Energy Conservation and Thermal Management in High-Performance Server Architectures

Adam Wade Lewis
The Center for Advanced Computer Studies
The University of Louisiana at Lafayette

The Center for Advanced Computer Studies

UNIVERSITY OF LOUISIANA Lafayette

Agenda

- Background and Related Work
- Chaotic Attractor Predictors
- A Thermal Aware Scheduler
- Scheduler Evaluation and Results
- Summary

Green computing is...
efficient & effective computing
with little or no impact on the
environment

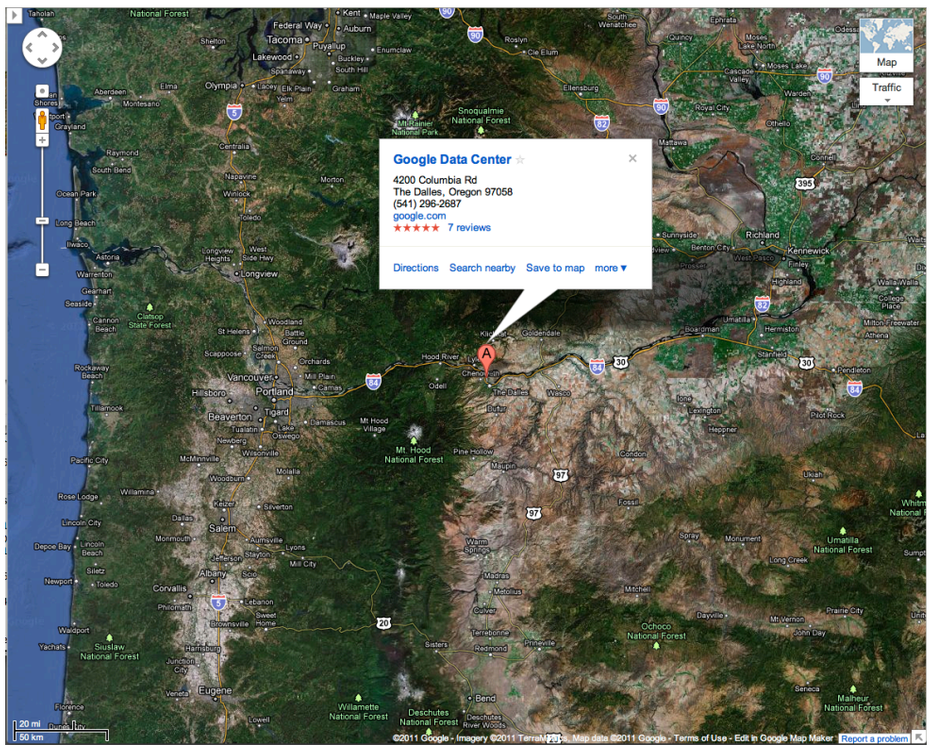
Green computing comes from
product longevity
effective recycling
power management
optimized software

The Center for Advanced Computer Studies

UNIVERSITY OF LOUISIANA Lafayette

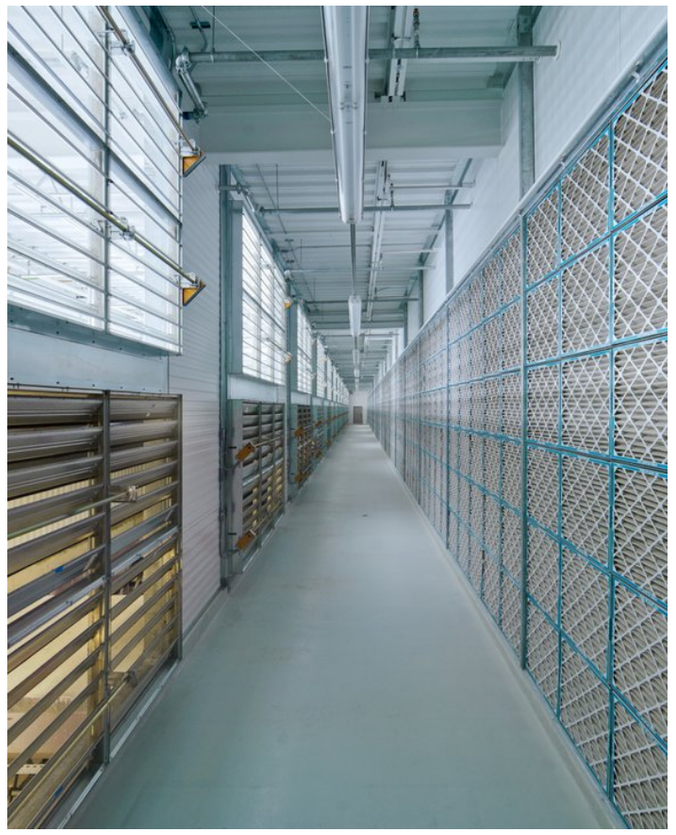
What do these pictures tell us?

