me Range: [0s, 53s] Cross Hair: (29s, RANK 0 GPUCONTEXT 1 GPUSTREAM 16)			www.com.com.com.com	
	an a	Call stack Statist	tics GPU Idleness	Blame
		<program root=""></program>		
		main		
		gamess		
		brnchx_		
Jewastral and the internet in the internet internet in the internet inte	75 18 [205-4000489511077c8h6ad42000d21f opubin]	energx_		
internet gpp_mi_jos_pps_etts	75_18_ [30584090489511077Cabbau459090511.gpub16]	wfn_		
		rhfcl_		
		twoei_		
		ompmod_ompmod_two	ei_jk_	
		gpu_ompmod_twoei_	jk_	
		gpu_rhf_j05_ppps_	51	
and the second	The second	nvomp_target [1	ibnvomp.so]	
		launchTarget [lib	nvomp.so]	
		launchHXTarget []	ibnvomp.so]	
		hxLaunch [libnvom	ip.so]	
		launchInternal []	ibnvomp.so]	
		[I] targetLaunch		
		launchInternal [l	ibnvomp.so]	
		<pre><gpu kernel=""></gpu></pre>	250) 253224355	5475535768
		nvkernel_gpu_rhf_	]05_pppsF1L575_18_	_ [305a4
		**********		
" se de de se ne se de ide ide ide ide ide ide se se se se se se ade de de de de de	also also also also also she si			_
v view Summary view	1+	Mini map		
	And A Read And A Long And A Read A Read And A Read A Read And A Re			
Antiparter Antiparter Antiparter Antiparter Antiparter Antiparter Antiparter Antiparter	NAME OF TAXABLE PARTY OF TAXABLE PARTY.			

**GAMESS** original



All CPU threads and GPU streams

	● Select rank to display 🛛	6 9 3 1 1 4 3 4 4 13 • 0 • H
view	Check all Uncheck all Regular expression	Danthi da
e Range: [0s, 53s] Cross Hair: (42s, RANK 0 THREAD 10)		100 Peptin. 40 -
The second se	Filter: GPU	Call stack Statistics GPU Idleness Blame
	Ranks or threads         1       Rawk 0 GPUCONTEXT 1 GPUSTBEAN 16         2       Rawk 1 GPUCONTEXT 2 GPUSTBEAN 26         3       Rawk 2 GPUCONTEXT 2 GPUSTBEAN 26         4       Rawk 3 GPUCONTEXT 2 GPUSTBEAN 26	<pre>chread root&gt; threadPoolEntryPoint [libnvomp.so] hxiExecuteHostTreeBarrierWithTasks [libnvomp [] executeHostTreeBarrierWithTasks [libnvomp [] executeHostTreeBarrier [] waitForNeighborThreads hxidresSWait [libnvomp.so] syscall [libc-2.31.so]</pre>
	Cancel	
25 35 45 55 65 75 85 95 105 125 145 165 185 20s	25 245 265 285 305 325 345 365 385 405 425 4	45 465 485 505 525
view Summary view		S+ 8-



GAMESS original All CF

### All CPU threads and GPU streams









**GAMESS** original

GPU streams: 1 iteration



#### 全 🗄 🍐 fot 🕼 🕅 🗮 🔺 🔐 비 🕶 🕷

Scope	REALTIME (sec): Sum (I)	REALTIME (sec): Sum (E)	GPUOP (sec): Sum (I) 🗢 GPUOP (sec): Sum (E)	GKER (sec): Sum (I) GKER (sec): Sum (E)	GXCOPY (sec): Sum (I) GXCOPY (sec): Sum (E)
▲ Experiment Aggregate Metrics	3.48e+03 100.0	3.48e+03 100.0%	1.66e+02 100.0% 1.66e+02 100.0	% 1.66e+02 100.0% 1.66e+02 100.	.0% 2.64e-01 100.0% 2.64e-01 10
<program root=""></program>	2.04e+02 5.9		1.66e+02 100.0%	1.66e+02 100.0%	2.64e-01 100.0%
▲ w main [gamess.00.x]	2.04e+02 5.9	¥	1.66e+02 100.0%	1.66e+02 100.0%	2.64e-01 100.0%
🖌 » gamess	2.04e+02 5.9	S	1.66e+02 100.0%	1.66e+02 100.0%	2.64e-01 100.0%
A 772 w brnchx_	2.03e+02 5.8	N	1.66e+02 100.0%	1.66e+02 100.0%	2.64e-01 100.0%
▲ 1095 energx_	2.03e+02 5.8	N. Contraction of the second s	1.66e+02 100.0%	1.66e+02 100.0%	2.64e-01 100.0%
▲ loop at gamess.F: 1316	2.03e+02 5.8	N .	1.66e+02 100.0%	1.66e+02 100.0%	2.64e-01 100.0%
▲ loop at gamess.F: 1436	1.97e+02 5.7	¥	1.66e+02 100.0%	1.66e+02 100.0%	2.64e-01 100.0%
loop at gamess.F: 1436	1.97e+02 5.7	N CONTRACTOR OF CONTRACTOR OFONTO OFO	1.66e+02 100.0%	1.66e+02 100.0%	2.64e-01 100.0%
▲ 1440	1.97e+02 5.7	N	1.66e+02 100.0%	1.66e+02 100.0%	2.64e-01 100.0%
▲ loop at gamess.F: 2645	1.97e+02 5.7	N. Contraction of the second s	1.66e+02 100.0%	1.66e+02 100.0%	2.64e-01 100.0%
▲ 2568 w rhfcl_	1.97e+02 5.7	1.04e-02 0.0%	1.66e+02 100.0%	1.66e+02 100.0%	2.64e-01 100.0%
Loop at rhfuhf.f: 2678	1.96e+02 5.7	\$.25e-03 0.0%	1.66e+02 100.0%	1.66e+02 100.0%	2.64e-01 100.0%
∠ loop at rhfuhf.f: 2723	1.96e+02 5.6	N	1.66e+02 100.0%	1.66e+02 100.0%	2.64e-01 100.0%
▲ loop at rhfuhf.f: 2723	1.96e+02 5.6	Ń	1.66e+02 100.0%	1.66e+02 100.0%	2.64e-01 100.0%
∠ 2859 » twoei_	1.69e+02 4.9	N	1.66e+02 100.0%	1.66e+02 100.0%	2.64e-01 100.0%
▲ 3994 ≫ onpmod_onpmod_twoei_jk_	1.69e+02 4.9	N	1.66e+02 100.0%	1.66e+02 100.0%	2.64e-01 100.0%
∠ 246 👾 gpu_ompmod_twoei_jk_	1.69e+02 4.9	4	1.66e+02 100.0%	1.66e+02 100.0%	2.64e-01 100.0%
656 s gpu_rhf_j16_dppp_	2.98e+01 0.9	1.00e-02 0.0%	2.97e+01 17.9%	2.97e+01 17.9%	1.07e-02 4.1%
\$89 gpu_rhf_106_pppp_	2.74e+01 0.8	1 5,00e-03 0.04	2.74e+01 16.5%	2.74e+01 15.5%	8,56e-03 3.24
1			a ee an ac an		



### **GAMESS** original

ile View Filter Help		
Profile: gamess.00.x 🗮 Trace: gamess.00.x 🖼 Profile: gamess.00.x 🗮 Trace: gamess.00.x	🖧   🔝 🖓	🗧 🛟 🌾 🔶 수 🕹 🗳 🕶 🗧 🗧
ain view		Depth: 39
Time Range: [0s, 54s] Cross Hair: (27s, RANK 2 THREAD 0)		
	ATTELLET AND	Call stack Statistics GPU Idleness Blame
		<pre><pre>program root&gt;</pre></pre>
		main
		t gamess
		brnchx_
		energx_
	an a an Thursdon an Thursdon a shakara . The Constant and this Theory same that Theory and the second second s	wfn_
		rhfcl_
		twoei_
		<pre>ompmod_ompmod_twoei_jk_</pre>
		gpu_ompmod_twoei_jk_
		gpu_rhf_j06_pppp_
and a state of the	an Chillion and Chil	nvomp_target [libnvomp.so]
		launchTarget [libnvomp.so]
		launchHXTarget [libnvomp.so]
		hxLaunch [libnvomp.so]
		launchInternal [libnvomp.so]
		[I] targetLaunch
		launchInternal [libnvomp.so]
		libcuda.so.515.48.07@0x2e8ca6
		Libcuda.so.515.48.0/@0x134400
		Libcuda.so.515.48.07@0x480005
		libruda co 515 48 0700x160bf6
		libruda so 515 48.0700x106510
the second descent in the second descent in the second descent in the second descent descent descent descent des		libcuda so 515 48 0700x4b012e
2 2 3 3 4 5 5 6 7 8 8 9 11 13 13 15 17 19 21 23 23 25 27 29 29	33s 35s 37s 39s 41s 43s 45s 47s 49s 51s	535 1:L
h view Summary view		Mini map

GAMESS improved



All CPU threads and GPU streams





GAMESS improved

#### All GPU streams, whole execution





GAMESS improved

All GPU streams: 2 iterations





#### GAMESS improved

### Time-centric Analysis: GAMESS 5 nodes, 40 ranks, 20 GPUs on Perlmutter





GAMESS improved

CPU Threads and GPU Streams

### Time-centric Analysis: GAMESS 5 nodes, 40 ranks, 20 GPUs on Perlmutter





### GAMESS improved


ATPESC 2024 EXTREME - SCALE COMPUTING

GAMESS improved + better manual distribution of work in input





Showing Rank 0 Thread 0 in addition to GPU streams



#### 1 CPU Stream, 2 GPU Streams: 6 Iterations

	빈
File View Filter Help	
🖼 Profile: gamess.00.x 🗮 Trace: gamess.00.x 🖼 Profile: gamess.00.x 🗮 Trace: gamess.00.x 🗮 Profile: gamess.00.x 🗮 Trace: gamess.00.x 🗮 Trace: gamess.01.x	
mthlib.f 23	
1053 END 1054 F	
1955 CHANDRE MITHER *DECK MITARER	
1057 C JOHN THE HITCHING POLICY ADDRESS TO A DRESS	
1059 C 1060 IMPLICIT DOUBLE PRECISION(A-H,0-Z)	
1061 C 1062 DIMENSION A(*), B(NA,MB), AB(NAB,MB)	
1063.C 1054 PARAMETER (ZERO=0.0D+00)	т
1065 C 1066 C* 31.0CT 1979	
1067.C* 1068.C*FURCTION - TO MULTIPLY SYMMETRIC MATRIX A	
1069 C* TIMES RECTANGULAR MATRIX B AND GET RECTANGULAR MATRIX AB	
1071 (* PARAMETERS 1073 (* A - THE INPUT REAL SYMMETRIC MATRIX OF ORDER NA 1073 (* B - THE UNPUT REAL NYMMETRIC STAAGE MODE. 1073 (* NA - THE ORDER OF MATRIX A 1075 (* NA - THE ORDER OF MATRIX A 1075 (* NA - THE ORDER OF MATRIX A 1076 (* NA - THE UNPUT RON DIMENSION OF MATRICS B AND AB 1078 (* NA - THE UNPUT RON DIMENSION OF MATRIX AB 1079 (* INCA - ADDRESS DIFFRENCE OF TWO ADJACENT ELEMENTS OF A 1080 (* INCA - ADDRESS DIFFRENCE OF TWO ADJACENT ELEMENTS OF A 1080 (* INCA - ADDRESS DIFFRENCE OF TWO ADJACENT ELEMENTS OF A 1080 (* INCA - ADDRESS DIFFRENCE OF INPUT MATRIX A 1081 (* INCA 1082 (* INCA - ADDRESS DIFFRENCE OF INPUT MATRIX A 1084 (* INCA - ADDRESS DIFFRENCE OF INPUT MATRIX A 1085 (* IJ=-1:NC 1085 (* IJ=-1:NC 1086 (* IJ=-1:NC 1086 (* IJ=-1:NC 1087 (* IJ=-1:NC 1088 (* IJ=-1:NC 1089 (* IJ=-1:NC) 1089 (* AJJA-A(IJ) 1090 (* AJJA-A(IJ) 1091 (* IICK) 1091 (* IICK) 10	
1995 C         1996 C         1997 DJ-INC         1998 DJ ISD IS2, NA         1999 DJ-JINC         1901 DD 140 J-1, IMI         1912 JJ-JINC         1910 IML=T-1         1911 DD 140 J-1, IMI         1912 JJ-JINC         1913 AIJ=A(IJ)         1914 IF(AIJ.60, ZERO) GO TO 140         1915 CALL DAXPY(MB, AIJ, B(I, I), NA, AB(J, I), NAB)         1916 CONTINUE         1917 IAO CONTINUE         1918 ISO CONTINUE         1919 RETURN         1910 END         1910 END         1910 END         1910 END         1910 END	
Top-down view Bottom-up view Flat view	

Inpoviewer	2
File View Filter Help	a/)
🖼 Profile: gamess.00.x 🗮 Trace: gamess.00.x 🖼 Profile: gamess.00.x 🗮 Trace: gamess.00.x 🖳 Profile: gamess.00.x 🖳 Profile: gamess.01.x	
mthlib.f 83	
SNO           SNO	1
1097       IJ=1:NC         1098       D0 150 1=2.NA         1099       1J=1:NC         1000       IM1=r.1         1101       D0 140 3=1.IM1         1022       IJ=1:NC         1033       AIJ=A(IJ)         1044       IF(AIJ:0, E0, ZERO) GO TO 140         1055       CALL DAXPY(MB, AIJ, B(I, I), NA, AB(J, I), NAB)         1066       CALL DAXPY(MB, AIJ, B(J, I), NA, AB(I, I), NAB)         1071       140 CONTINUE         1080       SCONTINUE         1080       RETURN         1090       RETURN	
Top-down view Bottom-up view Flat view	

hpcvlewer     File View Filter Help		1 1 1
🕶 Profile: gamess.00.x 🖷 Trace: gamess.00.x 🖁	🖼 Profile: gamess.00.x 🗮 Trace: gamess.00.x 🔄 Profile: gamess.00.x 🗮 Trace: gamess.00.x 🔤 Profile: gamess.01.x	= 0
mthlib.f 🛛		
1096 C		
1097 IJ=	=1-INC	
1098 DO	150 I=2,NA	
1099	IJ=IJ+INC	
1100	IM1=I-1	
1101	DO 140 J=1,IM1	
1102	IJ=IJ+INC	
1103	AIJ=A(IJ)	
1104	IF(AIJ.EQ.ZERO) GO TO 140	
1105	CALL DAXPY(MB,AIJ,B(I,1),NA,AB(J,1),NAB)	
1106	CALL DAXPY(MB,AIJ,B(J,1),NA,AB(I,1),NAB)	
1107 <b>140</b>	CONTINUE	
1108 <b>150 CON</b>	NTINUE	
1109 RET	TURN	
1110 END		
Top-down view Bottom-up view Flat view		

