Improving the Energy Efficiency of Software Systems for Multi-Core Architectures

Maxime Colmant

ADEME - University Lille 1 - INRIA

maxime.colmant@inria.fr

2014-12-09

Outline

Introduction

Increasing usage of IT devices

- Estimated at 0.83 $GtCO_2$ in 2007, 1.43 $GtCO_2$ in 2020 (6%) [ClimateGroup:2008]

Complexity of modern processors

- Limited power-aware interfaces [Hahnel:2012, Zhai:2014]

Software power estimation, a cornerstone

- Identify the largest power consumers, make informed decisions
- Architecture-agnostic solution is needed

Motivation

- In general, performance $>$ energy efficiency
- ICT has an huge impact on the world CO_2 emissions
- Main power consumer: processor (increasingly complex)
- Multi-core CPU are widely used nowadays
- On the hardware side (*e.g.* SMT, DVFS, C-states)
- On the software side?

Software power efficiency

Can play a deterministic role!

Power monitoring

Hardware-centric approach

- Coarse-grained
- Expensive

Software-centric approach

- Fine-grained
- Awkward

Software-centric approach

Needs

- Efficient and accurate power models
- Trade-off between accuracy/overhead

Existing solutions

- Specific softwares and architectures [Bertran:2010, Bircher:2007, Spiliopoulos:2012, Zhai:2014]
- As an example, Intel with RAPL [Zhai:2014, Hahnel:2012]

Our goal

- Provide an architecture-agnostic solution
- Identify green patterns as methodological guidelines

Research methodology

Power models

- Mostly linear [Mccullough:2011], trustfully represent the power consumption
- Component metrics are gathered with power consumption

CPU metrics

- CPU load [Versick:2013]
- Hardware Performance Counters (HPC) [Bertran:2010, Bircher:2007, Lim:2010, Spiliopoulos:2012, Zhai:2014]

HPCs

- Architecture-dependent
- Considered by state-of-the-art as the most accurate metrics

Assessments

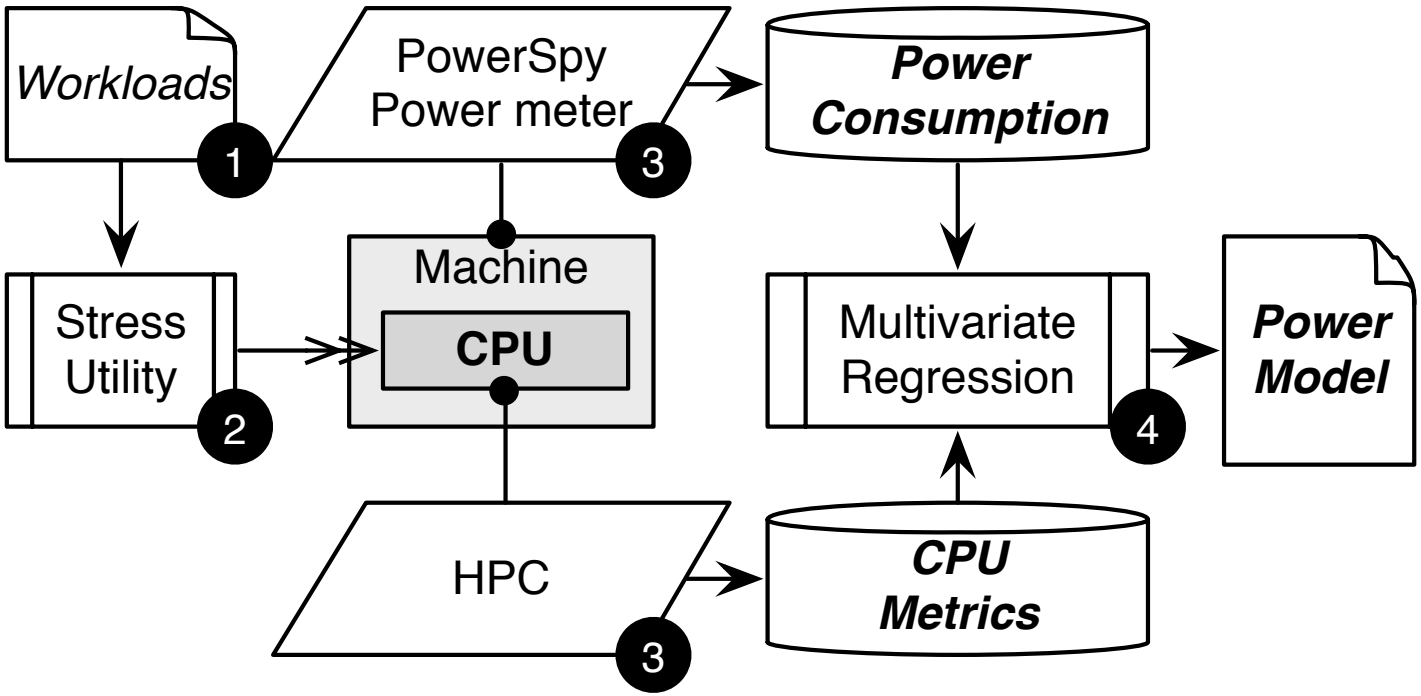
Problems

- Most of power models are architecture and software dependents
- Lack of informations, difficult to adapt and to reproduce