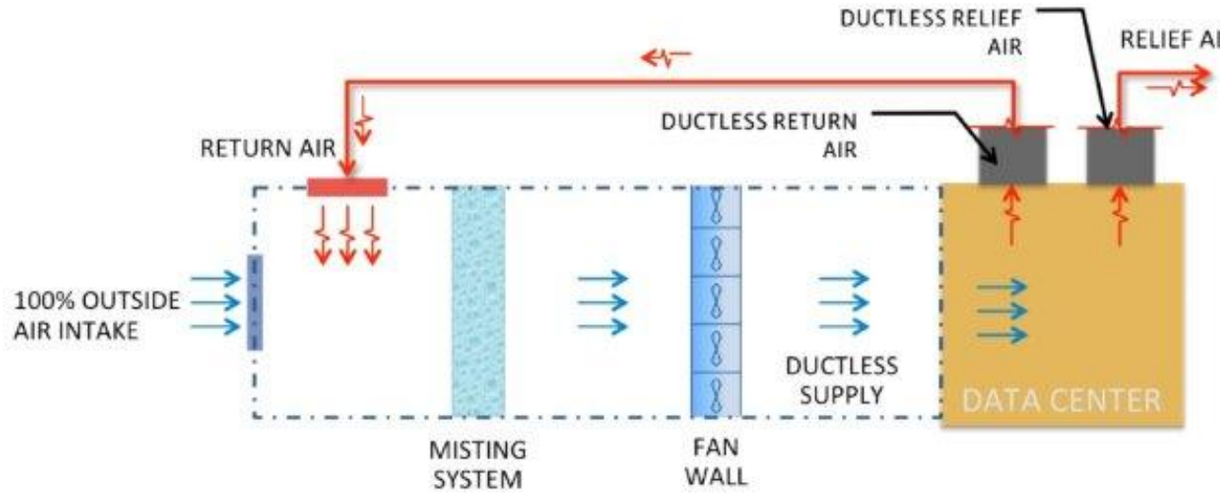
(c) The New York Times, June 14, 2006



- A major industrial facility
- In a remote location
- With heavy power and cooling demands

