Dwarfs on Accelerators

Enhancing OpenCL Benchmarking for Heterogeneous Computing Architectures

THE AUSTRALIAN NATIONAL UNIVERSITY

Beau Johnston and Josh Milthorpe
The Australian National University

August 13, 2018

Dwarfs on Accelerators

Enhancing OpenCL Benchmarking for Heterogeneous Computing Architectures

THE AUSTRALIAN NATIONAL UNIVERSITY

Beau Johnston and Josh Milthorpe
The Australian National University

August 13, 2018

Trends in Supercomputing – A view from the Top500

1. Summit – GV100
2. Sunway TaihuLight
3. Sierra – GV100
4. Tianhe-2A
5. ABCI – V100
6. Piz Daint – P100
7. Titan – K20x
8. Sequoia
9. Trinity – Phi
10. Cori – Phi

Trends in Supercomputing – A view from the Top500

1. Summit – GV100
2. Sunway TaihuLight
3. Sierra – GV100
4. Tianhe-2A
5. ABCI – V100
6. Piz Daint – P100
7. Titan – K20x
8. Sequoia
9. Trinity – Phi
10. Cori – Phi

- 7 / 10 use accelerators
- Newest is Summit – based on IBM Power9 with NVLINK and CAPI
- All devices have an OpenCL runtime
- Small but representative benchmark suite is needed

OpenCL – The Language of Heterogeneous Computing

- Open Computing Language (OpenCL) is an open standard.
- Allows computationally intensive codes – kernels – to be written once and run on any compliant accelerator.
- Most vendors are compliant to basic standards.
- Application code can be written directly in OpenCL, and
- Can be used as a back-end for higher level languages – OpenMP runtime implemented for TI Keystone II DSP architecture[1].
- Increased relevancy for FPGA programming

[1] Mitra, G. et al. 2014. Implementation and optimization of the OpenMP accelerator model for the TI Keystone II architecture. International workshop on openmp (2014), 202–214.

Extended Open Dwarfs Benchmark Suite

- Extended Open Dwarfs (EOD) Benchmark Suite
- Based off the OpenDwarfs benchmark suite[2]
- Benchmarks selected following diversity analysis and 13 Berkeley Dwarfs taxonomy
- Built in OpenCL
- Purpose of OpenDwarfs was a to characterize a diverse set of parallel applications and architectures using a common language, but had deficiencies. . .

[2]Krommydas, K. OpenDwarfs: Characterization of dwarf-based benchmarks on fixed and reconfigurable architectures. Journal of Signal Processing Systems, vol. 85, no. 3, pp. 373-392, 2016

EOD Extensions

- Selection of problem size is critically affects HPC benchmarking
- Highest impact on CPU architectures.
- A major contribution of the work is facilitating 4 different problem sizes for all applications presented in the suite.
- Selected according to levels of cache
    - **tiny** : $<$ 32 KiB L1
    - **small**: $<$ 256 KiB L2
    - **medium**: $<$ 8192 L3
    - **large**: $>$ 8192 L3

EOD Extensions – Continued

- Diverse:
  - 4 different problem sizes per application
  - Added applications – currently 11 and 37 kernels
  - Real applications sampled from Bioinformatics, Computational Biology, Computational Chemistry and other fields
- Reproducible: Minimum of 2 sec runs per benchmark
- Precise:
  - High resolution timers with LibSciBench
  - Reported with one cycle resolution and roughly 6 ns of overhead
  - Also allows collection of energy and hardware events
- Portable:
  - Based on an OpenCL backend
  - Tested on a wide range of hardware
  - Consistent tuning – i.e. workgroup size arguments

Applications

Table 1: List of Extended OpenDwarfs Applications and their respective dwarfs

Dwarf	Extended OpenDwarfs Application
Dense Linear Algebra	LU Decomposition
Sparse Linear Algebra	Compressed Sparse Row
Spectral Methods	DWT2D, FFT
N-Body Methods	Gemnoui
Structured Grid	Speckle Reducing Anisotropic Diffusion
Unstructured Grid	Computational Fluid Dynamics
Map Reduce	K-Means
Combinational Logic	Cyclic-Redundancy Check
Graph Traversal	Breadth First Search
Dynamic Programming	Smith-Waterman
Backtrack and Branch and Bound	N-Queens
Graphical Methods	Hidden Markov Models
Finite State Machines	Temporal Data Mining