Solutions

- HPC criteria selection: Availability, exploitation overhead, evolution
- Architecture-agnostic power models

Learning the energy profile of modern processors



Selected HPC

- `instructions (i)`, `cache-references (r)`, `cache-misses (m)`

Example of power model, Intel Core i3 2120

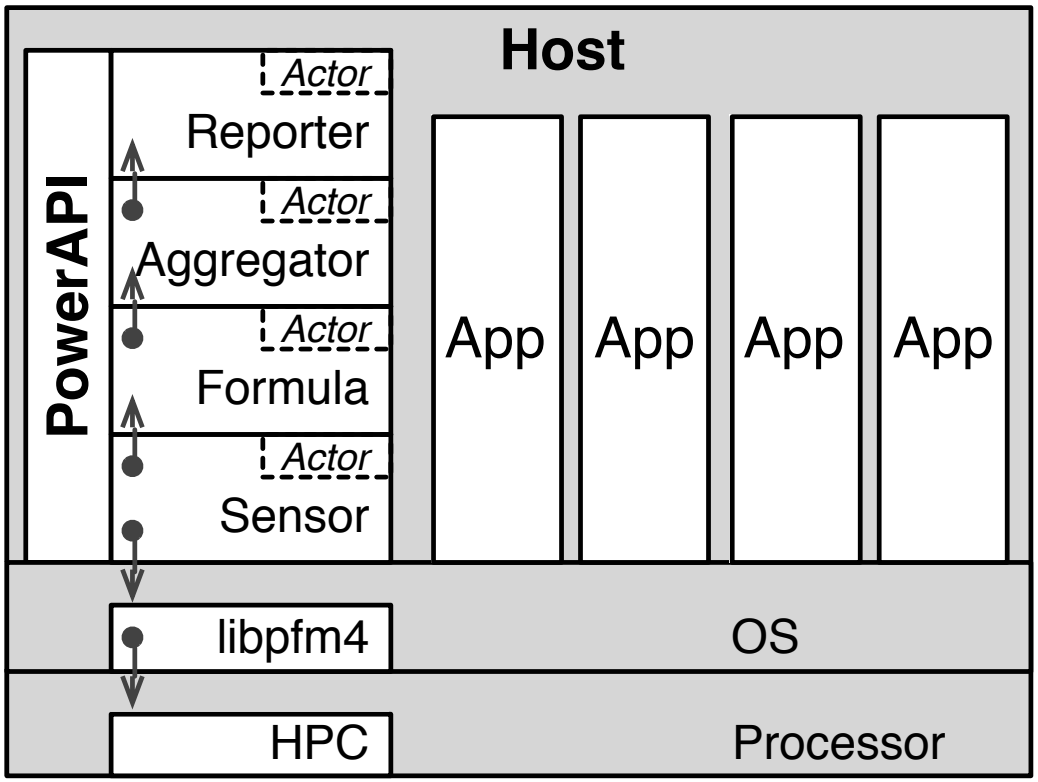
Overall formula

$$Power_{i3} = 31.48 + \sum_{f=1.6}^{3.30} Power_f \tag{1}$$

Frequency formula

$$Power_{3.30} = \frac{2.22 \cdot i}{10^9} + \frac{2.48 \cdot r}{10^8} + \frac{1.87 \cdot m}{10^7} \tag{2}$$