Data Centers: Where ME meets CS

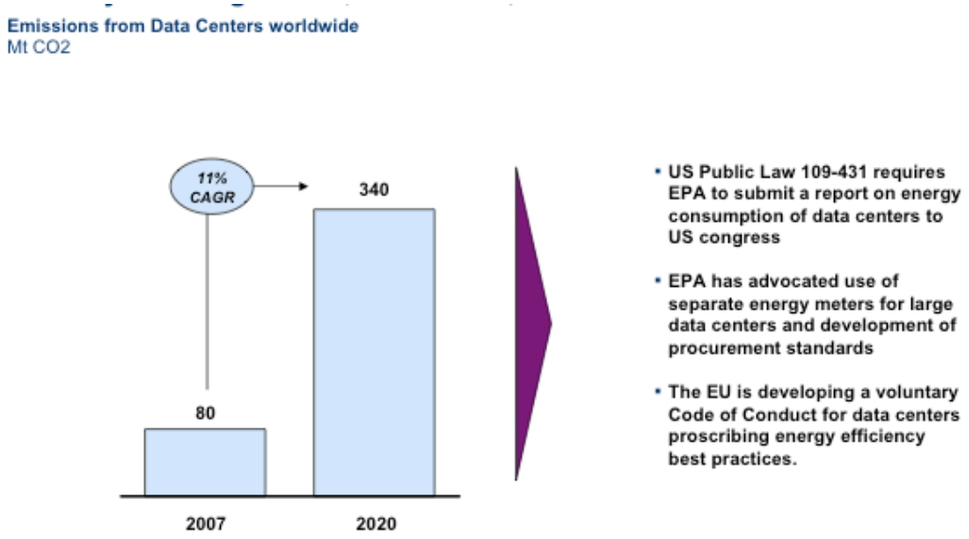




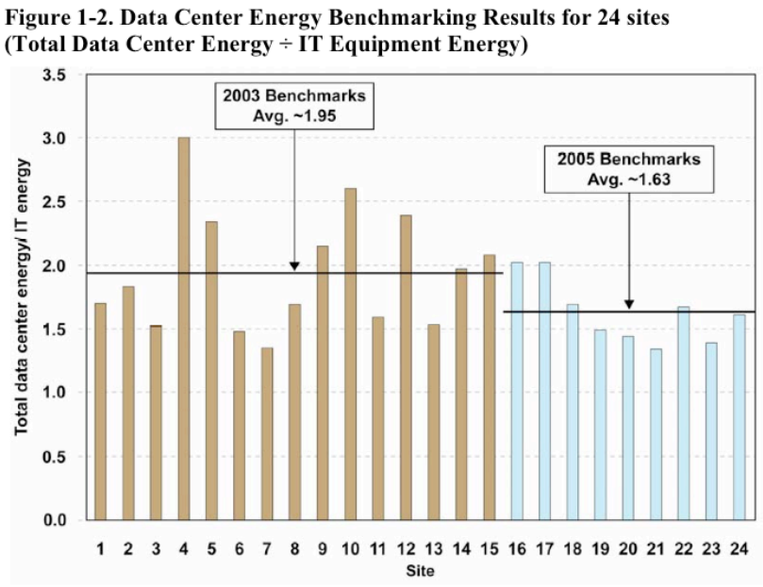
Source: Facebook, Open Compute Project

DOE FY2011 SBIR & STTR Program Solicitation:
"Standardized Energy Measurement Interfaces, Integration with Facility Infrastructure, and Energy-Aware Algorithms."

Need for data center energy efficiency



Source: McKinsey & Company 2008



Source: EPA 2008

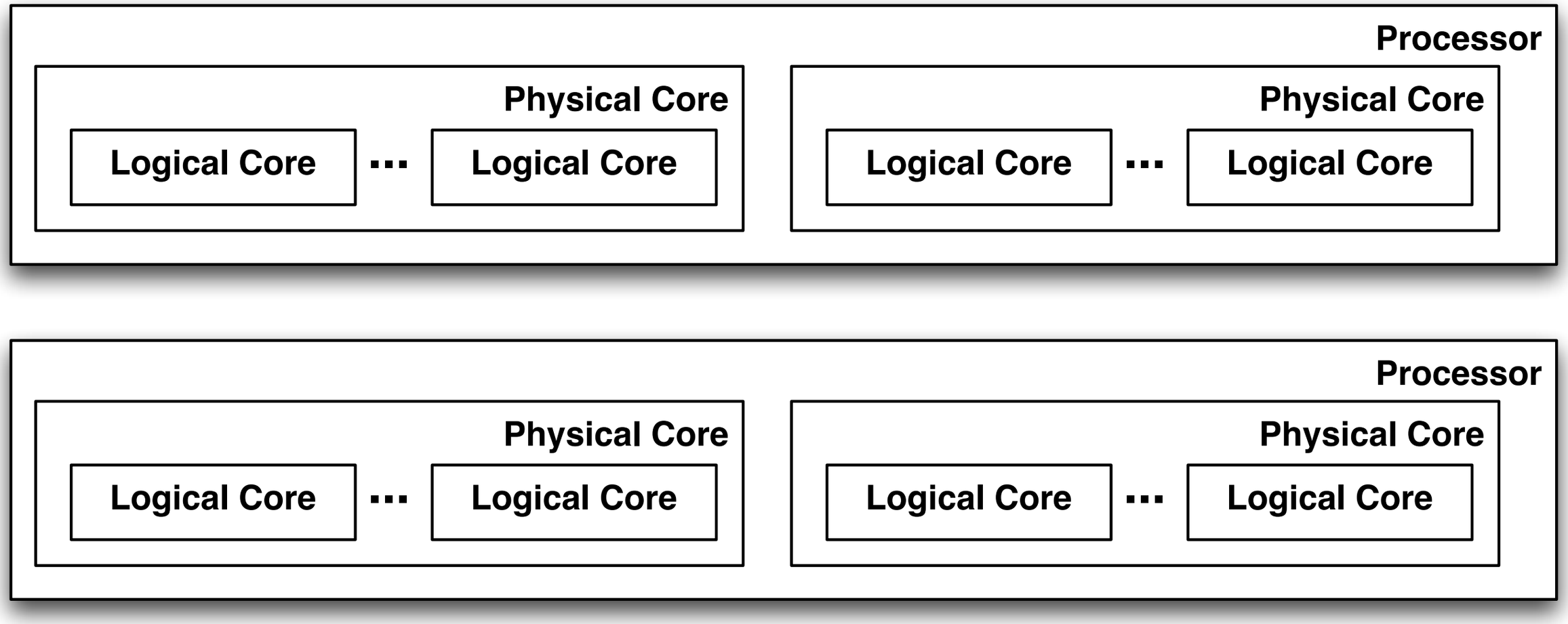
A 20% projected increase
in data center
emissions over next 5 years

