The Emergence of Heterogeneous Systems

- Increasingly demanding power/performance goals require designers to utilize heterogeneous components
- GPUs offer high performance-per-watt, but they are difficult to design
- Accelerators can substantially improve streaming application performance
- The problem: Heterogeneous systems are difficult to design and optimize
- Must account for computation AND communication
- Must account for performance AND power
- Existing automated design space exploration approaches often cannot handle real-system variance and subspace-induced nonlinearity

Approach 1: Starchart

(Publication) Starchart: Hardware and Software Optimization Using Recursive Partitioning Regression Trees, Wenhai Jia et al., Parallel Architectures and Compilation Techniques (PACT) 2013

Motivation

- GPU design spaces contain complex “performance cliffs” and “subspaces”
- Existing design space exploration approaches are insufficient

Our Work: Automated Design Space Partitioning

- Partition-based regression tree approach is powerful and robust
- Handles real-system measurement variance
- Handles “performance cliffs” and “subspaces” common for GPU systems
- Applicable to multiple metrics and CPUs
- Tree visualizations are intuitive
- For GPU users, tool builders and hardware designers
  - Optimize designs within or across different platforms
  - Reveal power/performance trade-offs
  - Measure a program’s input sensitivity
  - >300X speed-up in design space exploration

Starchart Method

- Step 1: Uniformly and randomly sample N designs from the whole space
- Step 2: Apply an iterative algorithm to recursively partition the samples
- Step 3: Use resulting tree representations to solve subspace-based problems

Approach 2: Designing Communication Accelerators

(Work in Progress)

Motivation

- Accelerator design is not just about computation!
- Moving data to/from the accelerator from/to the cores or memory can consume substantial amount of time and energy
- It is necessary to think about both communication and computation when utilizing an accelerator

Communication-Aware Accelerator Architecture

- An accelerator consists of three components: data load unit, computation unit, data store unit

Conclusion

- Heterogeneity calls for systematic and novel design space analysis techniques
- Automated regression tree methods can solve real-system power/performance optimization problems with >300X productivity speedup
- Communication-aware accelerators balance communication with computation to significantly reduce wasted energy consumption