PowerAPI

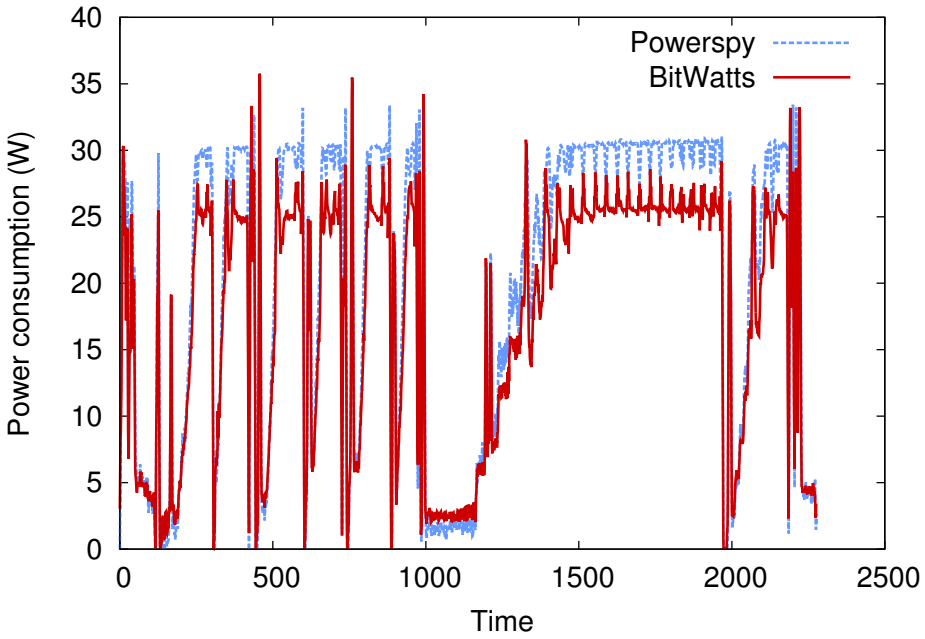


- Actor programming model (Scala / Akka)
- Modular & scalable Middleware
- Real-time power estimation

Intel Core i3 2120 specifications.

Vendor	**Intel**
Processor	**i3**
Model	2120
Design	4 threads
Frequency	3.30 GHz
TDP	65 W
SpeedStep (DVFS)	✓
HyperThreading (SMT)	✓
TurboBoost (Overclocking)	✗
C-states (Idle states)	✓
L1 cache	64 KB / core
L2 cache	256 KB / core
L3 cache	3 MB

Preliminary experiment on SPECJBB2013



- [Bertran:2010]: Average error of 4.63% on 6 applications (SPEC-CPU 2006), Intel Core2Duo
- [Zhai:2014]: Average error of 7.5% (private Google benchmarks), Intel Sandy-bridge

Conclusion

A Middleware to build software-defined power meters

- High-level API, modular and scalable
- Processor agnostic solution
- Sampling, power-model inference
- Real-time power estimation

Outlook

- Virtualization
- Identify automatically the HPCs
- Heuristics

Bibliography I

Ramon Bertran, Marc Gonzalez, Xavier Martorell, Nacho Navarro, and Eduard Ayguade.
Decomposable and responsive power models for multicore processors using performance counters.
In *Proceedings of the 24th ACM International Conference on Supercomputing*, ICS '10, pages 147–158, New York, NY, USA, 2010. ACM.

William Lloyd Bircher and Lizy K John.
Complete system power estimation: A trickle-down approach based on performance events.

In *Performance Analysis of Systems & Software, 2007. ISPASS 2007. IEEE International Symposium on*, pages 158–168. IEEE, 2007.

The Climate Group.
SMART 2020: Enabling the low carbon economy in the information age.

Marcus Hähnel, Björn Döbel, Marcus Völp, and Hermann Härtig.
Measuring energy consumption for short code paths using rapl.
SIGMETRICS Perform. Eval. Rev., 40(3):13–17, January 2012.

Min Yeol Lim, Allan Porterfield, and Robert Fowler.
Softpower: Fine-grain power estimations using performance counters.
In *Proceedings of the 19th ACM International Symposium on High Performance Distributed Computing*, HPDC '10, pages 308–311, New York, NY, USA, 2010. ACM.

Bibliography II


John C McCullough, Yuvraj Agarwal, Jaideep Chandrashekar, Sathyanarayan Kuppuswamy, Alex C Snoeren, and Rajesh K Gupta.
Evaluating the effectiveness of model-based power characterization.
In *USENIX Annual Technical Conference*, 2011.


Vasileios Spiliopoulos, Andreas Sembrant, and Stefanos Kaxiras.
Power-sleuth: A tool for investigating your program's power behavior.
In *MASCOTS*, pages 241–250, 2012.


Daniel Versick, Ingolf Wassmann, and Djamshid Tavangarian.
Power consumption estimation of CPU and peripheral components in virtual machines.
SIGAPP Appl. Comput. Rev., 13(3):17–25, September 2013.


Yan Zhai, Xiao Zhang, Stephane Eranian, Lingjia Tang, and Jason Mars.
Happy: Hyperthread-aware power profiling dynamically.
In *2014 USENIX Annual Technical Conference (USENIX ATC 14)*, pages 211–217, Philadelphia, PA, June 2014. USENIX Association.