Nearly ~50%
of power consumed
from IT equipment

The Center for Advanced Computer Studies

UNIVERSITY OF LOUISIANA Lafayette

Multicore Servers



The Center for Advanced Computer Studies

UNIVERSITY OF LOUISIANA Lafayette

OS Thread Scheduling

- Scheduling...
  - in time: who runs next
  - in space: who runs where
- Optimization problem
  - Who runs next: least use of energy with best performance quality of service
  - Who runs where: best utilization of resources with least increase in processor and/or ambient temperature

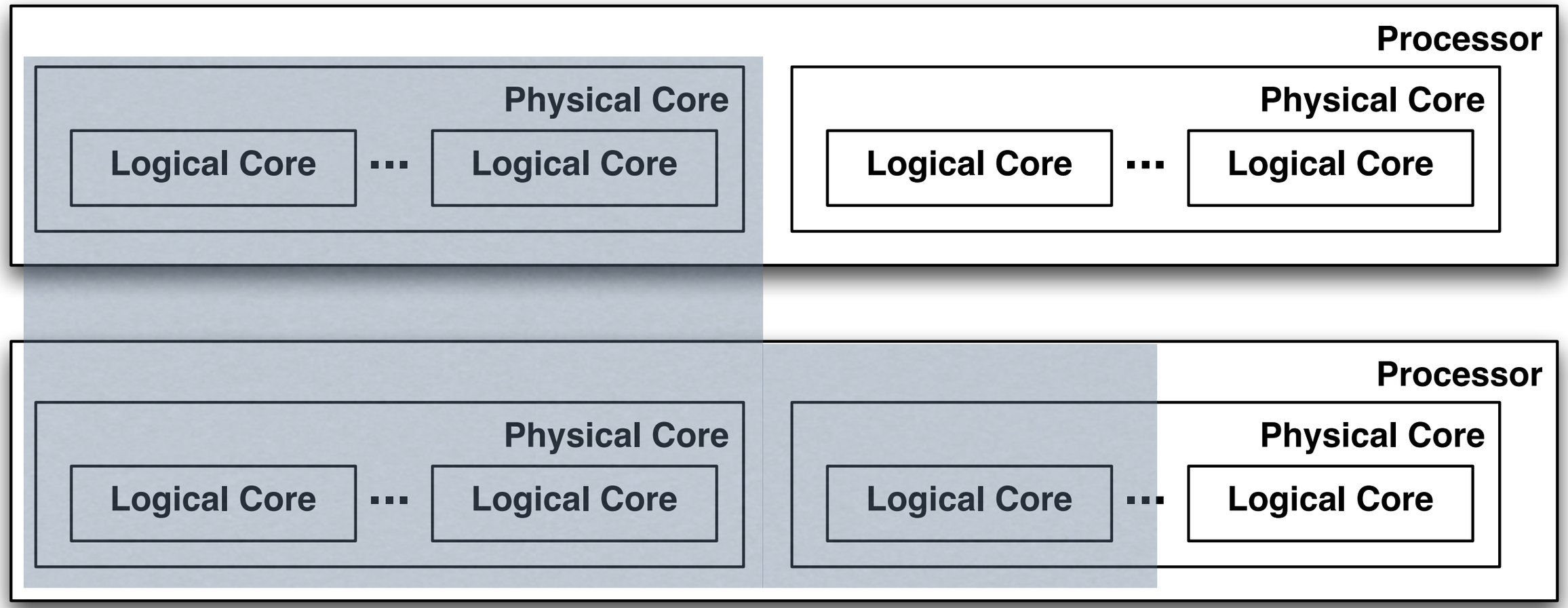
Current Practice

Completely Fair Scheduler
Domain-based Load Balancing
Power-state aware

Run-queue scheduling
Domain-based Load Balancing
Power-state aware (Solaris 11)

Run-queue scheduling
Interface w/ power manager?

Current Practice: Multicore Servers