# Measure Pytorch Deepwave (RTM) on CPU + NVIDIA GPU

rofile: python3.9 🗮 Trace: p	ython3.9 Profile: python3.9 Trace: python3.9								
lar_born.py × rtm.py		-				-			
ma	odel_gradient_sampling_interval= odel_gradient_sampling_interval)								
<pre>def scalar_born(vz.Tensor.sc grid_spacing: dt: float, sc source_locati receiver_loca accuracy: int pml_freq: 0pt max_vel: 0pti survey_pad: ( wavefield_0: wavefield_mi: -down view Bottom-up view Fla</pre>	<pre>catter: Tensor. : Union[int, float, List[float], Tensor], Tensor], Durce_amplitudes: Optional[Tensor] = None, ations: Optional[Tensor] = None, ta4, pmL_width: Union[int, List[int]] = 20, tional[float] = None, ional[float] = None, Dptional[Inion[int, List[Optional[int]]]] = None, Optional[Tensor] = None, : Optional[Tensor] = None, titue</pre>								
	11 - 12								
pe	Ar Ar III <b>* ¥</b> 5	CPUTIME (sec): Sum 3.91e+02 10	(I) CPUTIME	E (sec): Sum (E 3.91e+02 100.	) GINS: Sum	(I)	GINS:STL_ANY: Su	m (I) GINS:STL_ANY: Sum (E)	GINS:STL_IFET: 2,230+
Second Secon	Ar Ar III ▼ MC	CPUTIME (sec): Sum 3.91e+02 10 2.15e+02 5	(I) CPUTIME	E (sec): Sum (E 3.91e+02 100.	) GINS: Sum .0% 2.70e+12 1.75e+12	n (I) - GINS: Sum (E) 100.0% 2.70e+12 100.0 64.9%	GINS:STL_ANY: Su I.95e+12 1.41e+12	m (I) GINS:STL_ANY: Sum (E) 100.0% I.95e+12 100.0% 72.5%	GINS:STL_IFET 2.23e 9.69e
the second	A A III ▼ M	CPUTIME (sec): Sum 3.91e+02 10 2.15e+02 5 2.05e+02 5	(1) CPUTIME 00.0% 55.1% 52.5%	E (sec): Sum (E 3,91e+02 100.	) GINS: Su .0% 2./0e+12 1.75e+12 1.75e+12	n (I) → GINS: Sum (E) 100.0% 2.70e+12 100.0 64.9% 64.9%	GINS:STL_ANY: Su 1.95e+12 1.41e+12 1.41e+12	m (I) GINS:STL_ANY: Sum (E) 100.0% I.95e+12 100.0% 72.5% 72.5%	GINS:STL_IFET 2.23e 9.69e 9.69e
		CPUTIME (sec): Sum 3.91e+02 10 2.15e+02 5 2.05e+02 5 2.05e+02 5	(1) CPUTIME 00.0% 55.1% 52.5% 52.5%	E (sec): Sum (E 3,91e+02 100.	) GINS: Sum .0% 2.70e+12 1.75e+12 1.75e+12 1.75e+12	n (I) ♥ GINS: Sum (E) 100.0% 2.70e+12 100.0 64.9% 64.9%	GIN5:STL_ANY: Su 1.95e+12 1.41e+12 1.41e+12 1.41e+12 1.41e+12	m (1) GINS:STL_ANY: Sum (E) 100.0% I.95e+12 100.0% 72.5% 72.5%	GINS:STL_IFET: 2.23e 9.69e 9.69e 9.69e
the set of the se	Python calls	CPUTIME (sec): Sum 3.91e+02 10 2.15e+02 10 2.05e+02 10 2.05e+02 10 2.05e+02 10	(I) CPUTIME 00.0% 55.1% 52.5% 52.5% 52.5%	E (sec): Sum (E 3,91e+02 100.	) GINS: Sum .0% 2.70e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12	(I) ▼ GINS: Sum (E) 100.0% 2.70e+12 100.0 64.9% 64.9% 64.9%	GINS:STL_ANY: Su 1.95e+12 1.41e+12 1.41e+12 1.41e+12 1.41e+12	(1) GINS:STL_ANY: Sum (E) 100.0% I.05e+12 100.0% 72.5% 72.5% 72.5%	GINS:STL_IFET 2.23e 9.69e 9.69e 9.69e 9.69e
fix 20 IN E /  periment Aggregate Metrics  program root>  152 // wrapper      sus_grad      step	Python calls	CPUTIME (sec): Sum 3.91e+02 10 2.05e+02 1 2.05e+02 1 2.05e+02 1 2.05e+02 1 2.05e+02 1	(1) CPUTIME 00.0% 55.1% 52.5% 52.5% 52.5% 52.5%	E (sec): Sum (E 3,91e+02 100. 3.58e-02 0.	) GINS: Sum 2.70e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12	(1) GINS: Sum (E) 100.05 2.700+12 100.0 64.95 64.95 64.95 64.95	GINS:STL_AWY: Su 1.95e+12 1.41e+12 1.41e+12 1.41e+12 1.41e+12 1.41e+12	m (1) GINS:STL_ANY: Sum (E) 100.0% I.95e+12 100.0% 72.5% 72.5% 72.5% 72.5%	GINS:STL_IFET 2.23e 9.69e 9.69e 9.69e 9.69e 9.69e 9.69e
IN X X IVI E  periment Aggregate Metrics program root>  152 > wrapper      .use_grad      .step      .closure      .colar born	Python calls	CPUTIME (sec): Sun 3.91e+02 10 2.15e+02 5 2.05e+02 5 2.05e+02 5 2.05e+02 5 2.05e+02 5 2.05e+02 5	(1) CPUTIME 00.0% 55.1% 52.5% 52.5% 52.5% 52.5% 52.5%	E (sec): Sum (E 3.91e+02 100. 3.58e-02 0.	GINS: Sum 2.70e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12	GINS: Sum (E) 100.0% 2./00+12 100.0 64.9% 64.9% 64.9% 64.9% 64.9%	GIN5:STL_ANY: SU 1.95e+12 1.41e+12 1.41e+12 1.41e+12 1.41e+12 1.41e+12 1.41e+12 1.41e+12	m (1) GINS:STL_ANY: Sum (E) 100.0% 1.95e+12 100.0% 72.5% 72.5% 72.5% 72.5% 72.5%	GINS:STL_IFET 2.23e 9.69e 9.69e 9.69e 9.69e 9.69e 9.69e
<ul> <li>J× W IM E</li> <li>periment Aggregate Metrics</li> <li>program root&gt;</li> <li>152 w wrapper</li> <li>use_grad</li> <li>w step</li> <li></li> <l< td=""><td>Python calls</td><td>CPUTIME (sec): Sue 3.91e+02 10 2.15e+02 5 2.05e+02 5 2.05e+02 5 2.05e+02 5 2.05e+02 5 2.05e+02 5 2.03e+02 5</td><td>(1) CPUTIME 00.0% 55.1% 52.5% 52.5% 52.5% 52.5% 52.5%</td><td>E (sec): Sum (E 3,91e+02 100, 3,58e-02 0,</td><td>GINS:         Sum           .0%         2.70e+12           1.75e+12         1.75e+12           1.75e+12         1.75e+12           .0%         1.75e+12           .0%         1.75e+12           1.75e+12         1.75e+12</td><td>(1) → GTNS: Sum (E) 100.0% 2.7/0e+12 100.0 64.9% 64.9% 64.9% 64.9% 64.9% 64.9% 64.9%</td><td>GINS:STL_ANY: Su 1.95e+12 1.41e+12 1.41e+12 1.41e+12 1.41e+12 1.41e+12 1.41e+12 1.41e+12 1.41e+12 1.41e+12</td><td>n (1) GINS:STL_ANY: Sun (E) 100.05 T.95e+12 100.05 72.55 72.55 72.55 72.55 72.55 72.55 72.55 72.55</td><td>GINS:STL_IFFT 2.236 9.696 9.696 9.696 9.696 9.696 9.696 9.686</td></l<></ul>	Python calls	CPUTIME (sec): Sue 3.91e+02 10 2.15e+02 5 2.05e+02 5 2.05e+02 5 2.05e+02 5 2.05e+02 5 2.05e+02 5 2.03e+02 5	(1) CPUTIME 00.0% 55.1% 52.5% 52.5% 52.5% 52.5% 52.5%	E (sec): Sum (E 3,91e+02 100, 3,58e-02 0,	GINS:         Sum           .0%         2.70e+12           1.75e+12         1.75e+12           1.75e+12         1.75e+12           .0%         1.75e+12           .0%         1.75e+12           1.75e+12         1.75e+12	(1) → GTNS: Sum (E) 100.0% 2.7/0e+12 100.0 64.9% 64.9% 64.9% 64.9% 64.9% 64.9% 64.9%	GINS:STL_ANY: Su 1.95e+12 1.41e+12 1.41e+12 1.41e+12 1.41e+12 1.41e+12 1.41e+12 1.41e+12 1.41e+12 1.41e+12	n (1) GINS:STL_ANY: Sun (E) 100.05 T.95e+12 100.05 72.55 72.55 72.55 72.55 72.55 72.55 72.55 72.55	GINS:STL_IFFT 2.236 9.696 9.696 9.696 9.696 9.696 9.696 9.686
Jo Zo IVI E      Joz Zo IVI E      periment Aggregate Metrics      program root-      Is2 > wrapper      _ use_grad      step      /> sclosure      scalar_born      /> call_      //>     //>     //>     //	Python calls	CPUTIME (sec): Sue 3.91+02 10 2.15e+02 2 2.05e+02 2 2.05e+02 2 2.05e+02 2 2.05e+02 2 2.05e+02 2 2.03e+02 2 2.03e+02 2 2.03e+02 2	(1) CPUTIME 00.0% 55.1% 52.5% 52.5% 52.5% 52.5% 52.5% 51.8% 51.8%	E (sec): Sum (E 3.91e+02 100. 3.58e-02 0.	GINS: Sum 2.700+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12	(1) → GTNS: Sum (E) 100.0% 2.70e+12 100.0 64.9% 64.9% 64.9% 64.9% 64.9% 64.9% 64.9% 64.9% 64.9%	GINS:STL_ANY: Su 1.95e12 1.41e+12	a (1) GINS:STL_ANY: Sum (E) 100.00 T.956+12 100.00 72.55 72.55 72.55 72.55 72.55 72.55 72.55 72.55 72.55 72.55	GINS:STL_JFET 2.236 9.696 9.696 9.696 9.696 9.696 9.696 9.686 9.686
ix X X IVI I X periment Aggregate Metrics program root> 152 > wrapper 4 > use_grad 4 > step 4 > scalar_born 4 > scalar_born 4 > scalar_born lpyth 4 > sphindl1::cp fu	Python calls	CPUTIME (sec): Sun 3.91e+02 10 2.15e+02 5 2.05e+02 5 2.05e+02 5 2.05e+02 5 2.05e+02 5 2.05e+02 5 2.03e+02 5 2.05e+02 5 2.03e+02	(1) CPUTIME 00.0% 55.1% 52.5% 52.5% 52.5% 52.5% 51.8% 51.8%	5 (sec): Sum (E 3.91++02 100. 3.58e-02 0.	GINS: Sum 2. /0+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12	(1) → GINS: Sum (E) 100.05 2.700+12 100.05 64.95 64.95 64.95 64.95 64.95 64.95 64.95 64.95 64.95 64.95 64.95	GINS:SSTL_ANY: Su 1.99e+12 1.41e+12 1.41e+12 1.41e+12 1.41e+12 1.41e+12 1.41e+12 1.41e+12 1.41e+12 1.41e+12 1.41e+12 1.41e+12 1.41e+12 1.41e+12	(1) GINS:STL_ANY: Sum (E) 100.0% I.095e+12 100.0% 72.5% 72.5% 72.5% 72.5% 72.5% 72.5% 72.5% 72.5% 72.5% 72.5%	GINSISTL 1FET 2.23 9.69 9.69 9.69 9.69 9.69 9.69 9.68 9.68
IN THE AUGUST AND AND AUGUST AND AUGUST A	Python calls	CPUTIME (sec): Sun 3.91e+02 10 2.15e+02 2 2.05e+02 2 2.05e+02 2 2.05e+02 2 2.05e+02 2 2.05e+02 2 2.03e+02 2 2.03e+02 2 2.03e+02 2 2.03e+02 2 2.03e+02 2 2.03e+02 2	(1) CPUTIME 00.0% 55.1% 52.5% 52.5% 52.5% 52.5% 52.5% 51.8% 51.8%	E (soc): Sum (E 3,91e+02 100. 3.58e-02 0.	GINS: Sum 2. /0+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12	1 (1) ★ OTNS: Sum (E) 100.0% 2.70e+12 100.0 64.9% 64.9% 64.9% 64.9% 64.9% 64.9% 64.9% 64.9% 64.9% 64.9% 64.9%	GINS:SSTL_ANY: Su 1.95e+12 1.41e+12 1.41e+12 1.41e+12 1.41e+12 1.41e+12 1.41e+12 1.41e+12 1.41e+12 1.41e+12 1.41e+12 1.41e+12 1.41e+12 1.41e+12	m (1) GTN5:STL_ANY: Sum (E) 100.0v 1.95e+12 100.0v 72.5% 72.5% 72.5% 72.5% 72.5% 72.5% 72.5% 72.5% 72.5% 72.5%	01115151L2 1 FE 1 2.234 9.694 9.694 9.694 9.694 9.694 9.694 9.684 9.684 9.684 9.684
fx Xx IM E /// E /// E /// E //// E //////////	Python calls on.6585557/ nction::dispatcher(_object*, _object*, _object*) [ h_python.so]: 0 function::initialize <torch::jit::initjit8indings(< td=""><td>CPUTINE (sec): Sum 3.91+402 10 2.15+402 5 2.05+402 5 2.05+402 5 2.05+402 5 2.05+402 5 2.03+402 5 2.03+4025</td><td>(1) CPUTIME 00.0% 55.1% 52.5% 52.5% 52.5% 52.5% 52.5% 52.5% 52.5% 51.8% 51.8% 51.8%</td><td>E (sec): Sum (E 3.91e+02 100. 3.58e-02 0.</td><td>GINS: Sum VS 2.70+12 1.75+12 1.75+12 1.75+12 1.75+12 1.75+12 1.75+12 1.75+12 1.75+12 1.75+12 1.75+12 1.75+12 1.75+12 1.75+12</td><td>(1) ★ OTNG: Sum (E) 100.0% 2.7/0e+12 100.0 64.9% 64.9% 64.9% 64.9% 64.9% 64.9% 64.9% 64.9% 64.9% 64.9% 64.9%</td><td>GINS:STL_ANY: Su I.95e+12 1.41e+12</td><td>n (1) GINS:STL_ANY: Sun (E) 100.05 1.95e+12 100.05 72.55 72.55 72.55 72.55 72.55 72.55 72.55 72.55 72.55 72.55 72.55 72.55</td><td>0111515112 1 FE 1 2.234 9.694 9.694 9.694 9.694 9.694 9.684 9.684 9.684 9.684 9.684 9.684 9.684</td></torch::jit::initjit8indings(<>	CPUTINE (sec): Sum 3.91+402 10 2.15+402 5 2.05+402 5 2.05+402 5 2.05+402 5 2.05+402 5 2.03+402 5 2.03+4025	(1) CPUTIME 00.0% 55.1% 52.5% 52.5% 52.5% 52.5% 52.5% 52.5% 52.5% 51.8% 51.8% 51.8%	E (sec): Sum (E 3.91e+02 100. 3.58e-02 0.	GINS: Sum VS 2.70+12 1.75+12 1.75+12 1.75+12 1.75+12 1.75+12 1.75+12 1.75+12 1.75+12 1.75+12 1.75+12 1.75+12 1.75+12 1.75+12	(1) ★ OTNG: Sum (E) 100.0% 2.7/0e+12 100.0 64.9% 64.9% 64.9% 64.9% 64.9% 64.9% 64.9% 64.9% 64.9% 64.9% 64.9%	GINS:STL_ANY: Su I.95e+12 1.41e+12	n (1) GINS:STL_ANY: Sun (E) 100.05 1.95e+12 100.05 72.55 72.55 72.55 72.55 72.55 72.55 72.55 72.55 72.55 72.55 72.55 72.55	0111515112 1 FE 1 2.234 9.694 9.694 9.694 9.694 9.694 9.684 9.684 9.684 9.684 9.684 9.684 9.684
<pre>jx Xv IVI = / periment Aggregate Metrics program root&gt; 152 w wrapper * _use_grad * step * _call_ * _call_ * _callbern * _ callbern * _ call_bern * _ c</pre>	Python calls n.688845671 nction::dispatcher(_object*, _object*, _object*) [ h_python.sol: 0 function::initialize <torch::jit::initjjtbindings( get operation for overload or packet(std::vector<s< td=""><td>CPUTIME (sec): Sue 3.91+02 10 2.05+02 5 2.05+02 5 2.05+02 5 2.05+02 5 2.05+02 5 2.03+02 5 2.03+025 2.05</td><td>(1) CPUTIME UU.0% 55.1% 52.5% 52.5% 52.5% 52.5% 51.8% 51.8% 51.8% 51.8%</td><td>E (sec): Sum (E 3.91e+02 100. 3.58e-02 0.</td><td>) GINS: Sum 05 2.70e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12</td><td>(1) → GTNS: Sum (E) 100.0% 2.700+12 100.0 64.9% 64.9% 64.9% 64.9% 64.9% 64.9% 64.9% 64.9% 64.9% 64.9% 64.9%</td><td>GINS:STL_ANY Su 1.99e12 1.41e+12</td><td>a (1) GINS:STL_ANY: Sum (E) 100.00 T.95e+12 100.00 72.55 72.55 72.55 72.55 72.55 72.55 72.55 72.55 72.55 72.55 72.55 72.55 72.55</td><td>GINSSETL IFET 2,234 9,694 9,694 9,694 9,694 9,694 9,684 9,684 9,684 9,684 9,684 9,684</td></s<></torch::jit::initjjtbindings( 	CPUTIME (sec): Sue 3.91+02 10 2.05+02 5 2.05+02 5 2.05+02 5 2.05+02 5 2.05+02 5 2.03+02 5 2.03+025 2.05	(1) CPUTIME UU.0% 55.1% 52.5% 52.5% 52.5% 52.5% 51.8% 51.8% 51.8% 51.8%	E (sec): Sum (E 3.91e+02 100. 3.58e-02 0.	) GINS: Sum 05 2.70e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12	(1) → GTNS: Sum (E) 100.0% 2.700+12 100.0 64.9% 64.9% 64.9% 64.9% 64.9% 64.9% 64.9% 64.9% 64.9% 64.9% 64.9%	GINS:STL_ANY Su 1.99e12 1.41e+12	a (1) GINS:STL_ANY: Sum (E) 100.00 T.95e+12 100.00 72.55 72.55 72.55 72.55 72.55 72.55 72.55 72.55 72.55 72.55 72.55 72.55 72.55	GINSSETL IFET 2,234 9,694 9,694 9,694 9,694 9,694 9,684 9,684 9,684 9,684 9,684 9,684
