FC with RG – Near Memory

HPCG: Alexander Contratti, Yiwen Liu, Jeremy Wang, Yongnuo Yang
GraphBLAS: Yu Pan, Michael Nguyen, Saatvik Agrawal, Mitchell Teunissen, Mukund Chidambaram

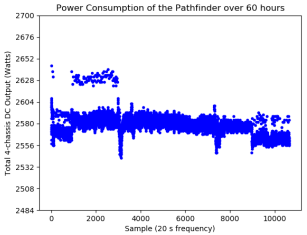
HPCG Project

Goals

- Gather data for multi-node and multi-chassis runs of the HPCG benchmark on the Pathfinder and extrapolate to draw conclusions for performance scalability
- Analyze power efficiency of the Pathfinder while running benchmarks

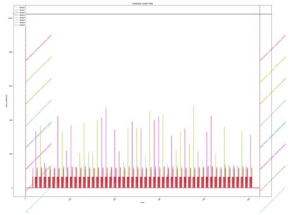
Benchmark Results

- Constraining iterations for convergence increased performance
- Power analysis inconclusive- more testing needed



Profiling Results

- Consistent performance patterns across simulation and hardware profiling
- Raw performance is still quite low



Lessons Learned

- Make the most of opportunities to run cleanly on hardware with better project organization, such as by queueing several benchmark variations that need to be run
- Start analysis alongside data gathering to better determine what further data needs to be gathered (power consumption data, larger problem sizes)

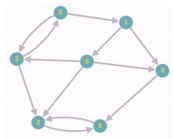
GraphBLAS Project

Goals

- Create a more accessible introduction to GraphBLAS than the preexisting notebook part of the Lucata tutorials.
- Do initial comparisons and profiling of x86 GraphBLAS, Lucata GraphBLAS, and Lucata Beedrill with emusim.x.

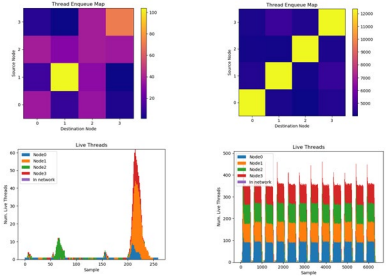
Notebook Progress

- Finished notebook introducing GraphBLAS, following the Mattson tutorial to eventually complete a BFS and learn the basics of GraphBLAS.
- The Notebook uses Python for visualization but has the C code for the exercises as well.



Result Analysis

- Benchmarks of lucata-beedrill are optimized towards different aspects of Lucata architecture.



Lessons Learned / Next Steps

- GraphBLAS concepts are difficult, take the learning process slowly and integrate Cilk gradually.
- Run lucata-beedrill and LucataGraphBLAS benchmarks on hardware