Find the set of processors that maximizes performance on the smallest possible number of logical cores

The Center for Advanced Computer Studies

UNIVERSITY OF LOUISIANA Lafayette

Thread Scheduling & Power Management

DVFS:

$$P = CV^2 f$$

SpeedStep

Multi-core/Many-core

- Cache affinity
- Load balancing
- Opportunity to turn off the lights?

- Performance issues [LLBL 2007]
  - Lack of slack
  - High load = No gain
- Reliability issues [Bircher 2008]
  - Under-clocking & MTBF
- Reactive rather than proactive

Proactively Avoid Thermal Emergencies

A Full-System Energy Model + Effective Prediction → Thermal Aware Scheduling

- Possible approaches
  - Heat-and-Run and related approaches

    [Gomaa2004] [Coskun2009] [Zhou2010]
  - Memory-resource focused approaches

    [Merkel2010]
  - Control-theoretic techniques

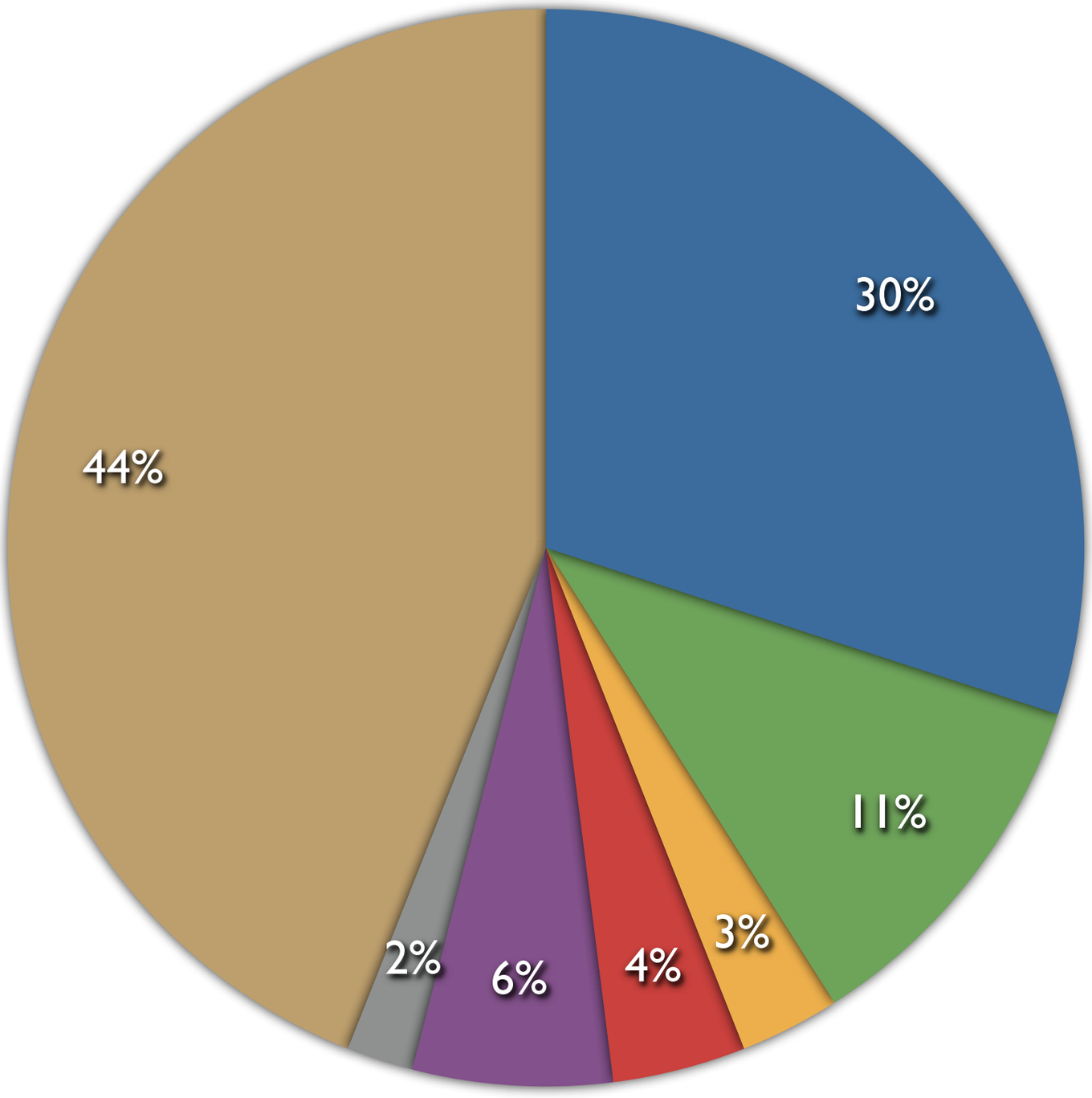
    [Ayoub2011]