Hardware

Name	Vendor	Type	Series	Core Count	Clock Frequency (MHz) (min/max/turbo)	Cache (KiB) (L1/L2/L3)	TDP (W)	Launch Date
Xeon E5-2697 v2	Intel	CPU	Ivy Bridge	24∗	1200/2700/3500	32/256/30720	130	Q3 2013
i7-6700K	Intel	CPU	Skylake	8∗	800/4000/4300	32/256/8192	91	Q3 2015
i5-3550	Intel	CPU	Ivy Bridge	4∗	1600/3380/3700	32/256/6144	77	Q2 2012
Titan X	Nvidia	GPU	Pascal	3584†	1417/1531/–	48/2048/–	250	Q3 2016
GTX 1080	Nvidia	GPU	Pascal	2560†	1607/1733/–	48/2048/–	180	Q2 2016
GTX 1080 Ti	Nvidia	GPU	Pascal	3584†	1480/1582/–	48/2048/–	250	Q1 2017
K20m	Nvidia	GPU	Kepler	2496†	706/–/–	64/1536/–	225	Q4 2012
K40m	Nvidia	GPU	Kepler	2880†	745/875/–	64/1536/–	235	Q4 2013
FirePro S9150	AMD	GPU	Hawaii	2816‖	900/–/–	16/1024/–	235	Q3 2014
HD 7970	AMD	GPU	Tahiti	2048‖	925/1010/–	16/768/–	250	Q4 2011
R9 290X	AMD	GPU	Hawaii	2816‖	1000/–/–	16/1024/–	250	Q3 2014
R9 295x2	AMD	GPU	Hawaii	5632‖	1018/–/–	16/1024/–	500	Q2 2014
R9 Fury X	AMD	GPU	Fuji	4096‖	1050/–/–	16/2048/–	273	Q2 2015
RX 480	AMD	GPU	Polaris	4096‖	1120/1266/–	16/2048/–	150	Q2 2016
Xeon Phi 7210	Intel	MIC	KNL	256‡	1300/1500/–	32/1024/–	215	Q2 2016

EOD Evaluation

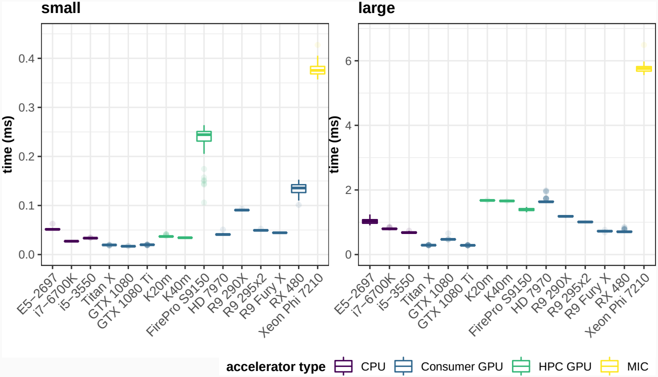


Figure 1: Comparison of performance on 2 sizes of csr application.

EOD Evaluation – Continued

- Just a sample of 1 of 11 applications
- Similar breakdown of 37 kernels
- 15 devices
- Time, performance events and energy (x50)
- Many more results and discussions presented in the full paper

What now?

- Small benchmark suite
- Wide diversity of scientific application codes
- Forms a large range of result times

What now?

