Introduction to Green Computing

- What do we mean by Green Computing?
- Why Green Computing?
- Measuring “greenness”
- Research into energy consumption reduction

What do we mean by Green Computing?

The green computing movement is a multifaceted global effort to reduce energy consumption and to promote sustainable development in the IT world. [Patrick Kurp, Green computing in Communications of the ACM, 2008]

Why Green Computing?

- Heat dissipation problems
- High energy bills
- Growing environmental impact

Measuring “greenness”

- Non-standard metrics
  - Energy (Joules)
  - Power (Watts)
  - Energy per-instructions (Joules / No. instructions)
  - Energy-delay*N-product (Joules * seconds*N)
  - Performance*N/Watt (No. instructions / second*N/Watt)
- Standard metrics
  - Data centers: Power Usage Effectiveness metric (The Green Grid consortium)
  - Servers: asl_ops/Watt metric (SPEC consortium)
Research into energy consumption reduction

Maximizing Power Efficiency with Asymmetric Multicore Systems
Fedorova et al., Communications of the ACM, 2009

• Outline
  – Asymmetric multicore processors
  – Scheduling for parallel and serial applications
  – Scheduling for CPU- and memory-intensive applications

Asymmetric multicore processors
• What makes a multicore asymmetric?
  – a few powerful cores (high clock freq., complex pipelines, OoO execution)
  – many simple cores (low clock freq., simple pipeline, low power requirement)
• Homogeneous ISA AMP
  – the same binary code can run on both types of cores
• Heterogeneous ISA AMP
  – code compiled separately for each type of core
    – examples: IBM Cell, Intel Larrabee

Efficient utilization of AMPs
• Efficient mapping of threads/workloads
  – parallel applications
    – serial part → complex cores
    – scalable parallel part → simple cores
  – microarchitectural characteristics of workloads
    – CPU intensive applications → complex cores
    – memory intensive applications → simple cores

Sequential vs. parallel characteristics
• Sequential programs
  – high degree of ILP
  – can utilize features of a complex core (super-scalar pipeline, OoO execution, complex branch prediction)
• Parallel programs
  – high number of parallel threads/tasks (compensates for low ILP and masks memory delays)
• Having both complex and simple cores, give AMPs applicability for wider range of applications

Parallelism-aware scheduling
• Goal: improve overall system efficiency (not the performance of a particular application)
• Idea: assign sequential applications/phases to run on the complex cores
• Does NOT provide fairness
Challenges of PA scheduling

- Detecting serial and parallel phases
  - Limited scalability of threads can yield wrong solutions
- Thread migration overhead
  - Migration across memory domains is expensive
  - Scheduler must be topology aware

“Heterogeneity”-aware scheduling

- Goal: improve overall system efficiency
- Idea:
  - CPU-intensive applications/phases → complex cores
  - Memory-intensive applications/phases → simple cores
- Inherently unfair

Challenges of HA scheduling

- Classifying threads/phases as CPU- or memory-bound
  - Two approaches presented: direct measurement and modeling
- Long execution time (direct measurement approach) or need of offline information (modeling approach)

Summary

- Green Computing focuses on improving energy-efficiency and sustainable development in the IT world
- AMPs promise higher energy-efficiency than symmetric processors
-Schedulers must be designed to take advantage of the asymmetric hardware

References

- Kirk W. Cameron, The road to greener IT pastures in IEEE Computer, 2009
- Dan Herrick and Mark Ritschard, Greening your computing technology, the near and far perspectives in Proceedings of the 37th ACM SIGUCCS, 2009
- Luiz A. Barroso, The price of performance in ACM Queue, 2005