

University of Oregon AMD APU Workshop

Jeremy L Thompson

University of Colorado Boulder

jeremy@jeremylt.org

13 Feb 2025

Overview

- 1 Introduction
- 2 Day 2
- 3 Day 3
- 4 Final Outbrief
- 5 Questions

libCEED, Ratel, and HONEE Team



libCEED Repo: <https://github.com/CEED/libCEED>

HONEE Repo: <https://gitlab.com/phypid/HONEE>

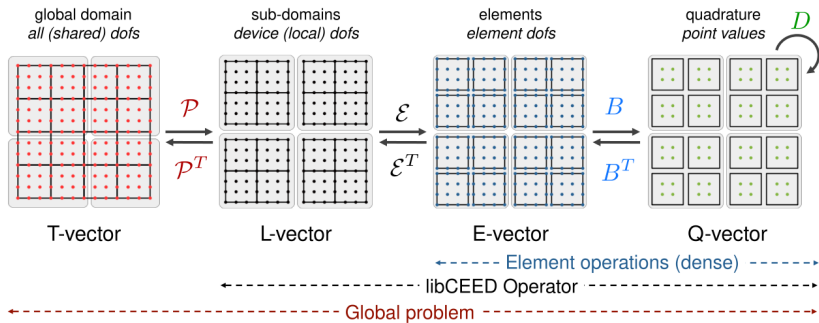
Ratel Repo: <https://gitlab.com/micromorph/ratel>

Developers: **Zach Atkins**, Jed Brown, Fabio Di Gioacchino, Leila Ghaffari,
Kenneth Jansen, **Rezgar Shakeri**, James Wright,
Jeremy L Thompson

The authors acknowledge support by the Department of Energy, National Nuclear Security Administration, Predictive Science Academic Alliance Program (PSAAP) under Award Number DE-NA0003962.

Matrix-Free Operators from libCEED

$$A = \mathcal{P}^T \mathcal{E}^T B^T D B \mathcal{E} \mathcal{P}$$



libCEED provides arbitrary order matrix-free operator evaluation

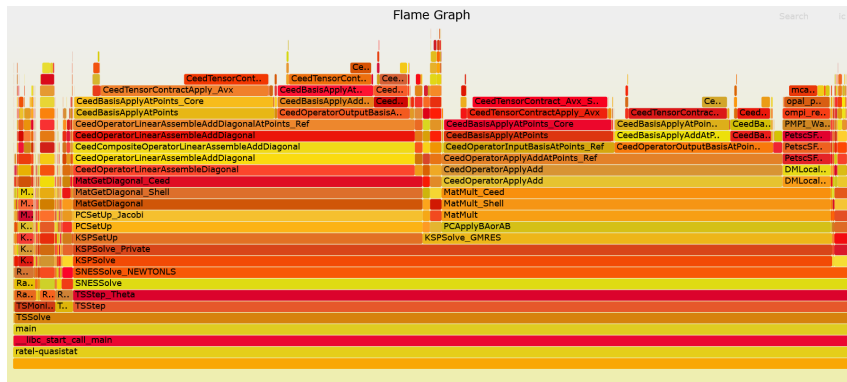
Workshop Goals

- CPU/GPU unified memory space testing and optimization
- Basis kernel optimization, standard and AtPoints quadrature
- Better understand user QF performance and best practices
- Diagonal and full assembly kernel tuning

Day 2 Update

- libCEED and PETSc built on Odyssey
- Basic profiling runs of Bakeoff Problems
 - rocprof-sys providing time in each kernel
 - Using `/gpu/hip/shared` over `/gpu/hip/gen` shows clearer picture
 - Goal is to create full flame graphs with CPU call stack
 - Have identified hand-rolled kernels to be replaced with HIP utils
- As expected - user QFunction kernels are bulk of time
 - How can we ID performance issues for single kernel?
 - Hope to generate "best practices" guide for users

Sample Flamegraph



Currently getting kernel names without above CPU call stack

Sample rocprof-sys commands:

```
1 rocprof-sys-instrument -o ex2.inst -- ./build/ex2-surface
2 rocprof-sys-run -- ./ex2.inst -c /gpu/hip/shared -b 20000 -s 5000000
```

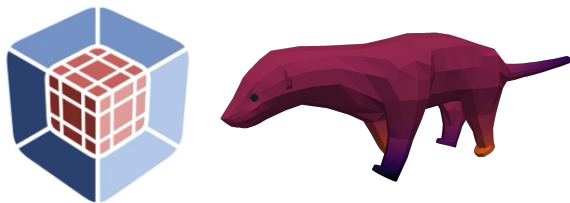
Day 3 Update

- libCEED and Ratel performance exploration on Odyssey
- Improved profiling runs of Bakeoff Problems
 - rocprof-sys-sample providing full flame graphs
 - Using
`rocprof-sys-sample -PTDH -I all --verbose 1 --freq 50 -- ...`
 - High number of people on the machine providing muddy perf data
- Replacing some hand-rolled replicas of BLAS operations
 - Pre ROCm 6.0/CUDA 12.0 lacks `*blas*_64` BLAS functions
- Unified memory changes running, showing good improvement

Final Outbrief

- Unified memory improves performance on APU hardware
 - Approximately 5% speedup with unified memory
- IDed and fixed minor internal performance issues
 - Prefer *BLAS calls over hand-rolled kernels where able
 - Prefer *memset() over hand-rolled kernels when zeroing
 - Reduce independent memory zeroing kernel calls in /gpu/*/shared
- Identified tools/processes for future profiling work
 - rocprof-sys-sample providing flame graphs
 - Analyzing individual QFunctions to improve, build guidelines

Questions?



libCEED Repo: <https://github.com/CEED/libCEED>

HONEE Repo: <https://gitlab.com/phypid/HONEE>

Ratel Repo: <https://gitlab.com/micromorph/ratel>

Grant: Predictive Science Academic Alliance Program (DE-NA0003962)