<pre>interface interface i</pre>	Python calls on.608845671 nction::dispatcher(_object*, _object*, _object*) [ h_python.so]: 0 function::initialize-torch::jit::initJJT8indings( get_operation_for_overload_or_packet(std::vector <s :invoke0peratorFromPython(std::vector<std::shared< td=""><td>CPUTIME (sec): Sur 3.91+02 10 2.15e+02 5 2.05e+02 5 2.05e+02 5 2.05e+02 5 2.05e+02 5 2.05e+02 5 2.03e+02 5</td><td>(1) CPUTIME U0.0% S5.1% S2.5% S2.5% S2.5% S2.5% S2.5% S1.8% S1.8% S1.8% S1.8% S1.8%</td><td>E (sec): Sum (E 3.91e+02 100. 3.58e-02 0.</td><td>GINS: Sum .0% 2.70e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12</td><td>(1) → GTNS: Sum (E)           100.0%         2.70e+12           64.9%         64.9%           64.9%         64.9%           64.9%         64.9%           64.9%         64.9%           64.9%         64.9%           64.9%         64.9%           64.9%         64.9%           64.9%         64.9%           64.9%         64.9%           64.9%         64.9%           64.9%         64.9%           64.9%         64.9%</td><td>GINS:STL_ANY: Su 1.95e12 1.41e+12</td><td>(1)         GINS:STL_ANY: Sum (E)           100.0%         T.956+12 100.0%           72.5%         72.5%           72.5%         72.5%           72.5%         72.5%           72.5%         72.5%           72.5%         72.5%           72.5%         72.5%           72.5%         72.5%           72.5%         72.5%           72.5%         72.5%           72.5%         72.5%           72.5%         72.5%           72.5%         72.5%           72.5%         72.5%           72.5%         72.5%           72.5%         72.5%</td><td>GINSESTL IFE 7 2,234 9,696 9,696 9,696 9,696 9,696 9,686 9,686 9,686 9,686 9,686 9,686 9,686</td></std::shared<></s 	CPUTIME (sec): Sur 3.91+02 10 2.15e+02 5 2.05e+02 5 2.05e+02 5 2.05e+02 5 2.05e+02 5 2.05e+02 5 2.03e+02 5	(1) CPUTIME U0.0% S5.1% S2.5% S2.5% S2.5% S2.5% S2.5% S1.8% S1.8% S1.8% S1.8% S1.8%	E (sec): Sum (E 3.91e+02 100. 3.58e-02 0.	GINS: Sum .0% 2.70e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12	(1) → GTNS: Sum (E)           100.0%         2.70e+12           64.9%         64.9%           64.9%         64.9%           64.9%         64.9%           64.9%         64.9%           64.9%         64.9%           64.9%         64.9%           64.9%         64.9%           64.9%         64.9%           64.9%         64.9%           64.9%         64.9%           64.9%         64.9%           64.9%         64.9%	GINS:STL_ANY: Su 1.95e12 1.41e+12	(1)         GINS:STL_ANY: Sum (E)           100.0%         T.956+12 100.0%           72.5%         72.5%           72.5%         72.5%           72.5%         72.5%           72.5%         72.5%           72.5%         72.5%           72.5%         72.5%           72.5%         72.5%           72.5%         72.5%           72.5%         72.5%           72.5%         72.5%           72.5%         72.5%           72.5%         72.5%           72.5%         72.5%           72.5%         72.5%           72.5%         72.5%	GINSESTL IFE 7 2,234 9,696 9,696 9,696 9,696 9,696 9,686 9,686 9,686 9,686 9,686 9,686 9,686
<pre>&gt;  fx &amp;  fx =  // =</pre>	Python calls on.08894567] nction::dispatcher(_object*, _object*, _object*) [ h_python.sol: 0 function::initialize <torch::jit::initjitbindings( get_operation_for_overload_or_packet(std::vector<s :invoke0peratorFromPython(std::vector<std::shared< td=""><td>CPUTIME (sec): Sun 3.91e+02 10 2.15e+02 2 2.05e+02 2 2.05e+02 2 2.05e+02 2 2.05e+02 2 2.05e+02 2 2.03e+02 2 2.03e+02</td><td>(1) CPUTINE 00.0% 55.1% 52.5% 52.</td><td>E (sec): Sum (E 3.91e+02 100. 3.58e-02 0.</td><td>) GINS: Sum 2.70e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12</td><td>(1) ★ OTNG: Sum (E)           100.0% 2.7/0e+12 100.0           64.9%</td><td>GINS:STL_ANY. Su I.95e+12 1.41e+12 1.41e+12 1.41e+12 1.41e+12 1.41e+12 1.41e+12 1.41e+12 1.41e+12 1.41e+12 1.41e+12 1.41e+12 1.41e+12 1.41e+12 1.41e+12 1.41e+12 1.41e+12 1.41e+12 1.41e+12 1.41e+12</td><td>m (1) GTUS&gt;STL_ANY: Sum (E) 100.0v 1.95e+12 100.0v 72.5% 72.5% 72.5% 72.5% 72.5% 72.5% 72.5% 72.5% 72.5% 72.5% 72.5% 72.5% 72.5% 72.5%</td><td>GINSESTL 1FFT 2.23e 9.69e 9.69e 9.69e 9.69e 9.68e 9.68e 9.68e 9.68e 9.68e 9.68e 9.68e 9.68e 9.68e 9.68e</td></std::shared<></s </torch::jit::initjitbindings( 	CPUTIME (sec): Sun 3.91e+02 10 2.15e+02 2 2.05e+02 2 2.05e+02 2 2.05e+02 2 2.05e+02 2 2.05e+02 2 2.03e+02	(1) CPUTINE 00.0% 55.1% 52.5% 52.	E (sec): Sum (E 3.91e+02 100. 3.58e-02 0.	) GINS: Sum 2.70e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12	(1) ★ OTNG: Sum (E)           100.0% 2.7/0e+12 100.0           64.9%	GINS:STL_ANY. Su I.95e+12 1.41e+12 1.41e+12 1.41e+12 1.41e+12 1.41e+12 1.41e+12 1.41e+12 1.41e+12 1.41e+12 1.41e+12 1.41e+12 1.41e+12 1.41e+12 1.41e+12 1.41e+12 1.41e+12 1.41e+12 1.41e+12 1.41e+12	m (1) GTUS>STL_ANY: Sum (E) 100.0v 1.95e+12 100.0v 72.5% 72.5% 72.5% 72.5% 72.5% 72.5% 72.5% 72.5% 72.5% 72.5% 72.5% 72.5% 72.5% 72.5%	GINSESTL 1FFT 2.23e 9.69e 9.69e 9.69e 9.69e 9.68e 9.68e 9.68e 9.68e 9.68e 9.68e 9.68e 9.68e 9.68e 9.68e
<pre>&gt;</pre>	Python calls Python calls on.688845671 nction::dispatcher(_object*, _object*, _object*) [ h_python.sol: 0 function::initializetorch::jit::initJITBindings{ get_operation_for_overload_or_packet(std::vector <s iinvokeOperatorFromPython(std::vector<std::shared tcher::callBoxed(c10::0peratorHandle const6, std:: tcher::callBoxed from unboxed functor<c10::impl::detai< td=""><td>CPUTIME (sec): Sue 3.91e+02 10 2.15e+02 20 2.05e+02 20 2.05e+02 20 2.05e+02 20 2.05e+02 20 2.03e+02 20</td><td>(1) CPUTIME 00.0% 55.1% 52.5% 52.5% 52.5% 52.5% 52.5% 51.8% 51.8% 51.8% 51.8% 51.8% 51.8% 51.8% 51.8%</td><td>E (sec): Sum (E 3.91e+02 100. 3.58e-02 0.</td><td><pre>GING: Sum .vs 270e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12</pre></td><td>(1) → GTNS: Sum (E) 100.0% 2.7/0e+12 100.0 64.9% 64.9% 64.9% 64.9% 64.9% 64.9% 64.9% 64.9% 64.9% 64.9% 64.9% 64.9% 64.9% 64.9%</td><td>GINS:STL_ANY: Su I.99e+12 1.41e+12</td><td>n (1) GINS:STL_ANY: Sun (E) 100.05 1.95e+12 100.05 72.55</td><td>GINSESTL IFFT 2,24 9,69e 9,69e 9,69e 9,69e 9,69e 9,68e 9,68e 9,68e 9,68e 9,68e 9,68e 9,68e 9,68e 9,68e</td></c10::impl::detai<></std::shared </s 	CPUTIME (sec): Sue 3.91e+02 10 2.15e+02 20 2.05e+02 20 2.05e+02 20 2.05e+02 20 2.05e+02 20 2.03e+02 20	(1) CPUTIME 00.0% 55.1% 52.5% 52.5% 52.5% 52.5% 52.5% 51.8% 51.8% 51.8% 51.8% 51.8% 51.8% 51.8% 51.8%	E (sec): Sum (E 3.91e+02 100. 3.58e-02 0.	<pre>GING: Sum .vs 270e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12</pre>	(1) → GTNS: Sum (E) 100.0% 2.7/0e+12 100.0 64.9% 64.9% 64.9% 64.9% 64.9% 64.9% 64.9% 64.9% 64.9% 64.9% 64.9% 64.9% 64.9% 64.9%	GINS:STL_ANY: Su I.99e+12 1.41e+12	n (1) GINS:STL_ANY: Sun (E) 100.05 1.95e+12 100.05 72.55	GINSESTL IFFT 2,24 9,69e 9,69e 9,69e 9,69e 9,69e 9,68e 9,68e 9,68e 9,68e 9,68e 9,68e 9,68e 9,68e 9,68e
<pre>&gt;</pre>	Python calls Python calls on.68864567] mction::dispatcher(_object*, _object*, _object*) [ h_python.sol: 0 function::initialize <torch::jit::initjitbindings( get_operation_for_overload_or_packet(std::vector<s :invokeOperatorFromPython(std::vector<std::shared tcher::callBoxed(c10::0peratorHandle const6, std:: 1::make_boxed_from_unboxed_functor<c10::impl::mdetai c10::guts:iif consteapretrue, c10::impl::mdetai</c10::impl::mdetai </std::shared </s </torch::jit::initjitbindings( 	CPUTIME (sec): Sur 3.91+02 10 2.05+02 5 2.05+02 5 2.05+02 5 2.05+02 5 2.05+02 5 2.03+02 5 2.03+025 2.03+025 2.03+025 2.03+025 2.03+025 2.03+025 2.03+025 2.03+025 2.03+025 2.03+025 2.03+025 2.05+025 2.05+025 2.0	(1) CPUTIME UU.0% SS.1% S2.5% S2.5% S2.5% S2.5% S2.5% S1.8% S1.8% S1.8% S1.8% S1.8% S1.8% S1.8% S1.8%	E (sec): Sum (E 3.91e+02 100. 3.58e-02 0.	<ul> <li>GINS: Sum</li> <li>GINS: Sum</li> <li>C.70e+12</li> <li>1.75e+12</li> <li>1.75e+1</li></ul>	(1) → GTNS: Sum (E) 100.0% 2.700+12 100.0 64.9% 64.9% 64.9% 64.9% 64.9% 64.9% 64.9% 64.9% 64.9% 64.9% 64.9% 64.9% 64.9% 64.9% 64.9%	GINS:STL_ANY Su 1.99e12 1.41e+12	a (1) GINS:STL_ANY: Sum (5) 100.05 T.95e+12 100.05 72.55	GINSESTL IFET 2,234 9,696 9,696 9,696 9,696 9,686 9,686 9,686 9,686 9,686 9,686 9,686 9,686 9,686 9,686 9,686
<pre>&gt;</pre>	Python calls on.068845671 nction::dispatcher(_object*, _object*, _object*) [ h_python.so]: 0 function::initialize-torch::jit::initJITBindings( get_operation_for_overload_or_packet(std::vector <s invokeOperatorFromPython(std::vector<std::shared atcher::callBoxed(c10::0peratorHandle_const6, std:: t:make_boxed_from_unboxed_functor<c10::impl::detai c10::guts::if_constexpr<trues.ical:impl::imake_boxe c10::guts::if_constexpr<trues.ical.c10< td=""><td>CPUTIME (sec): Sur 3.91+02 10 2.15e+02 5 2.05e+02 5 2.05e+02 5 2.05e+02 5 2.05e+02 5 2.05e+02 5 2.03e+02 5</td><td>(1) CPUTIME U0.0% S5.1% S2.5% S2.5% S2.5% S2.5% S2.5% S1.8% S1.8% S1.8% S1.8% S1.8% S1.8% S1.8% S1.8% S1.8% S1.8%</td><td>E (sec): Sum (E 3.91e+02 100. 3.58e-02 0.</td><td><ul> <li>GINS: Sum</li> <li>GINS: Sum</li> <li>C.70e+12</li> <li>1.75e+12</li> <li>1.75e+1</li></ul></td><td>(1) → GTNS: Sum (E)           100.0%         2.70e+12           64.9%         64.9%           64.9%         64.9%           64.9%         64.9%           64.9%         64.9%           64.9%         64.9%           64.9%         64.9%           64.9%         64.9%           64.9%         64.9%           64.9%         64.9%           64.9%         64.9%           64.9%         64.9%           64.9%         64.9%           64.9%         64.9%           64.9%         64.9%           64.9%         64.9%</td><td>GINS:STL_ANY: Su I.95e12 1.41e+12</td><td>(1)         GINS:STL_ANY: Sum (E)           100.00         T.956+12 100.00           72.55         72.55           72.55         72.55           72.55         72.55           72.55         72.55           72.55         72.55           72.55         72.55           72.55         72.55           72.55         72.55           72.55         72.55           72.55         72.55           72.55         72.55           72.55         72.55           72.55         72.55           72.55         72.55           72.55         72.55           72.55         72.55           72.55         72.55           72.55         72.55           72.55         72.55</td><td>GINSTOLL 1151 2.234 9.694 9.694 9.694 9.694 9.684</td></trues.ical.c10<></trues.ical:impl::imake_boxe </c10::impl::detai </std::shared </s 	CPUTIME (sec): Sur 3.91+02 10 2.15e+02 5 2.05e+02 5 2.05e+02 5 2.05e+02 5 2.05e+02 5 2.05e+02 5 2.03e+02 5	(1) CPUTIME U0.0% S5.1% S2.5% S2.5% S2.5% S2.5% S2.5% S1.8% S1.8% S1.8% S1.8% S1.8% S1.8% S1.8% S1.8% S1.8% S1.8%	E (sec): Sum (E 3.91e+02 100. 3.58e-02 0.	<ul> <li>GINS: Sum</li> <li>GINS: Sum</li> <li>C.70e+12</li> <li>1.75e+12</li> <li>1.75e+1</li></ul>	(1) → GTNS: Sum (E)           100.0%         2.70e+12           64.9%         64.9%           64.9%         64.9%           64.9%         64.9%           64.9%         64.9%           64.9%         64.9%           64.9%         64.9%           64.9%         64.9%           64.9%         64.9%           64.9%         64.9%           64.9%         64.9%           64.9%         64.9%           64.9%         64.9%           64.9%         64.9%           64.9%         64.9%           64.9%         64.9%	GINS:STL_ANY: Su I.95e12 1.41e+12	(1)         GINS:STL_ANY: Sum (E)           100.00         T.956+12 100.00           72.55         72.55           72.55         72.55           72.55         72.55           72.55         72.55           72.55         72.55           72.55         72.55           72.55         72.55           72.55         72.55           72.55         72.55           72.55         72.55           72.55         72.55           72.55         72.55           72.55         72.55           72.55         72.55           72.55         72.55           72.55         72.55           72.55         72.55           72.55         72.55           72.55         72.55	GINSTOLL 1151 2.234 9.694 9.694 9.694 9.694 9.684
<pre>c ix % IMI c // c // c // c // c // c // c //</pre>	Python calls Python calls on.bbb#4567J nction::dispatcher(_object*, _object*, _object*) [ h_python.so]: 0 function::initialize-torch::jit::initJITBindings( get_operation_for_overload_or_packet(std::vector <s invokeOperatorFromPython(std::vector<std::shared tcher::callBoxed(c10::OperatorHandle const6, std:: l::make_boxed_from_unboxed_functor<c10::impl::detai c10::guts::if_constexpr<true, :10::impl::make_boxe<br="">] c10::guts::detail::_if_constexpr<true>:call<c10< td=""><td>CPUTINE (sec): Sun 3.914+02 10 2.05e+02 5 2.05e+02 5 2.05e+02 5 2.05e+02 5 2.05e+02 5 2.05e+02 5 2.03e+02 5 2.03e+02</td><td>(1) CPUTINE 00.0% 55.1% 52.5% 52.5% 52.5% 52.5% 51.8% 51.8% 51.8% 51.8% 51.8% 51.8% 51.8% 51.8% 51.8%</td><td>E (sec): Sum (E 3.91e+02 100. 3.58e-02 0.</td><td>) GINS: Sum 2.70e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12</td><td>(1) → OTNS: Sum (E) 100.0% 2.7/0e+12 100.0 64.9% 64.9% 64.9% 64.9% 64.9% 64.9% 64.9% 64.9% 64.9% 64.9% 64.9% 64.9% 64.9% 64.9% 64.9% 64.9%</td><td>GINS:STL_ANY. Su I.95e+12 1.41e+12</td><td>n (1) GTUS&gt;STL_ANY: Sum (E) 100.0v 1.95e+12 100.0v 72.5%</td><td>GINSESTL 1FE1 2.23 9.69 9.69 9.69 9.69 9.69 9.68 9.68 9.68</td></c10<></true></true,></c10::impl::detai </std::shared </s 	CPUTINE (sec): Sun 3.914+02 10 2.05e+02 5 2.05e+02 5 2.05e+02 5 2.05e+02 5 2.05e+02 5 2.05e+02 5 2.03e+02	(1) CPUTINE 00.0% 55.1% 52.5% 52.5% 52.5% 52.5% 51.8% 51.8% 51.8% 51.8% 51.8% 51.8% 51.8% 51.8% 51.8%	E (sec): Sum (E 3.91e+02 100. 3.58e-02 0.	) GINS: Sum 2.70e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12 1.75e+12	(1) → OTNS: Sum (E) 100.0% 2.7/0e+12 100.0 64.9% 64.9% 64.9% 64.9% 64.9% 64.9% 64.9% 64.9% 64.9% 64.9% 64.9% 64.9% 64.9% 64.9% 64.9% 64.9%	GINS:STL_ANY. Su I.95e+12 1.41e+12	n (1) GTUS>STL_ANY: Sum (E) 100.0v 1.95e+12 100.0v 72.5%	GINSESTL 1FE1 2.23 9.69 9.69 9.69 9.69 9.69 9.68 9.68 9.68







