Load Balancing in a Changing World: Dealing with Heterogeneity and Performance Variability

Michael Boyer, Shuai Che, Kevin Skadron
University of Virginia
boyer@cs.virginia.edu

Nuwan Jayasena
AMD Research

GRC Task 1972.001
Heterogeneous Load Balancing: Motivation
Heterogeneous Load Balancing: Motivation

- Integrated GPU only: 0.94 s
- Discrete GPU only: 0.67 s
- Both GPUs: 0.40 s
Dynamic Load Balancing: Motivation

- Device performance is not fixed
- Example: Host memory contention

![Diagram showing host memory, CPU, integrated GPU, and discrete GPU connections]
Dynamic Load Balancing: Motivation

- Performance ratio can change at runtime
Dynamic Load Balancing: Motivation

- Optimal partition can change significantly
Dynamic Scheduling Algorithm

Device 1

Device 2

Diagram shows the scheduling of devices over time, with Device 1 and Device 2 competing for resources.
Dynamic Scheduling: Example

- **Data Transfer**
- **Kernel Execution**

### Discrete GPU
- Data Transfer: 3.1%, 6.2%, 12.5%, 25.0%, 26.6%
- Kernel Execution: 17.2%

### Integrated GPU
- Data Transfer: 3.1%, 6.2%
- Kernel Execution: 12.5%, 25.0%, 17.2%
Evaluation

Benchmarks: OpenCL applications from AMD APP SDK / Rodinia benchmark suite

• Comparison among:
  – Native execution on fastest device
  – Best static partition
  – Proposed dynamic load balancer

• Metric: total time
  – Kernel execution time + data transfer time
Case Study: Matrix Multiplication

• Dynamic: more consistent performance
Case Study: Matrix Multiplication

- Dynamic: never slower than native
Results: Overall

Overall Speedup vs. Native Execution

K-means | Black-Scholes | Mersenne Twister | Matrix Mult. | DCT | FFT | Average

Static | Dynamic

Application

0.0 | 0.2 | 0.4 | 0.6 | 0.8 | 1.0 | 1.2 | 1.4 | 1.6
Dealing with Unresponsive Devices

- Host Memory (DRAM)
- CPU
- Integrated GPU
- GPU Memory
- Discrete GPU

Note: Discrete GPU is marked with a red cross.
Results: Unresponsive Devices

- Black-Scholes
- DCT
- K-means
- Matrix Multiply
- Mersenne Twister
- FFT

- All Work to Discrete GPU
- All Work to Integrated GPU
Conclusions

• Load balancing can provide significant performance benefits
• But optimal partition can change at runtime!

• Our dynamic load balancer provides:
  – Better performance on average than statics
  – Ability to deal with blocked devices

• Choice between dynamic or static depends on the amount of performance variation
Tech Transfer

- Collaboration with Nuwan Jayasena at AMD Research
  - Weekly meetings
  - Jointly developed software infrastructure

- Presented at AMD Fusion Developer Summit 2011

- Goal: incorporate technology into AMD software stack
Questions?

Visit poster 14.3 during TechFair