Chaotic Attractor Predictors

The Center for Advanced Computer Studies

UNIVERSITY OF LOUISIANA Lafayette

Who is using all that power?



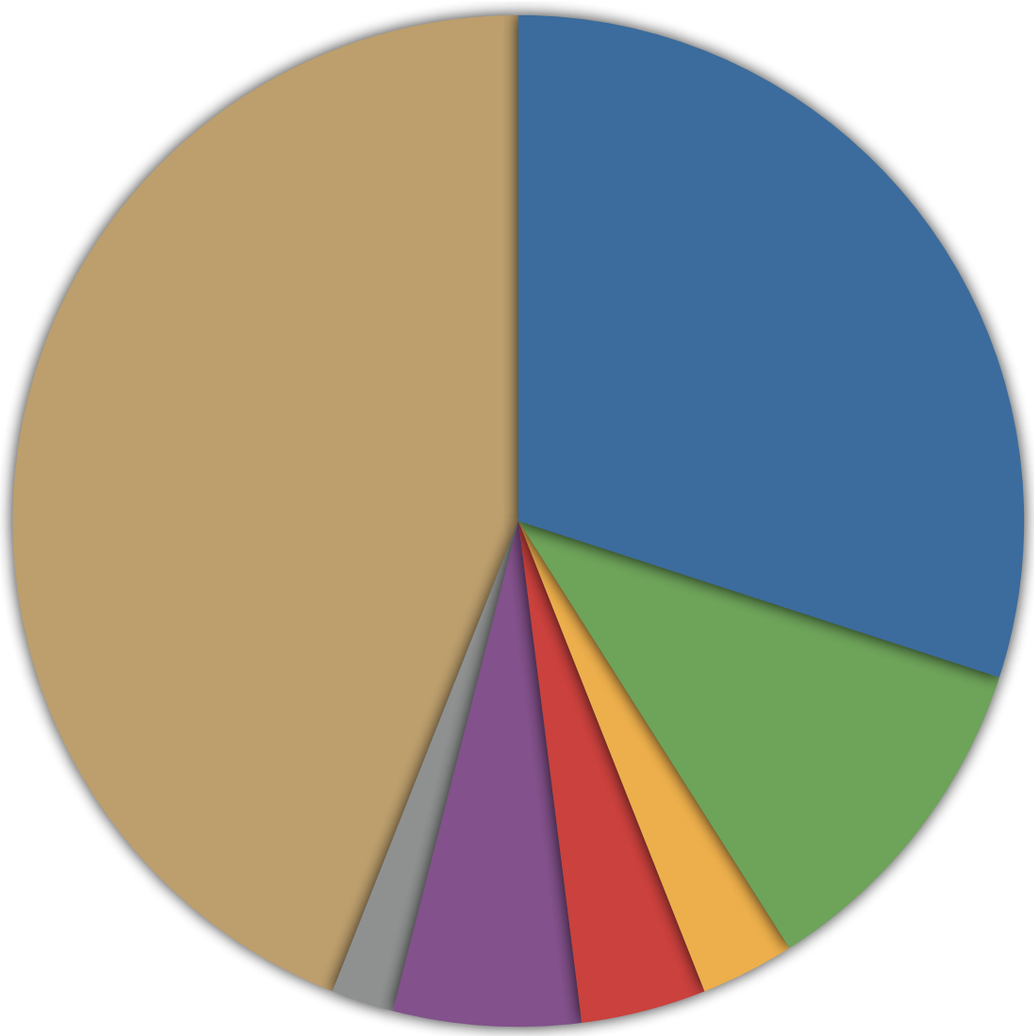
- Processor
- Memory
- PCI
- Planar
- HDD
- Standby
- Other*