### Trace Pytorch Deepwave (RTM): CPU threads 0, 48; GPU stream







# Analyze Pytorch Deepwave (RTM) Kernel using PC Sampling

Ipcviewer		
ile View Filter Help		
Profile: python3.9 🗮 Trace: python3.9 🔤 Profile: python3.9 🗮 Trace: python3.9		-
calar_born.py rtm.py scalar_born.cu X Metric properties		
260 wsc_store[i] = wsc_sum;		
<pre>261  } 262  wfo[i] = v[i] * v[i] * dt2<t>() * w sum + 2 * wfc[i] - wfo[i]:</t></pre>	Lich ODL utilization for forward korrol	
<pre>263 wfpsc[i] = v[j] * v[j] * dt2<t>() * wsc_sum +</t></pre>	High GPU utilization for forward kernel	
264 2 * scatter[j] * v[j] * dt2 <t>() * w_sum + 2 * wfcsc[i] - wfosc[i]:</t>	<b>-</b>	
266 }	Many instruction stalls on global moment	
267 }	Many instruction stars on global memory	
208 269 template <typename t=""></typename>		
<pre>270 _global_ void add_sources(T *_restrict wf, T const *_restrict f,</pre>		
271 int64_t const *_restrict sources_i,		
<pre>273 auto source_idx{blockIdx.x * blockDim.x + threadIdx.x};</pre>		
<pre>274 auto shot_idx{blockIdx.y * blockDim.y + threadIdx.y};</pre>		
<pre>275 if (source_idx &gt;= n_sources_per_shot or shot_idx &gt;= n_shots) return; 276 auto b(chot_idx + n_sources_per_shot + source_idx);</pre>		
op-down view Bottom-up view Flat view		
Scope	GINS: Sum (I) GINS:STL_ANY: Sum (I) GINS:STL_GMEM: Sum (E) GSAMP	:UTIL (I)
<pre># w cl0::impl::make_boxed_from_unboxed_functor<cl0::impl::detail::wrapfunctionin< pre=""></cl0::impl::detail::wrapfunctionin<></pre>	htoRuntimeFunctor_ <std::vector<at::tensor, 1.41e+12="" 1.75e+12="" 64.9%="" 72.5%<="" std::al="" td=""><td>83.58</td></std::vector<at::tensor,>	83.58
▲ 575 » [I] cl0::guts::if_constexpr <true, cl0::impl::make_boxed_from_unboxed_fr<="" td=""><td>unctor<cl0::impl::detail::wrapfunctionintoruntimef 1.41e+12="" 1.75e+12="" 64.9%="" 72.5%<="" td=""><td>83.58</td></cl0::impl::detail::wrapfunctionintoruntimef></td></true,>	unctor <cl0::impl::detail::wrapfunctionintoruntimef 1.41e+12="" 1.75e+12="" 64.9%="" 72.5%<="" td=""><td>83.58</td></cl0::impl::detail::wrapfunctionintoruntimef>	83.58
<pre># 368 &gt;&gt; [I] cl0::guts::detail::_if_constexpr<true>::call<cl0::impl::make_boxe< pre=""></cl0::impl::make_boxe<></true></pre>	d_from_unboxed_functor <c10::impl::detail::wrapfun 1.41e+12="" 1.75e+12="" 64.9%="" 72.5%<="" td=""><td>83.58</td></c10::impl::detail::wrapfun>	83.58
<pre># 239 # [I] c10::impl::make_boxed_from_unboxed_functor<c10::impl::detail::wr< pre=""></c10::impl::detail::wr<></pre>	apFunctionIntoRuntimeFunctor_ <std::vector<at::ten 1.41e+12="" 1.75e+12="" 64.9%="" 72.5%<="" td=""><td>83.58</td></std::vector<at::ten>	83.58
<pre># 584 * [I] cl0::impl::call_functor_with_args_from_stack<cl0::impl::detail;< pre=""></cl0::impl::detail;<></pre>	::WrapFunctionIntoRuntimeFunctor_ <std::vector<at:: 1.41e+12="" 1.75e+12="" 64.9%="" 72.5%<="" td=""><td>State Constants</td></std::vector<at::>	State Constants
<pre># 511 w [I] c10::impl::call_functor_with_args_from_stack_<c10::impl::deta< pre=""></c10::impl::deta<></pre>	il::WrapFunctionIntoRuntimeFunctor_ <std::vector<at 1.41e+12="" 1.75e+12="" 64.9%="" 72.5%<="" td=""><td>83.58</td></std::vector<at>	83.58
<pre>4 499 » [I] cl0::impl::wrap_kernel_functor_unboxed_ccl0::impl::detail::W</pre>	rapEunctionIntoRuptimeEunctor estd::vectoreat::Te 1 75e+12 64 9% 1 41e+12 72 5%	83.58 83.58
		83.58 83.58 83.58
# 461 » [I] cl0::impl::detail::WrapFunctionIntoRuntimeFunctor_ <std::vec< td=""><td>tor<at::tensor, std::allocator<at::tensor=""> &gt; (*)( 1.75e+12 64.9% 1.41e+12 72.5%</at::tensor,></td><td>83.58 83.58 83.58 83.58</td></std::vec<>	tor <at::tensor, std::allocator<at::tensor=""> &gt; (*)( 1.75e+12 64.9% 1.41e+12 72.5%</at::tensor,>	83.58 83.58 83.58 83.58
<pre>4 461 &gt;&gt; [I] cl0::impl::detail::WrapFunctionIntoRuntimeFunctor_<std::vec 18="" 4="">&gt; scalar_born_cuda_autograd(at::Tensor const&amp;, at::Tensor const&amp;,</std::vec></pre>	Art:Tensor, std::allocator <t::tensor> (*)(         1.75e+12         64.9%         1.41e+12         72.5%           . at::Tensor const&amp;, at::Tensor const&amp;, at::Tensor         1.75e+12         64.9%         1.41e+12         72.5%</t::tensor>	83.58 83.58 83.58 83.58 83.58 83.58
<ul> <li>461 » [I] cl0::impl::detail::WrapFunctionIntoRuntimeFunctor_<std::vec< li=""> <li>4 8 » scalar_born_cuda_autograd(at::Tensor const&amp;, at::Tensor const&amp;,</li> <li>4 » torch::autograd::Function<scalarborncudafunction>::apply<scalarborncudafunction>::apply</scalarborncudafunction></scalarborncudafunction></li> </std::vec<></li></ul>	inspirationalized director active       inspiration       inspiration         inspirationalized director       inspiration       inspiration         inspirating director       inspiration	83.58 83.58 83.58 83.58 83.58 83.58 83.58
<ul> <li>461 » [I] c10::impl::detail::WrapFunctionIntoRuntimeFunctor_<std::vec< li=""> <li>4 8 » scalar_born_cuda_autograd(at::Tensor const&amp;, at::Tensor const&amp;,</li> <li>4 » torch::autograd::Function<scalarborncudafunction>::apply<scalarborncudafunction>::avtogradContex</scalarborncudafunction></scalarborncudafunction></li> <li>4 » ScalarBornCUDAFunction::forward(torch::autograd::AutogradContex</li> </std::vec<></li></ul>	http://drvatiles.outlast.outlas	83.58 83.58 83.58 83.58 83.58 83.58 83.58 83.58
<pre>4 461 % [I] c10::impl::detail::WrapFunctionIntoRuntimeFunctor_<std::vec 4 18 % scalar_born_cuda_autograd(at::Tensor const&amp;, at::Tensor const&amp;, 4 % torch::autograd::Function</std::vec </pre> scalarBornCUDAFunction>::apply <scalarb 4 % ScalarBornCUDAFunction::forward(torch::autograd::AutogradContex 4 % % ScalarBornCUDAFunction::forward(torch::Autograd::A</scalarb 	tor <at::tensor, std::allocator<at::tensor="">&gt; (*)(       1.75e+12       64.9%       1.41e+12       72.5%         introvention, at::Tensor const&amp;, at::Tensor con       1.75e+12       64.9%       1.41e+12       72.5%         introvention, at::Tensor const&amp;, at::Tensor con       1.75e+12       64.9%       1.41e+12       72.5%         introvention, at::Tensor const&amp;, at::Tensor con       1.75e+12       64.9%       1.41e+12       72.5%         introvention, at::Tensor const&amp;, at::Tensor con       1.75e+12       64.9%       1.41e+12       72.5%         introvention, at::Tensor const&amp;, at::Tensor const&amp;, at::Tensor       1.75e+12       64.9%       1.41e+12       72.5%         introvention, at::Tensor const&amp;, at::Tensor const&amp;, at::Tensor       1.75e+12       64.9%       1.41e+12       72.5%</at::tensor,>	83.58 83.58 83.58 83.58 83.58 83.58 83.58 83.58 83.58
<pre>461 &gt;&gt; [I] cl0::impl::detail::WrapFunctionIntoRuntimeFunctor_<std::vec 4 18 &gt;&gt; scalar_born_cuda_autograd(at::Tensor const6, at::Tensor const6, 4 &gt;&gt; torch::autograd::Function</std::vec </pre> ScalarBornCUDAFunction: autograd::AutogradContex 4 >> ScalarBornCUDAFunction::forward(torch::autograd::AutogradContex 4 >> ScalarBornCUDAFunction::forward(torch::autograd::AutogradContex 4 >> (anonymous namespace)::forward(torch[loat, 4>(float, float))	interactivitiensor, std::allocator       interactivitiensor, std::allocator         interactivitiensor, std::allocator       interactivitiensor         interactivitiensor       interactivitiensor<	83.58 83.58 83.58 83.58 83.58 83.58 83.58 83.58 83.60 83.60
<pre>461 &gt;&gt; [I] cl0::impl::detail::WrapFunctionIntoRuntimeFunctor_<std::vec 4 18 &gt;&gt; scalar_born_cuda_autograd(at::Tensor const6, at::Tensor const6, 4 &gt;&gt; torch::autograd::Function<scalarborncudafunction>::apply<scalarb 4 &gt;&gt; ScalarBornCUDAFunction::forward(torch::autograd::AutogradContex 4 &gt;&gt; ScalarBornCUDAFunction::forward(torch::autograd::AutogradContex 4 &gt;&gt; (anonymous namespace)::forward(torch:autograd::AutogradContex 4 &gt;&gt; (anonymous namespace)::forward_batch<float, 4="">(float*, float* 6 loop at [deepwave.so]: 0</float,></scalarb </scalarborncudafunction></std::vec </pre>	interference       interference       interference         interference       interference       in	83.58 83.58 83.58 83.58 83.58 83.58 83.58 83.60 83.61 83.61
<pre>461 &gt;&gt; [I] cl0::impl::detail::WrapFunctionIntoRuntimeFunctor_<std::vec 4 18 &gt;&gt; scalar_born_cuda_autograd(at::Tensor const6, at::Tensor const6, 4 &gt;&gt; torch::autograd::Function<scalarborncudafunction>::apply<scalarb 4 &gt;&gt; ScalarBornCUDAFunction::forward(torch::autograd::AutogradContex 4 &gt;&gt; ScalarBornCUDAFunction::forward(torch::autograd::AutogradContex 4 &gt;&gt; ScalarBornCUDAFunction::forward(torch::autograd::AutogradContex 4 &gt;&gt; (anonymous namespace)::forward(torch:autograd::AutogradContex 4 &gt;&gt; loop at [deepwave.so]: 0 4 &gt;&gt;</scalarb </scalarborncudafunction></std::vec </pre>	http://terratiitessor       1.75e+12       64.9%       1.41e+12       72.5%         , at::Tensor const&, at::Tensor const&, at::Tensor       1.75e+12       64.9%       1.41e+12       72.5%         ornCUDDAFunction, at::Tensor const&, at::Tensor const       1.75e+12       64.9%       1.41e+12       72.5%         t*, at::Tensor const&, at::Tensor const&, at::Tensor const       1.75e+12       64.9%       1.41e+12       72.5%         t*, at::Tensor const&, at::Tensor const&, at::Tensor       1.75e+12       64.9%       1.41e+12       72.5%         xt*, at::Tensor const&, at::Tensor const&, at::Tensor       1.75e+12       64.9%       1.41e+12       72.5%         st, float*, float*	83.58 83.58 83.58 83.58 83.58 83.58 83.58 83.60 83.61 83.61 89.50
<pre>461 &gt;&gt; [I] cl0:impl:idetal::WrapFunctionIntoRuntimeFunctor_<std::vec 418 &gt;&gt; scalar_born_cuda_autograd(at::Tensor const&amp;, at::Tensor const&amp;, 4 &gt;&gt; torch::autograd::Function</std::vec </pre> scalarBornCUDAFunction::isplyscalarB 4 >> ScalarBornCUDAFunction::forward(torch::autograd::AutogradContex 4 >> ScalarBornCUDAFunction::forward(torch::autograd::AutogradContex 4 >> ScalarBornCUDAFunction::forward(torch::autograd::AutogradContex 4 >> loop at [deepwave.so]: 0 4 >>dvice_stubZN47_GLOBAL_Nfe1446af_14_scalar_born_cu_ 4 >> gou kernel>	http://terratiitessor       1.75+12       64.9%       1.41e+12       72.5%         , at::Tensor const&, at::Tensor const&, at::Tensor       1.75+12       64.9%       1.41e+12       72.5%         ornCUDDAFunction, at::Tensor const&, at::Tensor const       1.75+12       64.9%       1.41e+12       72.5%         t*, at::Tensor const&, at::Tensor const&, at::Tensor const       1.75+12       64.9%       1.41e+12       72.5%         t*, at::Tensor const&, at::Tensor const&, at::Tensor const&, at::Tensor       1.75e+12       64.9%       1.41e+12       72.5%         t*, at::Tensor const&, at::Tensor const&, at::Tensor const&, at::Tensor       1.75e+12       64.9%       1.41e+12       72.5%         st:       float*, float*, float*, float*, float*, float*,       1.75e+12       64.9%       1.41e+12       72.5%         r, float*, float*, float*, float*, float*,       1.75e+12       64.9%       1.41e+12       72.5%         r, float*, float*, float*, float*, float*,       1.75e+12       64.9%       1.41e+12       72.5%         r, float*, float*, float*, float*, float*,       1.75e+12       64.9%       1.41e+12       72.5%         r, float*, float*, float*, float*,       1.75e+12       64.9%       1.41e+12       72.5%         r, float*, float*, float*, float*, float*,       1.44e+12	83.58 83.58 83.58 83.58 83.58 83.58 83.60 83.61 83.61 89.50 89.50
<pre>461 % [I] cl0::impl::detail::WrapFunctionIntoRuntimeFunctor_<std::wer< td=""><td>http://ternative.com/ternat</td><td>83.58 83.58 83.58 83.58 83.58 83.58 83.58 83.60 83.61 83.61 89.50 89.50</td></std::wer<></pre>	http://ternative.com/ternat	83.58 83.58 83.58 83.58 83.58 83.58 83.58 83.60 83.61 83.61 89.50 89.50
<pre>461 % [I] cl0::impl::detail::WrapFunctionIntoRuntimeFunctor_<std::vec 418 % scalar_born_cuda_autograd(dt::Tensor const6, at::Tensor const6 4 » torch::autograd::Function</std::vec </pre> ScalarBornCUDAFunction::autograd::AutogradContex 4 » ScalarBornCUDAFunction::forward(torch::autograd::AutogradContex 4 » ScalarBornCUDAFunction::forward(torch::autograd::AutogradContex 4 » (anonymous namespace)::forward_batch <float, 4="">(float*, float* 4 loop at [deepwave.so]: 0 4 »device_stub_ZN47_GLOBAL_N_fel446af_14_scalar_born_cu_ 4 » <gpu kernel=""> 4 % (anonymous namespace)::forward_kernel<float, 4,="" 4<br="" false,="">scalar_born.cu: 264</float,></gpu></float,>	http://ternsor.istdi:allocator       1/15:11       0/15:12       0/15:12       1/16:12       72.55         . at::Tensor const&, at::Tensor const&, at::Tensor con       1.75:12       64.95       1.41:12       72.55         . at::Tensor const&, at::Tensor const&, at::Tensor con       1.75:12       64.95       1.41:12       72.55         **, at::Tensor const&, at::Tensor const&, at::Tensor con       1.75:12       64.95       1.41:12       72.55         **, at::Tensor const&, float       1.75:12       64.95       1.41:12       72.55         . float*, float	83.58 83.58 83.58 83.58 83.58 83.58 83.58 83.60 83.61 83.61 83.61 89.50 89.50
<pre>461 * [T] cl0::impl::detail::WrapFunctionIntoRvmtimeFunctor_<std::wer< td=""><td>http://terratiilensor.jetu:/fensor&gt;&gt;(*)(       1.75e+12       64.9%       1.41e+12       72.5%         . at::Tensor const&amp;, at::Tensor const&amp;, at::Tensor       1.75e+12       64.9%       1.41e+12       72.5%         . at::Tensor const&amp;, at::Tensor const&amp;, at::Tensor       1.75e+12       64.9%       1.41e+12       72.5%         . at::Tensor const&amp;, at::Tensor const&amp;, at::Tensor       1.75e+12       64.9%       1.41e+12       72.5%         . at::Tensor const&amp;, at::Tensor const&amp;, at::Tensor       1.75e+12       64.9%       1.41e+12       72.5%         . xt*, at::Tensor const&amp;, at::Tensor const&amp;, at::Tensor       1.75e+12       64.9%       1.41e+12       72.5%         . float*, float       1.44e+12       53.3%       1.16e+12       59.3%         false&gt;(float const*, float*, float const*, float c       1.44e+12       53.3%       1.16e+12       59.3%       8.89e+11       59.7         3.63e+11       13.5%       3.12e+11       16.0%       2.65e+11       17.8%</td><td>83.58 83.58 83.58 83.58 83.58 83.58 83.58 83.60 83.61 83.61 83.61 89.50 89.50</td></std::wer<></pre>	http://terratiilensor.jetu:/fensor>>(*)(       1.75e+12       64.9%       1.41e+12       72.5%         . at::Tensor const&, at::Tensor const&, at::Tensor       1.75e+12       64.9%       1.41e+12       72.5%         . at::Tensor const&, at::Tensor const&, at::Tensor       1.75e+12       64.9%       1.41e+12       72.5%         . at::Tensor const&, at::Tensor const&, at::Tensor       1.75e+12       64.9%       1.41e+12       72.5%         . at::Tensor const&, at::Tensor const&, at::Tensor       1.75e+12       64.9%       1.41e+12       72.5%         . xt*, at::Tensor const&, at::Tensor const&, at::Tensor       1.75e+12       64.9%       1.41e+12       72.5%         . float*, float       1.44e+12       53.3%       1.16e+12       59.3%         false>(float const*, float*, float const*, float c       1.44e+12       53.3%       1.16e+12       59.3%       8.89e+11       59.7         3.63e+11       13.5%       3.12e+11       16.0%       2.65e+11       17.8%	83.58 83.58 83.58 83.58 83.58 83.58 83.58 83.60 83.61 83.61 83.61 89.50 89.50



extremecomputingtraining.anl.gov



## **Case Study: Quicksilver**

- Proxy application that solves a simplified dynamic Monte Carlo particle transport problem
  - Attempts to replicate memory access patterns, communication patterns, and branching or divergence of LLNL's Mercury for problems using multigroup cross sections
- Parallelization: MPI, OpenMP, and CUDA
- Performance Issues
  - load imbalance (for canned example)
  - latency bound table look-ups
  - a highly branchy/divergent code path
  - poor vectorization potential





### **Quicksilver: Detailed Analysis within a Kernel using PC Sampling**

hpcview	hpcviewer					
File Vi	ile View Filter Help					
Profil	Profile: qs					- 6
main.cc	ain.cc CollisionEvent.cc X					
69 70 71 72 73 75 75 76 77 78 79 80 81 82 83 84 85	<pre>int uniqueNumber = monteCarlo-&gt;_materialDatabase-&gt;_mat[globalMatIndex]iso[isoIndex]gid; int numReacts = monteCarlo-&gt;_nuclearData-&gt;getNumberReactions(uniqueNumber); for (int reactIndex = 0; reactIndex &lt; numReacts; reactIndex++) { currentCrossSection -= macroscopicCrossSection(monteCarlo, reactIndex, mc_particle.doma) isoIndex, mc_particle.energy_group); if (currentCrossSection &lt; 0) { selectedIso = isoIndex; selectedIso = isoIndex; selectedReact = reactIndex; break; } } } g_sassert(selectedIso != -1); } } </pre>	n, mc_particle.coll,				
Top-down	op-down view Bottom-up view Flat view					
<b>1</b> 5	1 중 6 f× 중 [X] 프 A* A III ▼ 22					
Scope	GINS: Sum	(I) - GINS: Sum (E)	GINS:STL_ANY: Sum (I)	GINS:STL_ANY: Sum (E)	GINS:STL_IFET: Sum (I)	GINS:STL_IFET: Sum (E) GINS:STL_IDEP:

Scope		(I) = GI			GINS:STL_ANY: Sum					n (1)	GINS:STL_IFET: Sum		GINS:STL_IDER:
<pre>4 14 » [1] cudaLaunchKernel<char></char></pre>	1.30e+11	100.0%			1.19e+11	100.0	*		5.2/e+09	100.0%			9.34e+
▲ 211 » cudaLaunchKernel [qs]	1.30e+11	100.0%			1.19e+11	100.0	\$		5.27e+09	100.0%			9.34e+
⊿ » <gpu kernel=""></gpu>	1.30e+11	100.0%			1.19e+11	100.0	8		5.27e+09	100.0%			9.34e+
A >> CycleTrackingKernel(MonteCarlo*, int, ParticleVault*, ParticleVau	1.30e+11	100.0% 4.	.08e+07	0.0%	1.19e+11	100.0	% 3.62e+07	0.0%	5.27e+09	100.0%	2.11e+07	0.4%	9.34e+
# 132 » CycleTrackingGuts(MonteCarlo*, int, ParticleVault*, Particle	1.30e+11	100.0% 9.	.03e+09	7.0%	1.19e+11	100.0	% 9.01e+09	7.6%	5.24e+09	99.5%	8.98e+06	0.2%	9.32e+
▲ 26 » [I] CycleTrackingFunction(MonteCarlo*, MC_Particle&, int, P	8.36e+10	64.4% 4.	.12e+08	0.3%	7.25e+10	61.1	% 3.65e+08	0.3%	5.21e+09	98.9%	1.02e+08	1.9%	9.25e4
▲ loop at CycleTracking.cc: 118	8.35e+10	64.3% 3.	.76e+08	0.3%	7.25e+10	61.1	% 3.34e+08	0.3%	5.21e+09	98.8%	9.90e+07	1.9%	9.24e+
▲ 63 ∞ CollisionEvent(MonteCarlo*, MC_ParticleS, unsigned int) [	5.20e+10	40.1% 4.	.99e+09	3.8%	4.44e+10	37.4	% 4.02e+09	3.4%	3.85e+09	73.1%	4.89e+08	9.3%	6.37e4
▲ loop at CollisionEvent.cc: 67	4.09e+10	31.5% 8.	.15e+08	0.6%	3,42e+10	28.8	% 6.54e+08	0.6%	3.54e+09	67.1%	1.27e+08	2.4%	5.67e+
loop at CollisionEvent.cc: 71	3.85e+10	29.6% 2.	.70e+09	2.1%	3.22e+10	27.1	% 2.06e+09	1.7%	3.27e+09	62.0%	2.28e+08	4.3%	5.33e+
# 73 • macroscopicCrossSection(MonteCarlo*, int, int, i	3.58e+10	27.5% 1.	.22e+10	9.4%	3.01e+10	25.4	\$ 9.85e+09	8.3%	3.04e+09	57.74	1.79e+09	33.9%	4.600
4 41 w NuclearData::getReactionCrossSection(unsigned int, u	2.09e+10	16.1% 1.	.09e+10	8.4%	1.79e+10	15.1	% 9.42e+09	7.9%	1.26e+09	23.8%	6.68e+08	12.7%	2.19e+
253 » [I] NuclearDataReaction::getCrossSection(unsigned	6.89e+09	5.3% 3.	.77e+09	2.9%	5.86e+09	4.9	% 3.32e+09	2.8%	2.25e+08	4.39	8.24e+07	1.6%	8.86e+
NuclearData.cc: 253	6.28e+09	4.8% 6.	.28e+09	4.8%	5,66e+09	4.8	% 5.66e+09	4.8%	4.76e+08	9.0%	4.76e+08	9.0%	6.11e-
NuclearData.cc: 251	1.85e+09	1.4% 1.	.85e+09	1.4%	1.64e+09	1.4	% 1.64e+09	1.4%	8.12e+07	1.5%	8.12e+07	1.5%	2.47e+
NuclearData.cc: 248	1.61e+09	1.2% 1.	.61e+09	1.2%	1.18e+09	1.0	% 1.18e+09	1.0%	1.10e+08	2.1%	1.10e+08	2.1%	3.62e+
<pre>&gt; 252 » [I] qs_vector<nucleardataspecies>::operator[](int)</nucleardataspecies></pre>	1.29e+09	1.0% 1.	.29e+09	1.0%	1,14e+09	1.0	% 1.14e+09	1.0%	7.37e+04	0.0%	7.37e+04	0.0%	1.24e+
NuclearData.cc: 252	1.12e+09	0.9% 1.	.12e+09	0.9%	9.48e+08	0.8	% 9.48e+08	0.8%	3.44e+05	0.0%	3.44e+05	0.0%	2.50e+
252 » [I] qs_vector <nucleardatareaction>::size() const</nucleardatareaction>	9.41e+08	0.7% 9.	.41e+08	0.7%	8.17e+08	0.7	% 8.17e+08	0.7%					4.63e+
• 0F5	2 262-100	0.25.2	10-100	0.00	2 47++00	0.0	· · · · · · · · · · · · · · · · · · ·	0.01	1 43-+00	0.70	1 41++00	2.76	7.774

### Quicksilver: Detailed analysis within a Kernel using PC Sampling

Scope
▲ 14 » [1] cudaLaunchKernel <char></char>
✓ 211 » cudaLaunchKernel [qs]
⊿ » <gpu kernel=""></gpu>
A » CycleTrackingKernel(MonteCarlo*, int, ParticleVault*, ParticleVau
▲ 132 » CycleTrackingGuts(MonteCarlo*, int, ParticleVault*, Particle
▲ 26 » [I] CycleTrackingFunction(MonteCarlo*, MC_Particle&, int, P
▲ loop at CycleTracking.cc: 118
▲ 63 » CollisionEvent(MonteCarlo*, MC_Particle&, unsigned int) [
✓ loop at CollisionEvent.cc: 67
Ioop at CollisionEvent.cc: 71
▲ 73 ≫ macroscopicCrossSection(MonteCarlo*, int, int, i
▲ 41 » NuclearData::getReactionCrossSection(unsigned int, u
253 » [I] NuclearDataReaction::getCrossSection(unsigned
NuclearData.cc: 253
NuclearData.cc: 251
NuclearData.cc: 248
252 » [I] qs_vector <nucleardataspecies>::operator[](int)</nucleardataspecies>
NuclearData.cc: 252
<pre>&gt; 252 &gt;&gt; [I] qs_vector<nucleardatareaction>::size() const</nucleardatareaction></pre>