- Small benchmark suite
- Wide diversity of scientific application codes
- Forms a large range of result times

But,

- How representative?
- Work since publishing uses EOD to study:
    1. workload characterization
    2. performance prediction
    3. scheduling

Workload characterisation with AIWC

- Architecture-Independent Workload Characterisation (AIWC)[3]
- Plugin for OclGrind – an Extensible OpenCL device simulator[4]
- Beta available – https://github.com/BeauJoh/Oclgrind – and will be merged into default OclGrind

---

[3] B. Johnston and J. Milthorpe, "OpenCL Performance Prediction using Architecture-Independent Features," arXiv:1805.04207 [cs.SE]

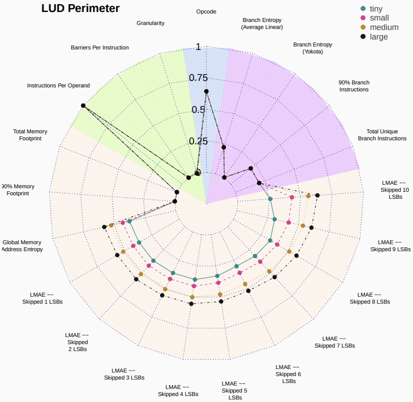
[4] J. Price and S. McIntosh-Smith, "Oclgrind: An extensible opencl device simulator," in Proceedings of the 3rd International Workshop on OpenCL, 2015, p. 12.

Overview of AIWC

- Simulation of OpenCL kernels occur on LLVM IR – SPIR
- AIWC tracks and measures hardware agnostic events
- Large number of metrics collected (34)
- Over a wide spectrum computation, thread communication and memory access patterns

AIWC Example

- 4 major types: Compute, Parallelism, Memory, Control
- Various number of metrics per type: based on statistical measurements – distributions, entropy and absolute counts

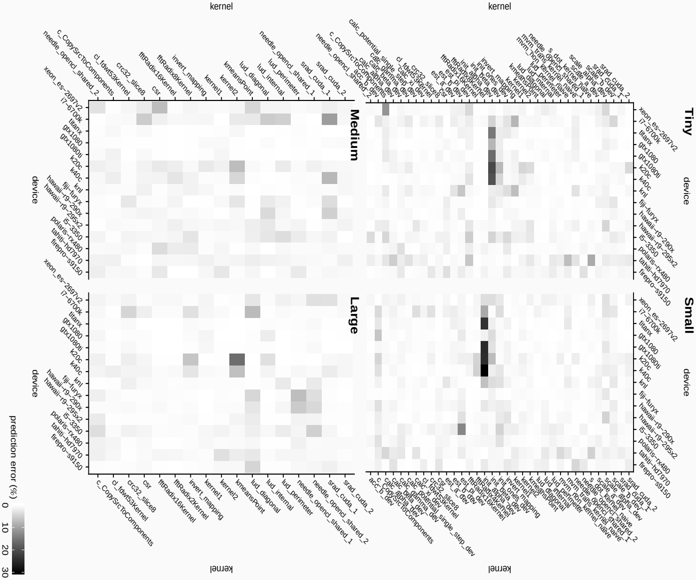


Performance Prediction – Combining EOD and AIWC

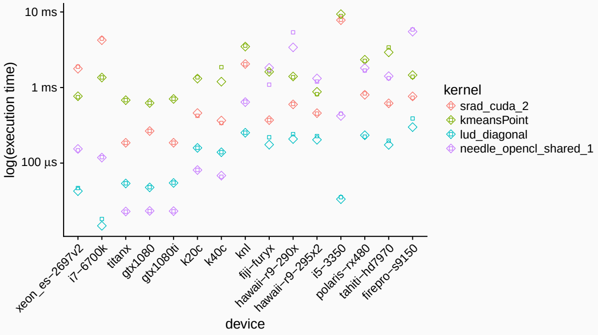
- Development of a regression model can now occur!
- AIWC's 34 metrics form input variables
- EOD response variables
- R language and Ranger – a Random Forest implementation – was used
- Performs recursive partitioning of high dimensional data
- Accepts 3 parameters:
    - num.trees – number of trees grown in forest – 10-10k by 500
    - mtry – number of features tried to split in a node – 1-34
    - min.node.size – minimal size per tree – 1-50
- Optimal model needs careful tuning[5]

[5] B. Johnston, G. Falzon and J. Milthorpe, "OpenCL Performance Prediction using Architecture-Independent Features,", High Performance Computing and Simulation (HPCS 2018), Jul 2018, Orléans, France.