✳ Other captures the power that runs the fans in the chassis & power supply inefficiency

Source: Intel Corp

A Full System Energy Model

$$E_{system} = E_{proc} + E_{mem} + E_{hdd} + E_{board} + E_{em}.$$



$$E_{proc} = \int_{t1}^{t2} (P_{proc}(t))dt$$

$$E_{mem} = \int_{t_1}^{t_2} \left((\sum_{i=1}^{N} CM_i(t) + DB(t)) \times P_{DR} + P_{ab} \right) dt$$

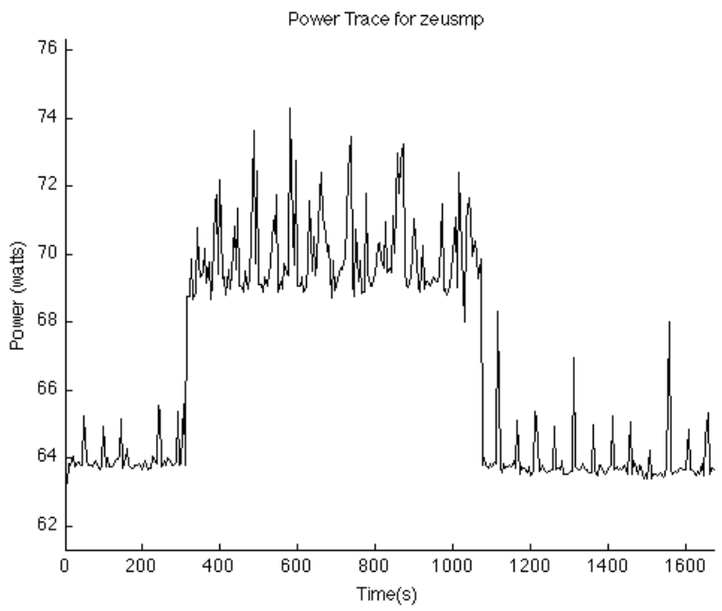
$$E_{hdd} = P_{spin-up} \times T_{su} + P_{read} \sum N_r \times T_r + P_{write} \sum N_w \times T_w + \sum P_{idle} \times T_{id}$$

$$E_{board} = \left(\sum V_{power-line} \times I_{power-line} \right) \times t_{interval}$$

$$E_{em} = \int_0^{T_p} \left(V(t) \cdot I(t) + \sum_{i=1}^{N} P_{fan}^i(t) \right) dt$$

[Lewis2008]

Linear AR Time Series - A good idea?