## Analysis of PeleC using PC Sampling on an NVIDIA GPU

000	hpcvlewer
Profile: PeleC3d.gnu.TPROF.CUDA.ex	
<pre>reactor.cpp 33 Metric properties  438 UserData udata = static_cast<arkodeuserdata+>(user_data); 439 udata-&gt;dt_save = t; 440 441 #ifdef AMREX_USE_GPU 441 #ifdef AMREX_USE_GPU 442 const auto ec = amrex::Gpu::ExecutionConfig(udata-&gt;ncells_d); 443 amrex::launch_global&lt;&lt;</arkodeuserdata+></pre>	Cause: passed udata structure pointer to lambda capture
<pre>444</pre>	Improvement: pass udata components as scalars https://github.com/AMReX-Combustion/PelePhysics/pull/192 4% speedup on PeleC PMF drm19 test case
全 ♣ 🍐 f× 愛 🕅 A* ↔ 비· 교	

Scope		GINS.[0,0] (I)	*	GINS.[0,0] (E)	(	INS:STL_ANY.(0,0)	(0)	GINS:STL_ANY.[0,0] (E)	GINS:STL_GMEM.[0,0]	(1)	GINS:STL GMEM.[0.0] (	E)
loop at AMReX_Amr.cpp: 2061		1.24e+13	88.6%			1.05e+13	88.7%		5.58e+12	89.3%		
Boundary 2062: amrex::Amr::timeStep(int, double, int, int, double)		1.24e+13	88.6%			1.05e+13	88.7%		5.58e+12	89.3%		
2015: PeleC::advance(double, double, int, int)	CPU	1.24e+13	88.5%			1.05e+13	88.6%		5.57e+12	89.2%		
A B 36: PeleC::do_sdc_advance(double, double, int, int)		1.24e+13	88.5%			1.05c+13	88.6%		5.57e+12	89.2%		
Ioop at Advance.cpp: 302	context	1.24e+13	88.4%			1.05e+13	88.5%		5.57e+12	89.1%		
A B 308: PeleC::do_sdc_iteration(double, double, int, int, int, int)	CONTOXE	1.24e+13	88.4%			1.05e+13	88.5%		5.57e+12	89.1%		
S61: PeleC::react_state(double, double, bool, amrex::MultiFab*)		9.61e+12	68.5%			8.29e+12	70.0%		4.17e+12	66.8%		
loop at React.cpp: 109		9.43e+12	67.2%			8.14e+12	68.7%		4.06e+12	65.0%		
✓ ₽ 210: react(amrex::Box const&, amrex::Array4 <double> const&amp;,</double>	amrex::Array4 <double> cons</double>	9.39e+12	66.9%			8.10e+12	68.4%		4.03e+12	64.5%		
▲ 卧 234: arkEvolve [libsundials_arkode.so.4.7.0]		9.28e+12	66.2%			8.00e+12	67.6%		3.94e+12	63.1%		
erkStep_TakeStep [libsundials_arkode.so.4.7.0]		7.16e+12	51.1%			6.19e+12	52.3%		3.05e+12	48.9%		
FRHS(double, _generic_N_Vector*, _generic_N_Vector*, vo	id*)	6.27e+12	44.7%			5.49e+12	46.3%		2.48e+12	39.7%		
▲ ● 443: [I] amrex::launch_global<_nv_dl_wrapper_t<_nv_dl_t	ag <int (*)(double,="" _generic_n<="" th=""><td>6.27e+12</td><td>44.7%</td><td></td><td></td><td>5.49e+12</td><td>46.3%</td><td></td><td>2.48e+12</td><td>39.7%</td><td></td><td></td></int>	6.27e+12	44.7%			5.49e+12	46.3%		2.48e+12	39.7%		
A B 12: [I] _wrapper_device_stub_launch_global<_nv_dl_wrapper_device_stub_launch_global<_nv_dl_wrapper_device_stub_launch_global<_nv_dl_wrapper_device_stub_launch_global<_nv_dl_wrapper_device_stub_launch_global<_nv_dl_wrapper_device_stub_launch_global<_nv_dl_wrapper_device_stub_launch_global<_nv_dl_wrapper_device_stub_launch_global<_nv_dl_wrapper_device_stub_launch_global<_nv_dl_wrapper_device_stub_launch_global<_nv_dl_wrapper_device_stub_launch_global<_nv_dl_wrapper_device_stub_launch_global<_nv_dl_wrapper_device_stub_launch_global<_nv_dl_wrapper_device_stub_launch_global<_nv_dl_wrapper_device_stub_launch_global<_nv_dl_wrapper_device_stub_launch_global<_nv_dl_wrapper_device_stub_launch_global<_nv_dl_wrapper_device_stub_launch_global<_nv_dl_wrapper_device_stub_launch_global	apper_t<_nv_dl_tag <int (*)(do<="" th=""><td>6.27e+12</td><td>44.7%</td><td></td><td></td><td>5.49e+12</td><td>46.3%</td><td></td><td>2.48e+12</td><td>39.7%</td><td></td><td></td></int>	6.27e+12	44.7%			5.49e+12	46.3%		2.48e+12	39.7%		
A B 26: [I] _device_stub_ZN5amrex13launch_globallZ6cF_	RHSdP17_generic_N_VectorS2	6.27e+12	44.7%			5.49e+12	46.3%		2.48e+12	39.7%		
✓ ■ 24: [I] cudaLaunchKernel <char></char>		6.27e+12	44.7%			5.49e+12	46.3%	9.4% GPU stalls	2.48e+12	39.7%	mostly memo	ory
✓ I 211: cudaLaunchKernel [PeleC3d.gnu.TPROF.CUDA.e	x]	6.27e+12	44.7%			5.49e+12	46.3%	outside the loop	2.48e+12	39.7%	stalls	
CDII 4 III <gpu kernel=""></gpu>		6.27e+12	44.7%			5.49e+12	46.3%		2.48e+12	39.7%		
OF 0 # # amrex::launch_global <cf_rhs(double, _generic_n_<="" p=""></cf_rhs(double,>	Vector*, _generic_N_Vector*, v	6.27e+12	44.7%	1.75e+10	0.1%	5.49e+12	46.3%	1.70e+10 0.1%	2.48e+12	39.7%		
contovt / \$12: [I] cF_RHS(double, _generic_N_Vector, _generic_N_Vector	ric_N_Vector*, vold*)::{lambda	6.25e+12	44.6%	1.17e+12	8.3%	5.47e+12	46.2%	1.16e+12 9.8%	2.48e+12	39.7%	1.14e+12	18.2%
CONTEAL I loop at reactor.cpp: 446		5.14e+12	36.6%	5.35e+10	0.4%	4.36e+12	36.8%	4.62e+10 0.4%	1.38e+12	22.0%	3.29e+10	0.5%
reactor.cop: 446		1.11e+12	7.9%	1.11e+12	7,9%	1.11e+12	26.08	1.11e+12 9.4%	1,100+12	17.78	1.10e+12	17.78
AMReX_GpuLaunchGlobal.H: 12		1.75e+10	0.1%	1.75e+10	0.1%	1.70e+10	0.1%	1.70e+10 0.1%				
			11							11		

# **Key Metrics for GPU Kernels**

- GPUOP: GPU operation time (kernel launch, copies, etc.)
- GXCOPY:\* GPU copies of various kinds
- GKER: GPU kernel time
- GKER:FGP\_ACT: fine grain parallelism actual (active warps per SM)
- GKER:FGP\_MAX: maximum possible fine-grain parallelism (max warps per SM)
- GKER:BLK\_THR: threads per block
- GKER:BLK\_SM: block shared memory
- GKER:OCC\_THR: theoretical thread occupancy





## **Metrics for GPU Kernels with PC Samples**

- GINS: GPU instructions
- GINS:STL\_ANY: GPU instruction stalls for any reason
- GINS:STL\_IFET: GPU instruction stalls for instruction fetch
- GINS:STL\_GMEM: GPU instruction stalls for global memory
- GINS:STL\_CMEM: GPU instruction stalls for constant memory
- GINS:STL\_IDEP: GPU instruction stalls for instruction dependences
- GINS:STL\_PIPE: GPU instruction pipeline stalls
- GINS:STL\_MTHR: GPU instruction stalls for memory throttling
- GSAMP:EXP: expected number of samples
- GSAMP:TOT: total number of samples recorded
- GSAMP:UTIL: GPU utilization = (PC samples expected) / (PC samples total)





0mm / -+

Partie imp m hace imp										* 1
That provid MPL Tand HEALTON (1942) (1) That had pa	I grant have band to A	TIME CHART	10 × 10							
		Sortes	t plot graph MPL Sen	C REALTIME (	(h (see					
Several sectors and the sectors of										
182										1
										/
102-									-	1
5								_		
					-					
				-						
		-								-
2000 4000 4000 4000 1000 1000	14000 Incolo 18000 2000	0 22000 240	NC 24400 28000 3000 3000	0 34000 Notes 245	00 a0000 40		DEE 13930 1403			4000
			RavA in S	arted Order						
the draw way Britten in the data (Balance)										
1 · · · · · · · · · · · · · · · · · · ·	* 21									
The second s	Contraction of	COLUMN T	MULTION SHOT	1509400001 Be	een des	In the local section of	POLICE DISTU	1000	CONTRACTOR OF	-
Comment Lypique Vatures	Anna 1. sta-te	110.01	3,484-01 (0.00)	L-414+81 1	e en la cale	1,45e+01 100.00	ALCON [10011	100.05	2.344-57	
Comment Approprie Metrics     compare metric	0.00 (0410-120300) 3.00 1.45e-10 1.45e-10	100.00	3,48e+05 109.05	1.45e-00 1	1181) (380 147, 8%	1.45e+09.109.05	144000 (1997) 14-906,1	200.00	2.30e-61	
Experiment Approprie Memore     error program moto     error prod	1.01 (1000) (1000) 1.01 1.45e-5 1.45e-5 1.45e-5	100.00	3,456+05 109.05	1.45e+00 1 1.45e+00 1 1.45e+00 1	11 (12) (14) 147, 524 147, 524 147, 524	1,454-09 109,00	1. Ma-87 2. Ma-87 2. Ma-87 2. Ma-87	200.0%	2,34e-57	114.
Comment Approprie Matters     Comparement     Second Bridge     Atter (Addref S.N.) Input Hell (Milleninger an II)	0.00 (04000000000) 0.00 0.400-00 0.400-0 0.400-0 0.400-0 0.150-0	100.00	3,480-05 100.05	1.45e+00 1 1.45e+00 1 1.45e+00 1 1.45e+00 1	10 112 10 10 10 10 . 10 1 10 . 10 1 10 . 10 1 10 . 10 1	1,456-09 109.00	2.30s-87 2.30s-87 2.30s-87 2.30s-87 2.30s-87 3.50s-87	200.00 200.00 200.00 200.00 55.20	2,388=67 1,959-62 5,688-63	-
Contrant Approprie Network     Contrant Approprie Network     seam (Impl     with a CAMPERS' Nith Implif (Impl) (Interrupt in of (Impl)     a to CAMPERS' Implif events, community) (Impl)	0.00 0.0000000000000000000000000000000	5 110.00 110.00 110.00 110.00 110.00 110.00 10.000 10.000 10.000 10.000 100 1	3,40e+05 105.05	1.45e+00 1 1.45e+00 1 1.45e+00 1 1.45e+00 1 1.45e+00 1 1.45e+00	10 102 0100 100 , 876 100 , 876 100 , 876 100 , 876 100 , 876	1,45e+09 100,00	1,30=51 1,30=51 1,30=51 2,30=51 2,50=51 1,50=51	200.00 200.00 200.00 200.00 51.20 54.00	2,30+41 2,30+41 1,90+41 5,40+41 2,42+41	100 - 100 -
Coperment Approprie Methos     coperment Approprie Methos     security of the security of	2.154-10 2.491 2.494-10 2.491 2.494-10 2.494-10 2.154-10 2.154-10 1.414-10 1.414-10	100.00 100.000 100.000 100.000 100.00000000	10012030475am3033	1.45e+00 1 1.45e+00 1 1.45e+00 1 1.45e+00 1 1.45e+00 1.45e+00 1.45e+00	11 (11) (12) 10, 15) 10, 15) 10, 15) 10, 15) 10, 15) 10, 15) 10, 15)	1,45e+09 109,05	1.30=901 2.30=97 2.30=97 2.50=97 2.50=97 2.10=97 2.50=97 2.50=97	200.00 200.00 200.00 91.20 94.00 52.00	2,38+42 1,58+42 5,48+42 2,42+41 4,42+42	100. 2. 2. 2.
Comment Approprie Metros     Company moto     wongone moto     wongone moto     wongone moto     wongone moto     wongone moto     wongone     wo	2. 2. 2. 2. 2. 2. 2. 2. 2. 2. 2. 2. 2. 2	1 200.00 200.00 1 200.00 5 82.20 6 82.20 6 82.20 5 82.40	3,460-05 109.05	1.45++00 1 1.45++00 1 1.45++00 1 1.45++00 1 1.45++00 1.45++00 1.45++00 1.44++00	NI 101 ELE NI 101 NI 10	504(2000) 564 (1) 0 1,456-09 309,45	1,300-11 1,300-17 2,300-17 2,300-17 2,300-17 2,300-17 2,310-17 2,310-17	200.00 200.00 200.00 95.00 94.00 92.65	2,38+42 1,59+42 5,48+42 2,42+42 4,42+42 2,42+42 2,40+42 2,40+42	100. 1. 1. 1. 1.
Counterest-Approprie Datus     Counterest-Approprie Datus     senselling     sets = LAMMPE_NES input Heg) Differences in 18         « LAMMPE_NES input Heg) Differences in 18         « LAMMPE_NES the summarial data (Init**)         « LAMMPE_NES input Hegical (Init**)         « LAMMPE_NES (Init**)         « LAMMPE_NES (Init**)         « LAMMPE_NES         » LAMMPE_NES         » LAMMPE_NES         » LAMMPE_NES         »	Data         Data <thdata< th="">         Data         Data         <thd< td=""><td>1 100.00 1 100.</td><td>3,488465 188.00</td><td>1.45e+09 1 1.45e+09 1 1.45e+09 1 1.45e+09 1 1.45e+09 1 1.45e+09 1 1.44e+09 1</td><td>127 GAN 10, 85 20, 85 20, 95 20, 95 20, 95 20, 95 20, 25</td><td>2017(2018) Sue (2) 10</td><td>2,306-87 2,306-87 2,306-87 2,206-87 2,206-87 2,206-87 2,306-87 3,316-87 8,656-88</td><td>200.00 200.00 200.00 95.00 94.00 94.00 92.00 21.90</td><td>1,399-41 1,399-41 5,499-41 2,429-41 0,429-41 2,499-41 2,499-42 1,499-42</td><td>100 100 100 100 100 100 100 100 100 100</td></thd<></thdata<>	1 100.00 1 100.	3,488465 188.00	1.45e+09 1 1.45e+09 1 1.45e+09 1 1.45e+09 1 1.45e+09 1 1.45e+09 1 1.44e+09 1	127 GAN 10, 85 20, 85 20, 95 20, 95 20, 95 20, 95 20, 25	2017(2018) Sue (2) 10	2,306-87 2,306-87 2,306-87 2,206-87 2,206-87 2,206-87 2,306-87 3,316-87 8,656-88	200.00 200.00 200.00 95.00 94.00 94.00 92.00 21.90	1,399-41 1,399-41 5,499-41 2,429-41 0,429-41 2,499-41 2,499-42 1,499-42	100 100 100 100 100 100 100 100 100 100
Courses Aggregate Metros     Courses Aggregate Metros     - engage mole     - engage mole     - endingi     - endowed aggregate Aggregate Metros     - endowed aggregate Aggregate Aggregate     - endowed aggregate Aggregate Aggregate     - endowed Aggregate Aggregate Aggregate     - endowed     - endowed Aggregate     - endowed     - endowe	2. (2010) (2010) 3. 4% (2010) 3. 5% (2010)	100.00 100000000	3,486+06 109.00	1.45+49 1 1.45+49 1 1.45+49 1 1.45+49 1 1.45+49 1 1.45+49 1 1.44+19 1 1.44+19 1	10 000 10,00000 10,0000	004420487 New (2111) 1.45e+88 100.4%	2. Mar 82 2. Mar 87 2. Mar	200.00 200.00 200.00 94.00 94.00 92.00 92.00 21.00	2,3000000000000000000000000000000000000	100 100 100 100 100 100 100 100 100 100
Experiment Aggregation Darkson     Experiment Aggregation Darkson     experiment model     a second length     a second l	Cold (1)         Cold (1)           0.4%         0.4%         0.4%           0.4%         0.4%         0.4%           0.4%         0.4%         0.4%           0.4%         0.4%         0.4%           0.4%         0.4%         0.4%           0.4%         0.4%         0.4%           0.4%         0.4%         0.4%           0.4%         0.4%         0.4%           0.4%         0.4%         0.4%           0.4%         0.4%         0.4%           0.4%         0.4%         0.4%	5 210, 99 5 210, 99 5 210, 99 5 52, 29 5 225, 95 5 23, 45	3,456-68,169.05	1.45+00 1 1.45+00 1 1.45+00 1 1.45+00 1 1.45+00 1 1.45+00 1 1.45+00 1 1.45+00 1	100 0100 100,0000 100,0000 100,0000 100,0000 100,0000 100,0000 100,0000 100,00000000	1.45e+09 100,05	2. 300-07 2. 300-07 2. 300-07 2. 300-07 2. 300-07 2. 300-07 3. 300-07 5. 450-08 5. 450-08 5. 450-08 5. 450-08 5. 450-08 5. 450-08	209.0% 209.0% 209.0% 94.0% 94.0% 92.6% 21.9%	2, 344-57 2, 344-57 2, 344-57 2, 544-52 2, 544-5 2, 544-52, 544-5 2, 544-5	100 100 100 100 100 100 100 100 100 100
Coperand Aggregate Outros     Coperand Aggregate Outros     Coperand Aggregate Outros     Sector Secto	100         CMADE 1 FOXMULT           8         8         8         8         8         8           8         45         9         8         45         9         8         6         9         2         5         9         9         8         6         9         2         5         9         9         1         6         6         9         2         5         9         1         6         6         9         2         5         9         1         6         6         6         1         1         6         1         1         6         1         1         6         1	100 000 100 00 100 0	3,464-66 105.05	1.45e+09 1 1.45e+09 1 1.45e+09 1 1.45e+09 1 1.45e+09 1 1.45e+09 1 1.44e+09 1	11 11 12 12 12 12 12 12 12 12 12 12 12 1	1.45e+09 100.05	2.30x-07 2.30x-07 2.30x-07 2.30x-07 2.50x-07 2.50x-07 2.30x-07 2.30x-07 3.30x-07 5.85x-00 1.45x-00 7.11x-07 5.11x-	209.09 209.09 209.09 95.29 94.09 92.09 92.09 21.99 7.09 9.09	2,344-87 2,344-87 2,344-87 2,744-82 2,7	
Comparison di Aggingen Marcos     Comparison di Aggingen Marcos     Comparison multi     Service di India	2 228 (2460-8 240-001 8.46 2.460-0 8.460-0 2.55-0 2.55-0 2.55-0 1.650-0 5.350-0 5.350-0	1 100 100 1 100 1 100 100 1 10	3,456-65 105.05	1.45e+09 1 1.45e+09 1 1.45e+09 1 1.45e+09 1 1.45e+09 1 1.45e+09 1 1.44e+09 1	M. 04 M. 05 M. 05	1.45e+09 100.05	23.000 20002 2.000-07 2.000-07 2.000-07 2.000-07 2.000-07 2.000-07 2.000-07 2.000-07 0.000-07 0.000-07 0.000-07 0.000-07 0.000-07 0.000-07 0.000-07	200.00 200.00 200.00 94.00 94.00 93.00 92.00 92.00 7.00 8.00 8.00 8.00	23000000000000000000000000000000000000	
Common Laparent Button     Comparent Laparent Laparent     Comparent Laparent     Comparen	200 Edd 11 220001 3.450 3.450 - 50 8.450 - 50 8.450 - 50 2.250 - 50 2.250 - 50 3.50 - 50 5.30 - 50	1 100 00 1 1000	3,46e+65 109.05	1.45+00 1 1.45+00 1 1.45+00 1 1.45+00 1 1.45+00 1 1.45+00 1.45+00 1.45+00 1.44+00 0	M. 04 M. 05 M. 05	1,45e+09 100,4%	23.000-00 2.300-07 2.300-07 2.500-07 2.500-07 2.500-07 2.500-07 2.500-07 2.500-07 0.050-00 0.050-00 0.100-00 0.100-00 2.410-02	200.00 200.00 200.00 94.00 94.00 92.00 92.00 7.00 8.00 8.00 9.00	1. 000-01 1. 000-000-000-000 1. 000-000-0000-000-000-000-000-000	
Comment Appropriate Detects	22 Contra 220001 3. 490 S. 4900-0 8. 4900-0 8. 4900-0 8. 4900-0 8. 4900-0 8. 4900-0 9. 500-0 5. 300-0 5. 30	5 100,00 5 100,000 5 100,00	3,464-66 189.45	1.45e+00 1 1.45e+00 1 1.45e+00 1 1.45e+00 1 1.45e+00 1 1.45e+00 1 1.45e+00 1 1.45e+00 1 1.45e+00 1 1.45e+00 1	10 10 10 10 10 10 10 10 10 10 10 10 10 1	1.45e-09 100.00	23.0000 200000 2.000007 2.000007 2.000007 2.000007 2.000007 2.000007 2.000007 3.0000000000	200.05 200.05 201.05 31.25 34.95 32.05 32.05 7.05 7.05 8.05 8.05 20.05	1. 300 - 62 1. 30	
(generate A gargets bette)     • grapper nois     • grapper     • grapper nois     • grapper nois     •	22 Castro 220001 a. 48 a. 480-5 a. 480-5 a. 480-5 a. 480-5 a. 480-5 a. 480-5 a. 580-5 b. 500-5 b. 500-5 b. 500-5 b. 500-5 b. 480-5 b.	6 100.49 1 200.49 1 200.	3,464-00 100.00	1.45+49 1 1.45+49 1	10 10 10 10 10 10 10 10 10 10 10 10 10 1	1So-09 100,00	Allow [res 1 2.366-87 2.366-87 2.366-87 2.366-87 2.366-87 2.366-87 3.366-86 3.	200.05 200.05 200.05 94.05 94.95 92.05 92.05 7.05 8.05 8.05 8.05 8.05 8.05 8.05 8.05 8	1. 398-92 2. 398-92 2. 466-92 2. 466-92 2. 458-92 2. 458-92 2. 458-92 2. 458-92 2. 458-92 2. 458-92 1. 518-92 1. 518-92	
Image: Section 2016	22 Control 220001 3. 490 3. 490 2. 490 2. 550 3. 300 5. 400 5. 300 5.	5 100.49 5 100.49 5 100.39 6 100.39 6 82.39 6 82.39 6 82.39 5 125.49 5 125.49	3,464-09 106.09	1.45e+00 1 1.45e+00 1 2.45e+00 1 2.45e+00 1	2011 0 000 100,000	1,-25+-09 100,-00	Allow (1997) J. Meest J. Meest J. Meest J. Seet J. S	200.00 200.00 200.00 95.00 95.00 92.00 92.00 7.00 9.00 9.00 9.00 9.00 9.00 14.00	2.300-012 2.300-02 2.400-02 2.400-02 2.420-02 2.420-02 2.420-02 3.400-02 3.400-02 1.400	
Contrast Appropria Detects     Contrast Appropria Detects     Contrast Appropria     Contexet Appropria     Contrast Approprime Appropria     Contrast Appr	5. Elevel 5. Elevel 5. Elevel 5. Stevel 5. Stevel 5. Stevel 5. Stevel 5. Stevel 5. Stevel	5 148.49 6 120.49 6 120.49 6 12.17 6 12.17 6 12.17 5 12.49 5 12.49	3,464-05,164,000	1.45+40 1 1.45+40 1 1.45+40 1 1.45+40 1 1.45+40 1 1.45+40 1 1.45+40 1 1.45+40 1 1.45+40 1 1.44+10 1	2010 0 000 100,000	1,-See09 100,00	2.500 (1997) 2.500-07 2.500-07 2.500-07 2.500-07 2.500-07 3.5000-07 3.5000-07 3.5000-07 3.5000-07 3.5000-07 3.5000-07 3.500-	200 . 00 200 . 00 200 . 00 94 . 00 93 . 00 93 . 00 93 . 00 7 . 00 9 . 00 9 . 00 9 . 00 20 . 00 14 . 00 14 . 00 14 . 00	1, 100-11, 100-12, 100	
(generate Agregate Matter	30 (1993) 2000 3.0% 3.0% 2.0% 2.0% 2.5% 2.5% 3.5% 3.5% 3.5% 5.3% 5.3% 5.5	5 280.00 6 200.00 6 200.00 6 42.00 6 42.00 5 220.00 5 23.46 5 25.46	3,46+46 100,07	1.45e-00 1.45e-00 1.45e-00 1.45e-00 1.45e-00 1.45e-00 1.44e-	21 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2	1,45+49 104,64	2. Mar 1997 2. Mar 19 2. Mar 19 3. Mar 19	200.00 200.00 200.00 94.00 92.00 92.00 92.00 92.00 9.00 9.00 9.	2,000-000 2,000-02 4,	100 100 100 100 100 100 100 100











