Cool Supercomputing: Keepin' it Real

Scott Pakin

Applied Computer Science Group

Los Alamos National Laboratory

14 November 2012

Questions from the Organizers

- **What is good and bad about the current state of the art in tools and techniques for optimizing power on large-scale systems?**
- **How much more needs to be done to make power a first-class citizen for future extreme-scale systems?**

Current State of the Art

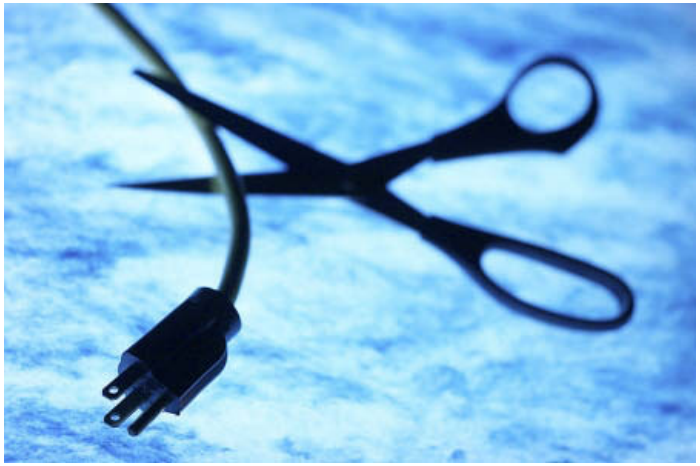
- **Good**
  - Lots of work being done to manage power throughout the system
  - Algorithms, compilers, job schedulers, operating systems, architecture
- **Bad**
  - Most of this work is totally oblivious to reality

The Wrong Way to Think about Power

- **Researcher: "If you use my {run-time system, compiler, language, etc.}, your application will draw *x*% less power *y*% of the time and degrade performance by only *z*%"**
- **Financial analyst (gov't): "Our budgets don't carry over across fiscal years; drawing *x*% less power doesn't save us any money"**
- **Facilities engineer: "We have to allocate infrastructure for worst-case usage; *y* < 100% is useless"**
- **User: "*What?!?* You're degrading my performance by *z*%. What did I ever do to *you*?**

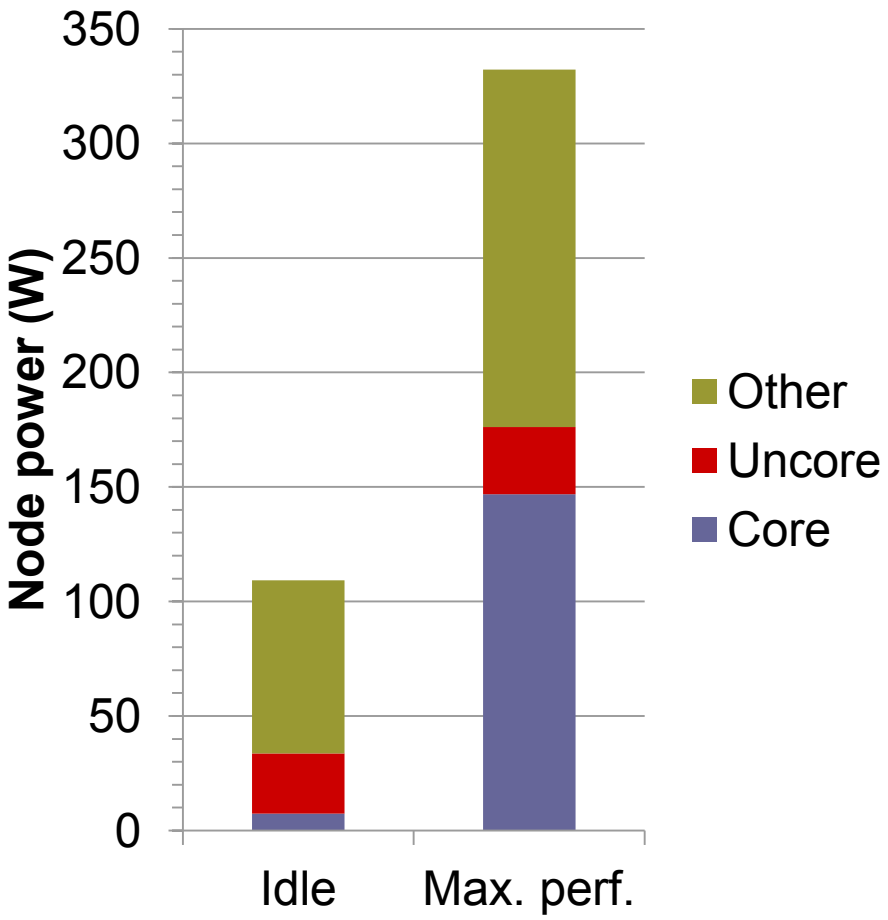
The Disconnect

- **Users and application developers don't care about power efficiency**
  - They don't pay for power
  - They barely know how to handle scalability, let alone programming for power efficiency
  - It's not worth their time to restructure code for power efficiency
  - Preferred metric: $ED^{\infty}P$
- **Race-to-halt does better than most researchers give it credit for**
  - DRAM, power supplies, I/O devices, various other components draw power whether used or not
  - Implication is that energy is minimized when these are used for as little time as possible

