



Dynamic Voltage and Frequency Scaling: characterization and usage in the real-world

Energy? Why do we care?

- Tianhe-2: 18MW
 - Electricity = \$12,000,000 / year



Save 10% = save \$1,200,000 / year!

+ technical and ecological concerns



What can we do about it?

- Hardware improvements
 - Great savings
 - Not today
- Software solutions
 - Today
 - Better exploit current hardware
 - Increase energy efficiency

Software solutions at UVSQ

• Extreme server consolidation

• Power modeling

- Dynamic Voltage and Frequency Scaling (DVFS)
 - DVFS potential
 - Runtime DVFS controller

DVFS

- CPU frequency transitioning
 - Available on all recent processors
- Lower frequency = lower power consumption
 - But also lower speed...

-e = Pxt

DVFS is hard



• FoREST, runtime DVFS controller

- Basic idea:
 - Periodically check the impact of frequency transitions
 - Pick the frequency leading to minimal energy

- e = P x t
 - Set a frequency for a short period of time
 - Check IPS (estimates t)
 - Check power consumption
 - Deduce e
- But...

- P cannot be measured on short period of times
 - Embedded CPU probes have a limited rate
- P(f1) / P(f2) is roughly program independent
 - P commonly simplified as k x Pdynamic
 - Pdynamic = aCV^2f

- Measure P(f) offline for all f
 - Deduce P(f) / P(fmax) for all programs
- e = P x t
 - Regularly test f for a short period
 - Measure IPS(f) / IPS(fmax) online
 - Compute e(f) / e(fmax)
 - IPS(fmax) / IPS(f) x P(f)/P(fmax)
 - Energy gain achieved when using f rather than fmax
 - Chose the f leading to maximal energy gains





Extra features

a.k.a. living in the real world

Real-world constraints

- Multicore processors
 - Shared frequency domains
 - Supported by FoREST

- Measurements over theoretical models
 - Many DVFS controllers outdated with SNB

In a hurry?

- Energy savings constrained by a maximal slowdown
 - Your needs, your choice
- FoREST determines the optimum within your constraints
 - Exploits the CPI measurements
 - Automatically determines the best slowdown

System or CPU?

- Saving CPU energy != saving energy
 - CPU gains must be > than system losses
- FoREST considers the system power
 - Estimation: constant average consumption ~50W
 - Could be extended to exploit power probes

CPU savings – 5% slowdown



System savings – 5% slowdown



Execution time



CPU savings – 100% slowdown



System savings – 100% slowdown



How good are those results?

UtoPeak

- How much energy can we save with DVFS?
 - For a given program and computer

What is the frequency sequence leading to minimal energy consumption?

- Profile the program for every frequency
 - Periodically measure
 - Energy consumption
 - # of executed instructions
- Lists of profiling samples for every frequency
 - Same duration but different instructions
 - Cannot pick the sequence of best samples



- Change scale from time to instruction count
 - Use # of executed instructions
 - Instruction count does not depend on the frequency
- Normalized samples are comparable
 - Iterate over samples with minimal energy consumption
 - Predict execution time and energy consumption



• Example output:

Timestep	1	2	3	4
Frequency	F1	F3	F3	F2

- Predicted energy: 487 J
- Predicted execution time: 5.6 s

Check prediction

- Run the program
- Apply the best frequency sequence
- Measure energy consumption and compare to prediction

Precision (SPEC CPU2006)



Precision (NAS OMP 3.0)



Maximal CPU savings (SPEC)



Maximal CPU savings (NAS)



Conclusion

- UtoPeak estimates the potential for DVFS
 - Is DVFS profitable for my program?
- FoREST precisely controls the frequencies
 - Achieves the energy savings
 - UtoPeak certifies it is close to optimal
- Prototypes working for Intel x86_64
 - Xeon Phi on its way

Perspectives

- Go towards multi-node:
 - Specifically handle MPI programs
 - Perform greater savings at the cluster level
 - Collaboration with Université de Strasbourg
- Support other architectures
 - Integrate power models to replace probes?

Thank you for your attention!

Questions?

http://code.google.com/p/forest-dvfs

http://code.google.com/p/utopeak