Top-down view Bottom-up view Flat view

#### 🟦 🗄 🌜 🕼 🕼 🖪 🕅 🧮 A\* 🔤 🖬 👻

Scope		GMSET COUNT Sum (I)			GXCOPY COUNT: Sum (E)	REALTIME (sec): Sum (I)	REALTIME (sec) Sum	n (E)
<ul> <li>Experiment Aggregate Metrics</li> </ul>	9.01	3.46e+06 100.0%	3.46e+06 100.0%	1.45e+09 100.0%	1.45e+09 100.0%	2.30e+07 100.09	6 2.30e+07 10	.00.1
✓ <program root=""></program>		3.46e+06 100.0%	£	1.45e+09 100.0%		2.30e+07 100.09	6	
⊿ » main [Imp]		3.46e+06 100.0%	5	1.45e+09 100.0%		2.30e+07 100.09	6 1.00e-02	0.1
▲ 98 » LAMMPS_NS:Input:file() [liblammps.so.0]		2.15e+06 62.1%	4	1.45e+09 99.9%		2.19e+07 95.29	6 5.66e-02	0.1
A » LAMMPS_NS:Input:execute_command() [li		2.15e+06 62.1%	s	1.45e+09 99.9%		2.18e+07 94.8	¢ 2.02e-01	0.1
A » LAMMPS_NS:Run:command(int, char**)		1.01e+06 29.3%	\$	1.44e+09 99.4%		2.14e+07 92.99	6.02e-02	0.1
A » LAMMPS_NS: VerletKokkos: run(int) [libl	5	5.33e+05 15.4%	6	1.44e+09 99.2%		2.13e+07 92.6	\$ 2.66e+02	0.1
A » LAMMPS_NS:CommBrick::forward_c						5.03e+06 21.9	3.49e+02	0.1
MPI_Send [libmpi_gnu_91 so 12.0.0]						2.45e+06 10.6	4.75e+02	0.1
MPI_Wait[libmpi_gnu_91.so.12.0.0]						1.81e+06 7.9	3.60e+01	0.1
» LAMMPS_NS AtomVec pack_comm						7.11e+05 3.19	6 7.11e+05	3.
MPI_Irecv [libmpi_gnu_91.so.12.0.0]						6.16e+04 0.3	6 1.91e+02	0.1
liblammps.so.0@0x425b2c3						2.61e-02 0.09	6 2.61e-02	0.1
LAMMPS_NS_ModifyKokkos_final_inte				2.85e+08 19.6%		4.72e+06 20.5	6 1.13e+02	0.1
LAMMPS_NS AtomVecAtomicKokkos				2.46e+08 17.0%		3.44e+06 14.99	6 7.25e+01	0.1
LAMMPS_NS_CommKokkos_forward				2.46e+08 17.0%		3.42e+06 14.89	6.48e+01	0.1
LAMMPS_NS_NeighborKokkos: build		5.01e+05 14.5%	6	1.04e+08 7.2%		1.59e+06 6.99	\$ 2.29e+01	0.1
LAMMPS_NS CommKokkos:borders(						1.10e+06 4.8	\$ 8.37e+00	0.1
A - CHIMPO NO OCHMIKING SAL				1 204-07 0.00		0.0705 0.50	0 100.00	



### LAMMPS on Frontier: 8K nodes, 64K MPI ranks + 64K GPU tiles

#### Kernel duration of microseconds



#### LAMMPS on Frontier: 8K nodes, 64K MPI ranks + GPU times

#### Kernel duration of microseconds

			hpcview	er (on login10)				_ • >
View Filter Help								
Profile: Imp 🔳 Trace: Im	ιp				🔠 🕈 😤 \$	* * * * * *	▶ 5) • (8 - 13)	
in view						Depth:	8 - +	9
Time Range: [384,424ms	s, 384,619ms] Cross Hair: (384,514ms, N	VODE 2148189184 CORE 49 RANK 2	30 THREAD 0)					19
4.4.4.4.4		2 22-				Call stack Statistics	GPU Idleness Blame	
	and the second se			and a standard and a standard	and the second sec	<pre>program root&gt;</pre>		
						= main		
			A vier annual and a state		7 11 1 1 1 - 1 1 - 1 1		autofile@ fliblements co.0	8
2.1.1. 1.1.						LAMIMPS_NS::In	put:meg[liblammps.so.0	1
	and the second s			Contraction of the second seco		LAMMPS NS::In	put::execute command() [	liblammp
A 8 - 5 -		1111111		11111			10	
						LAMMPS_NS::K	un::command(int, char**)	liblammp
						AMMPS NS-V	erletKokkos::run(int) [libla:	mmps so
10 10 10 10 10 10 10 10 10 10 10 10 10 1							cricences in animity febra	inipsise.
						LAMMPS_NS::C	ommKokkos::exchange() [l	iblammp:
2 2 2 2						LAMMOE NE	ammKakkasuavchanga de	wiene Kak
2 2 2 2 2						DAMINIPS_NSC	ommkokkosexchange_de	VICESKUP
	2-2-2-2-2					MPIABI Sendre	v flibmpiwrapper.sol	
A	and the second sec				the deside of th			
Real Property in the second						PMPI_Sendrecv	[libmpi_gnu_91.so.12.0.0]	(
2 - 8 - 6 - 6 - 8 - 8 - 8 - 8 - 8 - 8 - 8	a dia dia dia dia dia dia dia dia dia di		And and a second s		and and the local data data and the second s	ofi ca readfrom	Ilibfabric so 1 18 21	
And the Real Property lies and the Real Property	the second se					a on_eq_reduiron	[holdone.so.mo.z]	
	and the first of t							
		1.1.1.1.1.		11-11-11				
				11122		2		
State of the state								
1.1.1								
and the owned of the second	And and the standard on the standard		the second se					
A 48 A 48 A	and the Real Property lies of the second	A Real Provide American Amer		and a first				
1-1 1 1 1								
R . R . R . R						*		
the Barden								
						4		
1-1						=		
4.0 m - 0		and and a dealer of the	the set of	and the state of the state of the	and the second s			
			A de la contra de la					
		Contraction of the local division of the loc						
and and and and and	and the set of the set	212-21 20 201 00 201 00 201 00		and the set the set the set the	201 Con. 201 Con. 201 Con.	_		
364,430ms 364,440ms	304/430ms 304/400ms 304/470ms 304/4	oums 304,430ms 304,300ms 304,5	oms 384,520ms 384,530ms 384,540m	a aaa,aouma aaa,oouma aaa,a/uma aaa,aouma	see, souria see, souria see, souria	1.000		
h view Summary view					8+ 8-	Minimap		
	A CONTRACTOR OF A CONTRACTOR O	41 M 14 14 19 19 19 19 19 19 19 19 19 19 19 19 19				-		
and a second								

# **Coming Attractions**

- Emerging GUI support for accessing hpctoolkit databases on remote systems (try it today)
- More capabilities on AMD GPUs based on AMD's new rocprofiler-sdk (released yesterday)
  - PC sampling with precise attribution without serialization of concurrent kernels
  - improved support for hardware counters
- Integrated support for NVTX/ROCTX/Caliper/Kokkos labels
- Python-based interface for analysis of performance results
- Support for PC sampling on Intel GPUs





## **Quick Start Guide for HPCToolkit on Polaris**

#### Load HPCToolkit on Polaris

module use /soft/modulefiles

module load hpctoolkit

• Measure an execution

-Profile and trace CPU and GPU activity of your application in a job script

mpiexec -n <ranks> hpcrun -e CPUTIME -e gpu=nvidia -tt <myapp> <myapp arguments>

-Use PC sampling to collect instruction-level measurements within GPU kernels

mpiexec -n <ranks> hpcrun -e gpu=nvidia,pc <myapp> <myapp arguments>

Analyze measurement data

-Analyze the CPU and GPU binaries recorded as your application was measured without PC sampling

hpcstruct hpctoolkit-<myapp>-measurements.<jobid>

-Analyze the CPU and GPU binaries recorded as your application was measured with PC sampling

hpcstruct --gpucfg yes hpctoolkit-<myapp>-measurements.<jobid>

Combine measurement data with binary analysis results to produce a performance database

hpcprof hpctoolkit-<myapp>-measurements.<jobid>

• View your performance data

```
hpcviewer hpctoolkit-<myapp>-database.<jobid>
```





# Downloading, Installing, and Using Hpcviewer Graphical User Interface on Your Laptop







## **Hpcviewer Graphical User Interface on Your Laptop**

### Prepare to explore performance data on your laptop

Download and install hpcviewer: https://bit.ly/hpcviewer-install

Select the right one for your laptop: MacOS (Apple Silicon, Intel), Windows, Linux

User manual for hpcviewer: https://hpctoolkit.gitlab.io/hpcviewer





# **Viewing Performance Data**

- Copy a performance database directory to your laptop and open it locally
- Open a performance database on a remote system

Note: using a HPCViewer with a remote system presumes that hpcserver has already been installed on the remote system

- -hpcserver has been pre-installed on Polaris
- -You can install hpcserver anywhere: https://bit.ly/hpcserver-install





# **Configuring Hpcviewer Remote Access**

#### Run hpcviewer

- From the file menu, select "Open remote database"
- Fill in the hostname/IP address: polaris.alcf.anl.gov
- Fill in your username on Polaris
- Fill in the remote installation directory for hpcviewer's server: /eagle/ATPESC2024/hpctoolkit/hpcserver
- Select the authentication method: "Use password"

Click "OK"

- Authenticate using your token as you normally do
- Navigate to a database with the file chooser in /eagle/APTESC/hpctoolkit/databases: quicksilver, lammps, arborx
  - arborx: hpctoolkit-arborx-md.d hpctoolkit-arborx-md-pc.d
  - lammps: hpctoolkit-lmp.d hpctoolkit-lmp-pc.d
  - quicksilver: hpctoolkit-qs-gpu-cuda.d hpctoolkit-qs-gpu-cuda-pc.d





## **Opening a Remote Database**

é hpcviewer	File View Filter					0	) <b>(</b> ))		Q	90	Tue 1	0:27 PM
	New window	9€ N		hpcview	ver							
	Open database Open remote database		Open a remote database. The (hpcserver) has to be already	remote server launched.								Í.

## **Configuring for use with Polaris**

• • •	Remote connection					
Remote connection setu Enter the information need	<b>p</b> led to connect to the remote server					
Hostname/IP address:	polaris.alcf.anl.gov					
Username:	johnmc					
Remote installation directory:	/eagle/ATPESC2024/hpctoolkit/hpcserver					
Use private key: /Use	ers/johnmc/.ssh/id_rsa		•			
Use identity Pro	oxy agent 🔋					
O Use password						
SSH configuration: /Use	ers/johnmc/.ssh/config 😒					
	c	ancel	OK			
		_	-			

### **First View of Polaris: Your Home Directory**

hpcviewer
Pemete database browser
Browsing polaris.alcf.anl.gov (140.221.112.14)
Directory: //home/johnmc
Contents
allinea.polaris/
allinea.thetagpu/
conda/
Config/
in .dbus/
emacs.o/
gnupg/
🛅 .hpctoolkit/
hpss/
keras/
/vn. 🛅
🛅 .oracle_jre_usage/
Cancel OK

## **Navigate to Example Databases**

😑 💿 🔵 Remote database browser
Browsing polaris.alcf.anl.gov (140.221.112.12) Select a HPCToolkit database directory
Directory: //us/eagle/projects/ATPESC2024/hpctoolkit/databases
Contents / arborx/ lammps/ quicksilver/

### Select a Quicksilver Database with Traces

0.0	hpcviewer
	Domesta detabasa berunan
	Remote database browser
	Browsing polaris.alcf.anl.gov (140.221.112.14)
	Select a HPCToolkit database directory
	Directory: /lus/eagle/projects/ATPESC2024/hpctoolkit/databases/quicksilver
	Contents
	i -1
	npctoolkit-qs-gpu-cuda-pc.d/
	npotonic de Bha angre
	Cancel