Performance Prediction Example



Scheduling



- 4 random kernels.
- square → mean measured time
- diamond → mean predicted time
- order is important!

Conclusions

- Completed essential curation of the OpenDwarfs benchmark suite:
    - Increased application diversity
    - Tested on 15 devices
    - High precision measurements of time, energy and hardware events
- Required for:
    - Evaluating OpenCL on wide-range of parallel architectures
    - Workload characterization
    - Performance prediction
    - Scheduling

Open Questions and Next Steps

- Do you have real applications to add to EOD? Perhaps in biological, numerical computing or stochastic simulations?
- These workloads can test the extremes of EOD and prediction coverage? Poor predictions finds holes in the model → missing benchmark in EOD
- What about new hardware – Especially FPGA? Spare hours on a super-computer?
- Large applications that could benefit from using the predictive model?

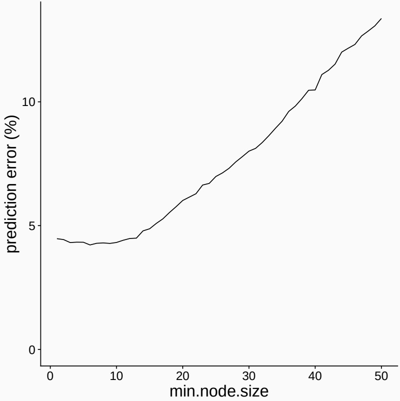
- We should collaborate!

Thanks

- The University of Bristol's High Performance Computing Research group for the use of "The Zoo" Research cluster
- Oracle
- ANU VC Travel Grant
- You!

beau.johnston@anu.edu.au

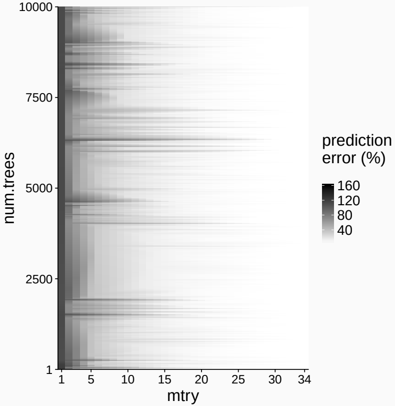
min.node.size



- Full coverage of min.node.size with fixed tuning parameters: num.trees = 300 and mtry = 30
- Smallest out-of-bag prediction error for values < 15
- Selection made to fix min.node.size = 9

num.trees and mtry

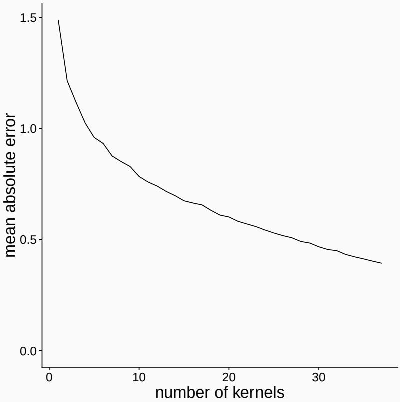
- optim_sa function used to find global minimum
- Full coverage achieved – 4 outer-most points and 8 random starting internal points
- intermediate results used and interpolation performed – using akima
- Model performance varies significantly for last 2 variables
- mtry $>$ 25, offers good fit
- num.trees less impact – fewer



Choosing Parameters for the Future

- num.trees=500, mtry=32, and min.node.size=9 look good
- train on a random selection of N kernels and test on remainder
- see paper for details but final values are num.trees = 505, mtry = 30 and min.node.size = 9

Increased Training Data



- How many kernels to add for training – what's enough?
- Another study performed to see how error changes w.r.t. number of kernels in training
- Uses random selection for each random count – again see paper for full details
- Error tapers off for more kernels!
- gradient still significant at 37 kernels → could still benefit from more.

Prediction Accuracy