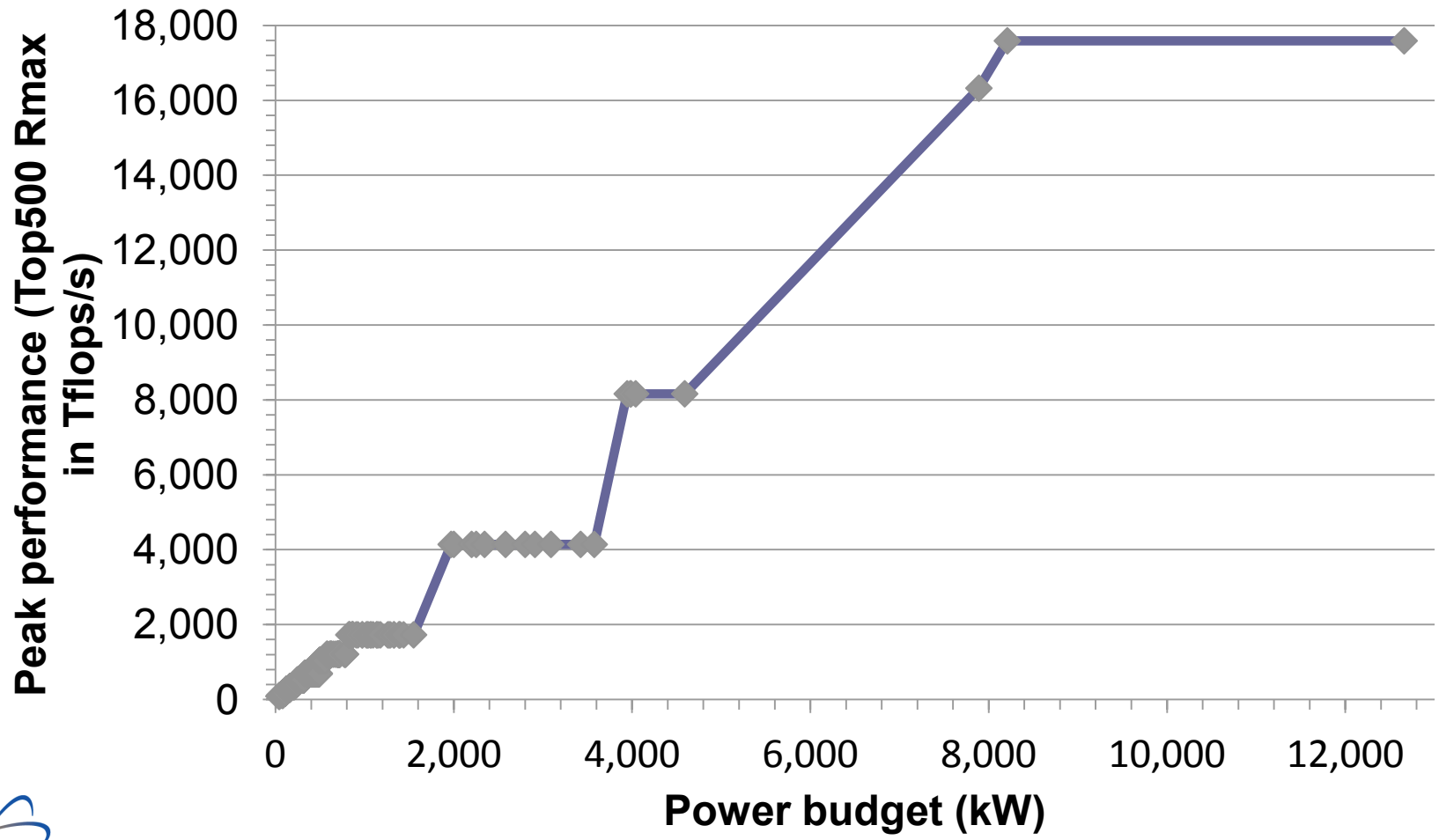


Race to Halt is Hard to Beat

- **Power data for xRAGE on a 150-node Sandy Bridge + InfiniBand cluster**
  - 109 W/node idle vs. 332 W/node at max. perf.
- **Best one can do**
  - Reduce power by 2/3
  - Increase run time by <2/3 to come out ahead energy-wise
  - Possible? Doubtful
- **Change of goals**
  - Reduce baseline power draw
  - Get most performance for a *given* power budget



The Right Way to Think about Power



Making Power a First-Class Citizen

- **Necessary pain at extreme scale**
  - Applications are granted a maximum power draw for the course of their execution
- **Pain relief (naproxen)**
  - Give application developers the mechanisms needed to stay within their budget
  - Libraries, language constructs, etc.
- **Pain relief (homeopathic)**
  - Throttle performance if application tries to exceed its power cap
  - *Bonus points*: Coschedule high- and low-power applications
  - (You can go over budget if you find a patsy who can stay under budget)

Closing Thoughts

"Gotta give us what we want.
Gotta give us what we need.

…

To revolutionize make a change,
Nothin's strange

…

What we need is awareness;
we can't get careless.

…

Lemme hear you say,
***Fight the power*."**

— Public Enemy,
Fight the Power