### After Selecting hpctoolkit-qs-gpu-cuda.d

Texe on a tex	Bittom:go Vice         Rat Vice           Im Vice         Bittom:go Vice         REALTIME (sec): Sum (I)         GPUOP (sec): Sum (I)         GKER (sec): Sum (I)           Im Vice         Bittom:go Vice         Bittom:go Vice         Scate-80 100.0%         2.24e+00 100.0%	And the second		hp	cviewer			
xm oter       Betten:-up view       Fat View         Image: Sum (I)       REALTIME (sec): Sum (I)       GPUOP (sec): Sum (I)       GPUOP (sec): Sum (I)       GRER (sec): Sum (I)	m vive Betten-up view Pat view	Profile: qs 🗮 Trace: qs						-
3 m view [8ttom-up view] [Att view]         3 m view] [8ttom-up view] [Att vie	nn vier Bettom-up View Fit view 6 Fit W M M M M M M M M M M M M M M M M M M							
am vier Betten-up vier Plat vier ● De Se Se REALTIME (sec): Sum (i) REALTIME (sec): Sum (i) GPUOP (sec): Sum (i) GPUOP (sec): Sum (i) GRER (sec): Sum (i) gerinent Aggregate Metrics 1.55e+01 100.0% 2.24e+00 100.0% 2.24e+	an view Betton-up View [Rt View]							
3 0 were Bettem-up view Plat view         3 0 were Bettem-up view Plat view         90       REALTIME (sec): Sum (i) * REALTIME (sec): Sum (ii)         GPC or graphing REALTIME (sec): Sum (i) * REALTIME (sec): Sum (ii)       GPUOP (sec): Sum (i)       GPUOP (sec): Sum (ii)         GPC or graphing REALTIME (sec): Sum (ii) * REALTIME (sec): Sum (iii)       REALTIME (sec): Sum (iii)       GPUOP (sec): Sum (iii)       GPUOP (sec): Sum (iii)         GPC or graphing REALTIME (sec): Sum (iii) * REALTIME (sec): Sum (iii)       REALTIME (sec): Sum (iii)       GPUOP (sec): Sum (iiii)       GPUOP (sec): Sum (iiii) <td< td=""><td>nn vier Bettom-spiker Pat vier 6 Prist Wei Pat vier 7 Prist Vier</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></td<>	nn vier Bettom-spiker Pat vier 6 Prist Wei Pat vier 7 Prist Vier							
Mm Yare         Battan-us view         Flat view                • 0 Point Service S	an view Bottom-up view  Filt view							
xm View Bottom-up View Pat View         > free Strategy View Pat View         > free Strategy View Pat View         > free Strategy View Pat View         Poc       REALTIME (see): Sum (i)       REALTIME (see): Sum (i)       GPUOP (see): Sum (i)       GKER (see): S	an view Bettom-up view Rat view							
min view Battem-up view Plat view	an view Bettom-up view  R14 view							
xm view Betten-up view Pat view         Image: Sum (i) Sum (i) REALTIME (sec): Sum (i) GPUOP (sec): Sum (i) GPUOP (sec): Sum (i) GPUOP (sec): Sum (i) GER (sec): Sum (i) GKER (sec): Sum (i)	an view Bottom-up view Flat view							
Image: Settom-sp Weel Flat view         REALTIME (sec): Sum (i)       GPUOP (sec): Sum (i)	an view Betom-up view Plat view							
Were Veter Bittimer Bittimer         Image: Stitum-spriver         Image: Stitum-spriver         Image: Stitum-spriver         Image: Stitum-spriver         Image: Stitum-spriver         Image: Stitum-spriver         Image: Stitum-spriver <td< td=""><td>an view Bottom-up view Plat view</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></td<>	an view Bottom-up view Plat view							
xm xiker       Bottom-up xiker       Filt xiker         xm xiker       Bottom-up xiker       Filt xiker         xm xiker       REALTIME (sec): Sum (I)       REALTIME (sec): Sum (I)       GKER (sec): Sum (I)       GKER (sec): Sum (I)       GKER (sec): Sum (I)         perfment Agregate Metrics       1.56e+01 100.0%       1.56e+01 100.0%       2.24e+00 100.0%       <	en siew Bettom-up view Fat view ► 6 K S THE X x 111 K be REALTIME (sec): Sum (i) KEALTIME (sec): Sum (i) KEALTIME (sec): Sum (i) GPUOP (sec): Sum (i) GPUOP (sec): Sum (i) GRER (sec): Sum (i) GRER (sec): Sum (i) I.56e+01 100.0% 2.24e+00 100.0% I.56e+01 I00.0% II.56e+01 I00.0% II.56e+01 I00.0% II.56e+01 I00.0% II.56e+01 II.56e							
Settom-up view         Fist view           Image: Setom-up view         Fist view	nn vier Bettem-up view Flat view							
Sentiment Aggregate Metrics       REALTIME (sec): Sum (I) * REALTIME (sec): Sum (E)       GPUOP (sec): Sum (I)       GPUOP (sec): Sum (E)       GKER (sec): Sum (I)       GKER (sec): Sum (I)         sprogram root>       1.56e+01 100.0%       1.56e+01 100.0%       2.24e+00 100.0%       2.24e+00 100.0%       2.24e+00 100.0%         sprogram root>       1.56e+01 100.0%       2.24e+00 100.0%       2.24e+00 100.0%       2.24e+00 100.0%	nn view Bottom-up view Rat view							
20m view   Fist view           20m view   Settem-up view   Settem-up view           20m view   Settem-up view   Settem-up view           20m view   Setem-up view   Settem-up view	nn vier Battem-up vier Int vier							
Sen view Bottom-up view Pat view	an view Battom-up view Flat view ReALTIME (sec): Sum (i) ~ REALTIME (sec): Sum (i) GPUOP (sec): Sum (i) GPUOP (sec): Sum (i) GKER (sec): Sum (i) GKER (sec): Sum (i) GKER (sec): Sum (i) CKER (sec): Sum							
Sam Meeri Bottom-up View         Fat View                ●              ▲	an view Botton-up view Fait view							
Sottom-up view         Pat view           Image: Sottom-up view         Fat view           Image: Sottom	een view Bottom-up view Plat view							
wm view Bottom-up view Flat view         Image: Settom-up view Flat view         Image: Settom-up view Flat view         Percent ReALTIME (sec): Sum (i)       REALTIME (sec): Sum (i)       GPUOP (sec): Sum (ii)       GPUOP (sec): Sum (i)	an view Bottom-up view   Fit view							
Som view         Bat view                • • • • • • • • • • • • •	An view Battom-up view (Piet view)           Image: Battom-up view (Piet view)							
wer Wer Batum-up view Flat view            • ● ● ● ● ■ REALTIME (sec): Sum (!) ~ REALTIME (sec): Sum (!)       GPUOP (sec): Sum (!)       GKER (sec): Sum (!)       GKER (sec): Sum (!)         xperiment Aggregate Metrics          1.56e+01 100.0%          1.56e+01 100.0%          2.24e+00 100.0%          2.24e	an view lightem-up view   Fet view							
Image         REALTIME (sec): Sum (l)         REALTIME (sec): Sum (c)         GPUOP (sec): Sum (c)         GPUOP (sec): Sum (c)         GKER (sec): Sum (c)         GKER (sec): Sum (c)         CKER (sec): Sum (c)         CLEARING	Image: Second	down view Bottom-up view Flat view						
• • • • • • • • • • • • • • • • • • •	Image: Second							
pe         REALTIME (sec): Sum (l)         REALTIME (sec): Sum (E)         GPUOP (sec): Sum (I)         GPUOP (sec): Sum (E)         GKER (sec): Sum (I)	pe         REALTIME (sec): Sum (l)         REALTIME (sec): Sum (E)         GPUOP (sec): Sum (l)         GPUOP (sec): Sum (E)         GKER (sec): Sum (I)		×.					
per         RCACIMAC (sec) sum (c)         RCACIMAC (sec) sum (c)         OFOOT (sec) sum (c)         OFOOT (sec) sum (c)         OFOOT (sec) sum (c)         ORCA (sec)	be       RCACIMIC (Sec) Juni ()       RCACIMIC (Sec) Juni (	2000	PEALTIME (sec): Sum (I) - PE	EALTIME (see): Sum (E)	CPLICE (sec): Sum (I) (	CPLIOP (coc): Sum (E)	GKED (coc): Sum (II)	GKER (sec): Sum (E)
Appendix Aggregate Metrics         1.50e+01 100.0%         2.24e+00 100.0%         2.24e+00 100.0%         2.24e+00 100.0%           cprogram root>         1.56e+01 100.0%         2.24e+00 100.0%         2.24e+00 100.0%         2.24e+00 100.0%         2.24e+00 100.0%	perment Aggregate Metrics         1.50e+01 100.0%         1.50e+01 100.0%         2.24e+00 100.0%         2.24e+00 100.0%         2.24e+00 100.0%           program root>         1.56e+01 100.0%         2.24e+00 100.0%         2.24e+00 100.0%         2.24e+00 100.0%	rohe	1 5Co : 01 100 0	1 ECo : 01 100 00	2 24++00 100 0°	2 24++00 100 00	OKER (Sec). Sum (I)	2 24-100 100 0
sprogram root>         1.56e+01 100.0%         2.24e+00 100.0%         2.24e+00 100.0%   <	program root>     1.56e+01 100.0%     2.24e+00 100.0%     2.24e+00 100.0%	Experiment Aggregate Metrics	1.566+01 100.0%	1.500+01 100.0%	2.240+00 100.0%	2.240+00 100.0%	2.240+00 100.0%	2.240+00 100.0
Normal ConstraintsNormal Cons	Ander and	<program root=""></program>	1.56e+01 100.0%		7 740+00 100 0%		7 74	
					2.240100 100.08		2.240+00 100.0%	
Image: section of the section of t					2.240100 100.08		2.240+00 100.0%	
Image: State in the state					21242400 100108		2.240+00 100.0%	
Image: Section of the section of t					2.240400 100.00		2.240+00 100.0%	
Image: Section of the section of t					2.240100 100.0%		2.240+00 100.0%	
An and an					2.240100 100.00		2.240+00 100.0%	
					2.240400 100.08		2.240+00 100.0%	
A A A A A A A A A A A A A A A A A A A					2.240400 100108		2.240+00 100.0%	
					21240700 100103		2.240+00 100.0%	
					21210100 100100		2.240+00 100.0%	
					21210100 100100		2.242+00 100.0%	
					21240700 100103		2.246+00 100.0%	
					21210700 100103		2.242+00 100.0%	
					21210100 100100		2.246+00 100.0%	
					21210100		2.246+00 100.0%	
					21210100 100100		2.246+00 100.0%	
					21210100		2.246+00 100.0%	
							2.242+00 100.0%	
							2.246+00 100.0%	
							2.246+00 100.0%	
							2.246+00 100.0%	
							2.246+00 100.0%	
							2.242+00 100.0%	
							2.242+00 100.0%	
							2.246+00 100.0%	

### Select the Tab "Trace: qs"



### Use the Filter to "Uncheck all" and Check "GPU" streams


### **See Load Imbalance Across the Four GPUs**



### The Profile View in the other "PC Sampling" Database

	hpcviewer	
Profile: qs 🗮 Trace: qs 🗺 Profile: qs		
CollisionEvent.cc X		
67 for (int isoIndex = 0; isoIndex < numIsos && cu	<pre>rrentCrossSection &gt;= 0; isoIndex++)</pre>	
68 {	where met [a] obelMetIndex] iso[isoIndex] aid:	
70 int numReacts = monteCarlo-> nuclearData->oe	tNumberReactions(uniqueNumber):	
<pre>71 for (int reactIndex = 0; reactIndex &lt; numRea</pre>	<pre>icts; reactIndex++)</pre>	
72 {		
73 CURRENTLY CONTRACTOR CONTRACTO	<pre>ction(montelario, reactindex, mc_particle.domain, mc_particle.ce) coup);</pre>	ί,
75 if (currentCrossSection < 0)	oup/,	
76 {		
<pre>77 selectedIso = isoIndex;</pre>		
<pre>/8 selectedUniqueNumber = uniqueNumber; 79 selectedPeact = reactIndex;</pre>		
80 break;		
81 }		
Fop-down view Bottom-up view Flat view		
全 🗄 🌢 fx 🖅 🕅 🚍 A* 🖉 비나 📽		
Scope	GINS: Sum (I) GINS: Sum (E) GINS:STL_ANY: Sum (I) GINS	STL_ANY: Sum (E)
Experiment Aggregate Metrics	2.15e+11 100.0% 2.15e+11 100.0% 2.03e+11 100.0%	2.03e+11 100.0%
<program root=""></program>	2.15e+11 100.0% 2.03e+11 100.0%	
<ul> <li>main</li> </ul>	2.15e+11 100.0% 2.03e+11 100.0%	
<ul> <li>loop at main.cc: 66</li> </ul>	2.15e+11 100.0% 2.03e+11 100.0%	
<ul> <li>58 » cycleTracking(MonteCarlo*)</li> </ul>	2.15e+11 100.0% 2.03e+11 100.0%	
Ioop at main.cc: 232	2.15e+11 100.0% 2.03e+11 100.0%	
<ul> <li>loop at main.cc: 232</li> </ul>	2.15e+11 100.0% 2.03e+11 100.0%	
✓ 127 » <gpu kernel=""></gpu>	2.15e+11 100.0% 2.03e+11 100.0%	
<ul> <li>» CycleTrackingKernel(MonteCarlo*, int, ParticleVault*, ParticleV</li> </ul>	ault*) 2.15e+11 100.0% 1.03e+08 0.0% 2.03e+11 100.0%	9.83e+07 0.0%
4 132 » CycleTrackingGuts(MonteCarlo*, int, ParticleVault*, Partic	cleVault*) 2.15e+11 99.9% 2.04e+09 1.0% 2.03e+11 99.9%	2.03e+09 1.0%
4 26 » [I] CycleTrackingFunction(MonteCarlo*, MC_Particle&, in	t, ParticleV 1.08e+11 50.4% 4.95e+08 0.2% 9.63e+10 47.5%	4.38e+08 0.2%
loop at CycleTracking.cc: 118	1.08e+11 50.4% 4.61e+08 0.2% 9.63e+10 47.5%	4.11e+08 0.2%
4 63 » CollisionEvent(MonteCarlo*, MC_Particle&, unsigned in	t) 7.08e+10 32.9% 7.69e+09 3.6% 6.21e+10 30.7%	6.42e+09 3.2%
<ul> <li>loop at CollisionEvent.cc: 67</li> </ul>	5.66e+10 26.3% 1.51e+09 0.7% 4.88e+10 24.1%	1.31e+09 0.6%
<ul> <li>loop at CollisionEvent.cc: 71</li> </ul>	5.27e+10 24.5% 3.97e+09 1.8% 4.54e+10 22.4%	3.08e+09 1.5%
4 73 » macroscopicCrossSection(MonteCarlo*, int, int, int, int, int, int, int, int	int, int) 4.87e+10 22.7% 1.78e+10 8.3% 4.23e+10 20.9%	1.49e+10 7.3%
41 » NuclearData::getReactionCrossSection(unsigned in the section of the section)	nt, unsigne 2.71e+10 12.6% 1.35e+10 6.3% 2.40e+10 11.8%	1.20e+10 5.9%
253 » [I] NuclearDataReaction::getCrossSection(unsidential)	uned int) 9,00e+09 4,2% 4,83e+09 2,2% 7,87e+09 3,9%	4 430+09 2 2%

6.76e+09

3.1% 6.76e+09

3.1%

6.45e+09 3.2%

6.45e+09

3.2%

NuclearData.cc: 253

# Collecting Performance Data with HPCToolkit: Turnkey Examples







### Hands-on Tutorial Examples

- % git clone https://github.com/hpctoolkit/hpctoolkit-tutorialexamples
- % cd hpctoolkit-tutorial-examples/gpu/nvidia
- %ls
  - arborx.kokkos lammps.kokkos quicksilver.cuda





### A Hands-on Example: Quicksilver

### A LLNL proxy application for dynamic Monte Carlo particle transport (MPI + CUDA)

cd hpctoolkit-tutorial-examples/gpu/nvidia/quicksilver.cuda source setup/polaris.sh

make build

make run

make run-pc

make view

make view-pc

#### Notes

- Running "make view" or "make view-pc" requires an X11 desktop to support the GUI
- Alternatively, you can use the hpcviewer's "open remote database" capability to view the databases
  - hpctoolkit-qs-gpu-cuda.d: profiles + traces
  - hpctoolkit-qs-gpu-cuda-pc.d: GPU PC samples





## **Analyzing Quicksilver Traces**

- Select the Trace tab "Trace: qs"
- Identifying the traces
  - Select a pixel on a trace line
  - Look at legend on the top of the display, which reports the location of the "cross hair"
  - Is this a CPU or GPU trace line?
  - Repeat this a few times to identify what each of the trace lines represents
- Notice that each time you select a colored pixel on a trace line, you will be shown the function call stack in the rightmost pane
- At the top of the pane is a "depth" indicator, that indicates what level in the call stack you are viewing. The selected level will also be highlighted
- You can change the depth of your view by using the depth up/down, typing a depth, or simply selecting a frame in the call stack at the desired depth
- You can select 📥 above the call stack frame to show the call stacks at the deepest depth
  - If a sample doesn't have an entry at the selected depth, its deepest frame will be shown





- Zoom in on a region in a trace by selecting it in the trace display
- Use the back button <sup>1</sup> to undo a zoom
- Use the control buttons
  - expand or contract the pane
  - move left, right, up, or down
- Keep an eye on the minimap in the lower right corner of the display to know what part of the trace you are viewing
- Use the home button do reset the trace view to show the whole trace





## **Analyzing Quicksilver Traces**

- Select the Trace tab "Trace: qs"
- Configure filtering
  - Use the Filter menu to select Filter Execution Contexts
  - In the filtering menu, select "Uncheck all"
  - Now, in the empty box preceded by "Filter:", type "GPU" and then click "Check all"
  - Select "OK".
  - Now, the Trace View will show only trace lines for the GPUs.
- Inspect the trace data
  - Is the work load balanced across the GPUs? How can you tell?
  - Bring up the filter menu again. Select "Uncheck all". Type in "RANK 3" in the Filter box. Select thread 0 and the GPU context. Select "OK".
  - Move the call stack to depth 2
    - What CPU function is Rank 3 thread 0 executing when the GPU is idle?
    - Does this suggest any optimization opportunities?





## Analyzing the Quicksilver Summary Profile

- Select the Profile Tab "Profile: qs"
- Use the column selector IV to deselect and hid the two REALTIME columns
- Select the GPU OPS column, which represents time spent in all GPU operations
- Select the <sup>6</sup> button to show the "hot path" according to the selected column
  - the hot path of parent will continue into a child as long as the child accounts for 50% or more of the parent's cost
- The hot path will select "CycleTrackingKernel" a GPU kernel that consumes 100% of the GPU cost in this profile
- Use the III button to graph "GPU OPS (I)" inclusive GPU operations across the profiles
  - Are the GPU operations balanced or not across the execution contexts (ranks)?





## Analyzing the Quicksilver Summary Profile

- You will notice that for quicksilver, HPCToolkit doesn't report any data copies between the host and device
  - The quicksilver code uses "unified memory" so that all of the data movement occurs between CPU and GPU using page faults rather than explicit copies
  - Today's GPU hardware doesn't support attribution of page faults to individual instructions
    - We could profile them, but not attribute them to code





## **Analyzing Quicksilver PC Samples**

#### Using a measurement database with traces that was collected \*with\* PC sampling enabled

Using the default top-down view of the profile

- Select the column "GINS (I)" to focus on the measurement of inclusive GPU Instructions
- Select use the flame button to look at where the instructions are executed
- In the call stack revealed, you will <gpu kernel> placeholder that separates CPU activity (above) from GPU kernel activity (below)
- Below the <gpu kernel> placeholder you will see the function calls, inlined functions, loops and statements in HPCToolkit's reconstruction of calling contexts within the CycleTrackingKernel
- Using the bottom-up view of the profile
  - · Select the bottom-up tab of above the control pane
  - Select the GINS STL\_ANY (E) column, which will sort the functions by the exclusive GPU instruction stalls within that function
  - Scroll right to see which of the types of contributing types of stalls accounts for most of the STL\_ANY amount
  - · Select the function that has the most exclusive stalls
  - Select the hot path to see where this function is called from.
    - · Where do the calls to the costly function come from?
    - Does there appear to be an opportunity to reduce the number of calls to this function?





## Filtering Tips to Hide Unwanted Implementation Details

- Filter "descendants-only" of CCT nodes with names \*MPI\* to hide the details of MPI implementation in profiles and traces
- Filter internal details of RAJA and SYCL templates to suppress unwanted detail using a "self-only" filter





### A Hands-on Example: ArborX

### Performance portable algorithms for geometric search MPI + Kokkos + OpenMP

cd hpctoolkit-tutorial-examples/gpu/nvidia/arborx source setup/polaris.sh make build make run make run-pc Notes • Running "make view" or

make view

make view-pc

- Running "make view" or "make view-pc" requires an X11 desktop to support the GUI
- Alternatively, you can use the hpcviewer's "open remote database" capability to view the databases
  - hpctoolkit-arborx-md.d: profiles + traces
  - hpctoolkit-arborx-md-pc.d: GPU PC samples





### **Analyzing ArborX Traces**

- Is the GPU active for most of the brief execution or not?
- Zoom in on the pair of trace lines that represents the GPU activity for a rank
  - You will see that there are two GPU trace lines per process
  - What happens on each?





### A Hands-on Example: LAMMPS

### A molecular dynamics code with a focus on materials modeling (Kokkos + MPI)

cd hpctoolkit-tutorial-examples/gpu/nvidia/lammps.cuda source setup/polaris.sh make build make run make run-pc

- Running "make view" or "make view-pc" requires an X11 desktop to support the GUI
- Alternatively, you can use the hpcviewer's "open remote database" capability to view the databases
  - hpctoolkit-Imp.d: profiles and traces
  - hpctoolkit-Imp-pc.d: GPU PC samples



make view

make view-pc



## Analyzing LAMMPS Profiles, Traces, and PC Samples

HPCToolkit can profile, trace, and collect PC samples for codes regardless of their complexity





### Troubleshooting: Only GPU kernel Name

Need to measure with PC sampling to measure within GPU kernels

```
....
                                                                                                                                                                                     hneviewer
Profile: ArborX, Example, MolecularDynamics.exe 🗮 Trace: ArborX, Example, MolecularDynamics.exe
                                                                                                                                                                                                                                                                                                                                                                              - -
Kokkos Cuda KernelLaunch.hpp X cuda runtime.h
   81 driver():
   82 }
   83
   84 template <class DriverType>
   85 global static void cuda parallel launch local memory(
   86 const DriverType driver) {
  87 driver():
   88 }
   89
   90 template <class DriverType, unsigned int maxTperB, unsigned int minBperSM>
   91_global__launch_bounds_(
   92
              maxTperB.
   93
              minBperSM) static void cuda_parallel_launch_local_memory(const DriverType
   94
                                                                                                                                          driver) +
   95 driver();
   96 }
   07
 Top-down view Bottom-up view Flat view
   1 4 6 fx 2 1 A a 11 2
   Scope
                                                                                                                                                                                                         GKER (sec): Sum (I) GKER (sec): Sum (E) GXCOPY (sec): Sum (I) GXCOPY (sec): Sum (E) GXCOPY:H2D (B)
              437 » [I] ArborX::Details::traverse<Kokkos::Cuda, ArborX::BoundingVolumeHierarchy<Kokkos::CudaSpace, ArborX::PairValue</p>
                                                                                                                                                                                                         3.63e-04 39.9%
                                                                                                                                                                                                         3.63e-04 39.9%
               497 » ArborX::Details::TreeTraversal<ArborX::BoundingVolumeHierarchy<Kokkos::CudaSpace, ArborX::PairValueIndex<Arbo.</p>
                4 63 » Kokkos::parallel_for<Kokkos::RangePolicy<Kokkos::Cuda, ArborX::Details::TreeTraversal<ArborX::BoundingVolumeHie.</p>
                                                                                                                                                                                                         3.63e-04 39.9%
                 4 144 » Kokkos::Impl::ParallelFor<ArborX::Details::TreeTraversal<ArborX::BoundingVolumeHierarchy<Kokkos::CudaSpace, A.</p>
                                                                                                                                                                                                         3.63e-04 39.9%
                                                                                                                                                                                                         3.63e-04 39.9%
                   108 » [I] Kokkos::Impl::CudaParallelLaunch<Kokkos::Impl::ParallelFor<ArborX::Details::TreeTraversal<ArborX::BoundingV.</p>
                    717 » [I] Kokkos::Impl::CudaParallelLaunchImpl<Kokkos::Impl::ParallelFor<ArborX::Details::TreeTraversal<ArborX::Boundi.</p>
                                                                                                                                                                                                         3.63e-04 39.9%
                      4 678 >> [I] Kokkos::Impl::CudaParallelLaunchKernelinvoker<Kokkos::Impl::ParallelFor<ArborX::Details::TreeTraversal<Arb.</p>
                                                                                                                                                                                                         3.63e-04 39.9%
                                                                                                                                                                                                         3.63e-04 39.9%
                       4 368 » [I] cuda_parallel_launch_local_memory<Kokkos::Impl::ParallelFor<ArborX::Details::TreeTraversal<ArborX::Boun.</p>
                                                                                                                                                                                                         3.63e-04 39.9%
                         486 » [I] wrapper device stub cuda parallel launch local memory<Kokkos::Impl::ParallelFor<ArborX::Details::Tr...</p>
                                                                                                                                                                                                         3.63e-04 39.9%
                           406 » ZL592 device stub ZN6Kokkos4Impl33cuda parallel launch local memorvINS0 11ParallelForIN6Arbor.
                                                                                                                                                                                                           3.63e-04 39.9%
                              4 216 » <qpu kernel>
                                                                                                                                                                                                          3.63e-04 39.9%
                                                                                                                                                                                                         3.63e-04 39.9% 3.63e-04 39.9%
                               A >> Kokkos::Impl::cuda parallel launch local memory<Kokkos::Impl::ParallelFor<ArborX::Details::TreeTraversal<Ar...</p>
                                    Kokkos Cuda KernelLaunch.hpp: 85
                                 2 10 27 SHIDU MHT R
            182 » ArborX::BoundingVolumeHierarchy<Kokkos::CudaSpace, ArborX::PairValueIndex<ArborX::Box, unsigned int>, ArborX::D...
                                                                                                                                                                                                         2.53e-04 27.8%
            209 » ArborX::Details::KokkosExt::exclusive_scan<Kokkos::Cuda, Kokkos::View<int*, Kokkos::CudaSpace>, Kokkos::CudaSpace
                                                                                                                                                                                                         9.15e-06
                                                                                                                                                                                                                                1.0%
            237 » Kokkos::parallel_for<Kokkos::RangePolicy<Kokkos::Cuda>, _nv_hdl_wrapper_t<false, false, false, _nv_dl_tag<void (*).</p>
                                                                                                                                                                                                         2.30e-06
                                                                                                                                                                                                                                0.3%
            205 » Kokkos::parallel_for<Kokkos::RangePolicy<Kokkos::Cuda>, _nv_hdl_wrapper_t<false, false, false, _nv_dl_tag<void (*).</p>
                                                                                                                                                                                                         2.18e-06
                                                                                                                                                                                                                                0.2%
            211 ArborX::Details::KokkosExt::lastElement<Kokkos::Cuda.int*, Kokkos::CudaSpace>(Kokkos::Cuda const&, Kokkos::View<i,</p>
                                                                                                                                                                                                                                                                                 1.92e-06 1.3%
                                                                                                                                                                                                                                                                                 1 00- 00
             • 040 a Ashary, Datalla, Kaldaa Tutuna Baskika ahabbababa Kaldaa, Orda Kaldaa, Afan Jata Kaldaa, Orda Casaa, - Waldaa
                                                                                                                                                                                                                                                                                                     A 00.
                                                                                                                                                                                                                                                                                                                                                                 1 200.02
```

### Troubleshooting: No GPU source code lines with PC sampling

• If you don't see source code with PC sampling on NVIDIA GPUs: compile with "-lineinfo" option

k 🗢 🔍	hpcviewer									
Profile: ArborX_Example_MolecularDynamics.exe										
kkos_Cuda_KernelLaunch.hpp Kokkos_Parallel.hpp ArborX_DetailsTreeTraversal.hpp X										
61 else										
<pre>22</pre>										
73 {} 74 75 struct OneLeafTree 76 {};										
p-down view Bottom-up view Flat view										
全 🗄 🍐 🖅 🕅 🚍 At 🖉 비 🕱										
Scope	GINS: Sum (	I) GINS: Sum (I	E)	GINS:STL_ANY:	Sum (I)	GINS:STL_ANY: S	Sum (E)	GINS:STL_IFET:	Sum (I) GINS	E
2044 P. Arbertalistici Sonaphyn appeningistiquer ynngistiokkostioudu, Arbertalistig den Arbertalister a beretter a berett	1 200.07	47.40		1 250.07	40.00			2.070.06	52.00	
<ul> <li>244 » ArborX::BoundingvolumeHierarchy<kokkos::cudaspace, arborx::pairvalueindex<arborx::box,="" li="" un<=""> <li>402 » (II) ArborX::Dotalloutowarea (Kalilaca: Oudo ArborX::Dauralia) (alurativalueIndex<arborx::box, li="" un<=""> </arborx::box,></li></kokkos::cudaspace,></li></ul>	1.390+07	47.4%		1.250+07	40.0%			2.070+00	53.9%	
<ul> <li>437 » [I] ArborX::Details::traverse<kokkos::cuda, arborx::boundingvolumehierarchy<kokkos::cudasp<="" li=""> <li>407 » ArborX::Details::traverse</li> <li>407 » ArborX::Details::traverse</li> </kokkos::cuda,></li></ul>	1.390+07	47.4%		1.250+07	40.0%			2.070+00	53.9%	1
497      ArborX::Details:: TreeTraversal <arborx::boundingvolumehierarchy<rokkos::cudaspace, 497="" 499="" 49<="" arborx="" td=""><td>1.390+07</td><td>47.45</td><td></td><td>1.250+07</td><td>40.0%</td><td></td><td></td><td>2.070+00</td><td>53.9%</td><td>1</td></arborx::boundingvolumehierarchy<rokkos::cudaspace,>	1.390+07	47.45		1.250+07	40.0%			2.070+00	53.9%	1
63 % Kokkos::paraliel_tor <kokkos::rangepolicy<kokkos::cuda, arborx::details::treetraversal<arborx<="" p=""></kokkos::rangepolicy<kokkos::cuda,>	1.390+07	47.4%		1.250+07	48.0%			2.070+00	53.9%	4
144 W Kokkos::impl::ParailelFol <arborx::details::treetraversal<arborx::boundingvolumehierarchy< 100="" 70="" kokkos::arborx::details::treetraversal<arborx::details::treetraversal<="" p="" p.=""></arborx::details::treetraversal<arborx::boundingvolumehierarchy<>	1.390+07	47.4%		1.250+07	40.0%			2.070+00	53.9%	1
<ul> <li>108 » [I] Kokkos::impi::CudaParallelLaunch<kokkos::impi::parallelfor< li=""> <li>108 » [I] Kokkos::impi::CudaParallelLaunch<kokkos::impi::parallelfor< li=""> </kokkos::impi::parallelfor<></li></kokkos::impi::parallelfor<></li></ul>	1.39e+07	47.4%		1.25e+07	40.0%			2.070+00	53.9%	1
4 717 » [I] Kokkos::Impl::CudaParallelLaunchimpl <kokkos::impl::parallelfor<arborx::details::treetra< p=""></kokkos::impl::parallelfor<arborx::details::treetra<>	1.39e+07	47.4%		1.25e+07	48.0%			2.0/e+06	53.9%	1
4 678 » [I] Kokkos::Impl::CudaParallelLaunchKernelInvoker <kokkos::impl::parallelfor<arborx::deta< p=""></kokkos::impl::parallelfor<arborx::deta<>	1.39e+07	47.4%		1.25e+07	48.0%			2.0/e+06	53.9%	ł
4 368 » [I] cuda_parallel_launch_local_memory <kokkos::impl::parallelfor<arborx::details::treeti accession of the second secon</kokkos::impl::parallelfor<arborx::details::treeti 	a 1.39e+0/	47.4%		1.25e+07	48.0%			2.0/e+06	53.9%	J
4 86 » [I]wrapperdevice_stub_cuda_parallel_launch_local_memory <kokkos::impl::parallel< p=""></kokkos::impl::parallel<>	b 1.39e+07	47.4%		1.25e+07	48.0%			2.0/e+06	53.9%	
406 » _ZL592_device_stubZN6Kokkos4Impl33cuda_parallel_launch_local_memoryINS0	1 1.39e+07	47.4%		1.25e+07	48.0%			2.07e+06	53.9%	
403 » [I] cudal aunchKernel <char></char>	1.39e+07	47.4%		1.25e+07	48.0%			2.07e+06	53.9%	
216 » <gpu kernel=""></gpu>	1.39e+07	47.4%		1.25e+07	48.0%			2.07e+06	53.9%	
<ul> <li>Kokkos::Impl::cuda_parallel_launch_local_memory<kokkos::impl::parallelfor<arborx::delaunch_local_memory<kokkos::impl::parallelfor<arborx::delaunch_local_memory<kokkos::impl::parallelfor<arborx::delaunch_local_memory<kokkos::impl::parallelfor<arborx::delaunch_local_memory<kokkos::impl::parallelfor<arborx::delaunch_local_memory<kokkos::impl::parallelfor<arborx::delaunch_local_memory<kokkos::impl::parallelfor<arborx::delaunch_local_memory<kokkos::impl::parallelfor<arborx::delaunch_local_memory<kokkos::impl::parallelfor<arborx::delaunch_local_memory<kokkos::impl::parallelfor<arborx::delaunch_local_memory<kokkos::impl::parallelfor<arborx::delaunch_local_memory<kokkos::impl::parallelfor<arborx::delaunch_local_memory<kokkos::impl::parallelfor<arborx::delaunch_local_memory<kokkos::impl::parallelfor<arborx::delaunch_local_memory<kokkos::impl::parallelfor<arborx::delaunch_local_memory<kokkos::impl::parallelfor<arborx::delaunch_local_memory<kokkos::impl::parallelfor<arborx::delaunch_local_memory<kokkos::impl::parallelfor<arborx::delaunch_local_memory<kokkos::impl::parallelfor<arborx::delaunch_local_memory<kokkos::impl::parallelfor<arborx::delaunch_local_memory<kokkos::impl::parallelfor<arborx::delaunch_local_memory<kokkos::impl::parallelfor<arborx::delaunch_local_memory<kokkos::impl::parallelfor<arborx::delaunch_local_memory<kokkos::impl::parallelfor<arborx::delaunch_local_memory<kokkos::impl::parallelfor<arborx::delaunch_local_memory<kokkos::impl::parallelfor<arborx::delaunch_local_memory<kokkos::impl::parallelfor<arborx::delaunch_local_memory<kokkos::impl::parallelfor<arborx::delaunch_local_memory<kokkos::impl::parallelfor<arborx::delaunch_local_memory<kokkos::impl::parallelfor<arborx::delaunch_local_memory<kokkos::impl::parallelfor<arborx::delaunch_local_memory<kokkos::impl::parallelfor<arborx::delaunch_local_memory<kokkos::impl::parallelfor<arborx::delaunch_local_memory<kokkos::impl::parallelfor<arborx::delaunch_local_memory<kokkos::impl::parallelfor<arborx::delaunch_local_memory<kokkos::impl::parallelfor< td=""><td>t 1.39e+07</td><td>47.4% 1.36e+07</td><td>46.2%</td><td>s 1.25e+07</td><td>48.0%</td><td>1.22e+07</td><td>46.8%</td><td>2.07e+06</td><td>53.9%</td><td></td></kokkos::impl::parallelfor<arborx::delaunch_local_memory<kokkos::impl::parallelfor<arborx::delaunch_local_memory<kokkos::impl::parallelfor<arborx::delaunch_local_memory<kokkos::impl::parallelfor<arborx::delaunch_local_memory<kokkos::impl::parallelfor<arborx::delaunch_local_memory<kokkos::impl::parallelfor<arborx::delaunch_local_memory<kokkos::impl::parallelfor<arborx::delaunch_local_memory<kokkos::impl::parallelfor<arborx::delaunch_local_memory<kokkos::impl::parallelfor<arborx::delaunch_local_memory<kokkos::impl::parallelfor<arborx::delaunch_local_memory<kokkos::impl::parallelfor<arborx::delaunch_local_memory<kokkos::impl::parallelfor<arborx::delaunch_local_memory<kokkos::impl::parallelfor<arborx::delaunch_local_memory<kokkos::impl::parallelfor<arborx::delaunch_local_memory<kokkos::impl::parallelfor<arborx::delaunch_local_memory<kokkos::impl::parallelfor<arborx::delaunch_local_memory<kokkos::impl::parallelfor<arborx::delaunch_local_memory<kokkos::impl::parallelfor<arborx::delaunch_local_memory<kokkos::impl::parallelfor<arborx::delaunch_local_memory<kokkos::impl::parallelfor<arborx::delaunch_local_memory<kokkos::impl::parallelfor<arborx::delaunch_local_memory<kokkos::impl::parallelfor<arborx::delaunch_local_memory<kokkos::impl::parallelfor<arborx::delaunch_local_memory<kokkos::impl::parallelfor<arborx::delaunch_local_memory<kokkos::impl::parallelfor<arborx::delaunch_local_memory<kokkos::impl::parallelfor<arborx::delaunch_local_memory<kokkos::impl::parallelfor<arborx::delaunch_local_memory<kokkos::impl::parallelfor<arborx::delaunch_local_memory<kokkos::impl::parallelfor<arborx::delaunch_local_memory<kokkos::impl::parallelfor<arborx::delaunch_local_memory<kokkos::impl::parallelfor<arborx::delaunch_local_memory<kokkos::impl::parallelfor<arborx::delaunch_local_memory<kokkos::impl::parallelfor<arborx::delaunch_local_memory<kokkos::impl::parallelfor<arborx::delaunch_local_memory<kokkos::impl::parallelfor<arborx::delaunch_local_memory<kokkos::impl::parallelfor<></li></ul>	t 1.39e+07	47.4% 1.36e+07	46.2%	s 1.25e+07	48.0%	1.22e+07	46.8%	2.07e+06	53.9%	
<ul> <li>loop at [29c7dccbe52b18735fc23021402e20bb.gpubin]: 0</li> </ul>	1.39e+07	47.3% 1.64e+05	0.6%	1.25e+07	47.9%	1.31e+05	0.5%	2.07e+06	53.8%	
<ul> <li>loop at [29c7dccbe52b18735fc23021402e20bb.gpubin]: 0</li> </ul>	1.37e+07	46.7% 1.34e+07	45.5%	1.24e+07	47.4%	1.21e+07	46.2%	2.06e+06	53.7%	
[29c7dccbe52b18735fc23021402e20bb.gpubin]: 0	1.34e+07	45.5% 1.34e+07	45.5%	1.21e+07	46.2%	1.21e+07	46.2%	1.85e+06	48.0%	
<ul> <li>* \$_ZN6Kokkos4Impl33cuda_parallel_launch_local_memoryINS0_11ParallelForIN6Arb</li> </ul>	r 3.40e+05	1.2% 3.40e+05	1.2%	3.03e+05	1.2%	3.03e+05	1.2%	2.17e+05	5.6%	
[29c7dccbe52b18735fc23021402e20bb.gpubin]: 0	3.40e+05	1.2% 3.40e+05	1.2%	3.03e+05	1.2%	3.03e+05	1.2%	2.17e+05	5.6%	
<ul> <li>* \$_ZN6Kokkos4Impl33cuda_parallel_launch_local_memoryINS0_11ParallelForIN6Arb</li> </ul>	r									

### Troubleshooting: Compiling ArborX with GPU Line Map Info

- ArborX cmake isn't set up to include GPU line mappings
- Force the compiler to record GPU line mappings
- cmake -DARBORX\_ENABLE\_EXAMPLES=true \
   -DCMAKE\_INSTALL\_PREFIX=`pwd`/../install \
   -DCMAKE\_CXX\_COMPILER=g++ \
   -DCMAKE\_BUILD\_TYPE=RelWithDebInfo \
   -DCMAKE\_CXX\_FLAGS\_RELWITHDEBINFO="-O2 -g -DNDEBUG -lineinfo"





### **HPCToolkit Resources**

- Documentation
  - -Man pages
    - https://hpctoolkit.org/man/hpctoolkit.html
  - —User manual
    - https://hpctoolkit.org/manual/HPCToolkit-users-manual.pdf
  - -Tutorial videos
    - https://hpctoolkit.org/training.html
    - recorded demo of GPU analysis of Quicksilver: https://youtu.be/vixa3hGDuGg
    - recorded tutorial presentation including demo with GPU analysis of GAMESS: <a href="https://vimeo.com/781264043">https://vimeo.com/781264043</a>
- Software
  - -Download hpcviewer GUI binaries for your laptop, desktop, cluster, or supercomputer
    - OS: Linux, Windows, MacOS
    - Processors: x86\_64, aarch64, ppc64le
    - https://hpctoolkit.org/download.html
  - -Install HPCToolkit on your Linux desktop, cluster, or supercomputer using Spack
    - https://hpctoolkit.org/software-instructions.html





## **Current Funding for HPCToolkit**

- Government
  - -Lawrence Livermore National Laboratory Subcontract B665301
  - -DOE Software Tools Ecosystem Project UT-Battelle Subcontract CW54422
  - -Argonne National Laboratory Subcontract 4F-60094
- Corporate
  - —Advanced Micro Devices
  - -TotalEnergies EP Research & Technology USA, LLC.