	AR		
Benchmark	**Avg Err %**	**Max Err %**	**RMSE**
astar	3.1%	8.9%	2.26
games	2.2%	9.3%	2.06
gobmk	1.7%	9.0%	2.30
zeusmp	2.8%	8.1%	2.14

Linear AR Model: AMD Opteron

- Linear Regression
  - Easy, simple
  - Odd mis-predictions
  - Corrective methods required

Benchmark	**Avg Err %**	**Max Err %**	**RMSE**
astar	5.9%	28.5%	4.94
games	5.6%	44.3%	5.54
gobmk	5.3%	27.8%	4.83
zeusmp	7.7%	31.8%	7.24

Linear AR Model: Intel Nehalem

Prediction w/ Chaotic Time Series

Chaotic behavior

Benchmark	Hurst Parameter (H)	Average Lyapunov Exponent
bzip2	(0.96, 0.93)	(0.28, 0.35)
cactusadm	(0.95, 0.97)	(0.01, 0.04)
gromac	(0.94, 0.95)	(0.02, 0.03)
leslie3d	(0.93, 0.94)	(0.05, 0.11)
omnetpp	(0.96, 0.97)	(0.05, 0.06)
perlbench	(0.98, 0.95)	(0.06, 0.04)

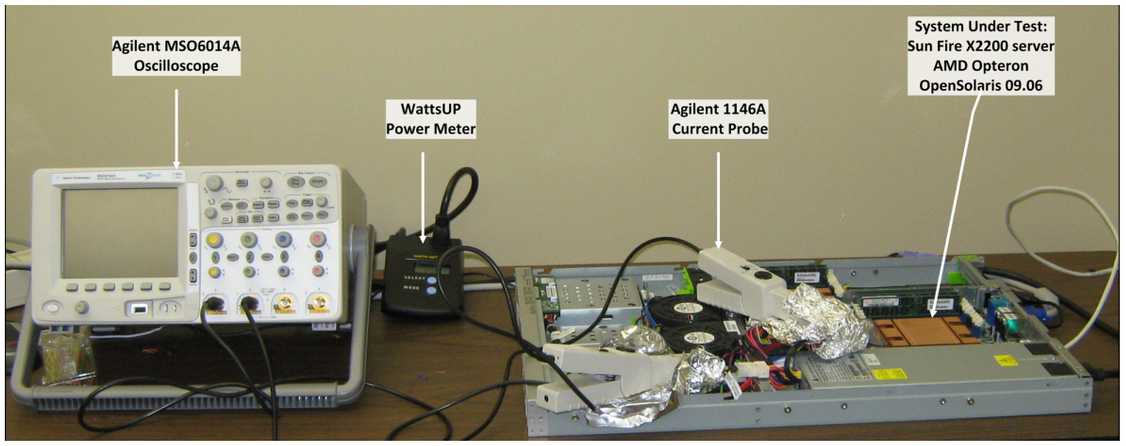
Chaotic behavior has been observed in measurement of DC-DC power converter output [Hamill1997] [Tse2002]

Chaotic Time Series

- Time-delay reconstructed state space
  - Uses Takens Embedding Theorem:
    - Time-delayed partition of observations to build function that preserves the topological and dynamical properties of our original chaotic system
- Find nearest neighbors on attractor to our observations
- Perform least-square curve fit to find a polynomial that approximates the attractor

[Lewis 2008] [Lewis 2010] [Lewis 2012]

Evaluation Environment



	Sun Fire 2200	**Dell PowerEdge R610**
CPU	2 AMD Opteron	2 Intel Xeon (Nehalem) 5500
CPU L2 cache	2x2MB	4MB
Memory	8GB	9GM
Internal disk	2060GB	500GM
Network	2x1000Mbps	1x1000Mbps
Video	On-board	NVIDA Quadro FX4600
Height	1 rack unit	1 rack unit

Training Benchmarks

Integer Benchmarks		
bzip2	C	Compression
mcf	C	Combinatorial Optimization
omnetpp	C++	Discrete Event Simulation
FP Benchmarks		
gromacs	C/F90	Biochemistry/Molecular Dynamics
cacstusADM	C/F90	Physics/General Relativity
leslie3d	F90	Fluid Dynamics
lbm	C	Fluid Dynamics

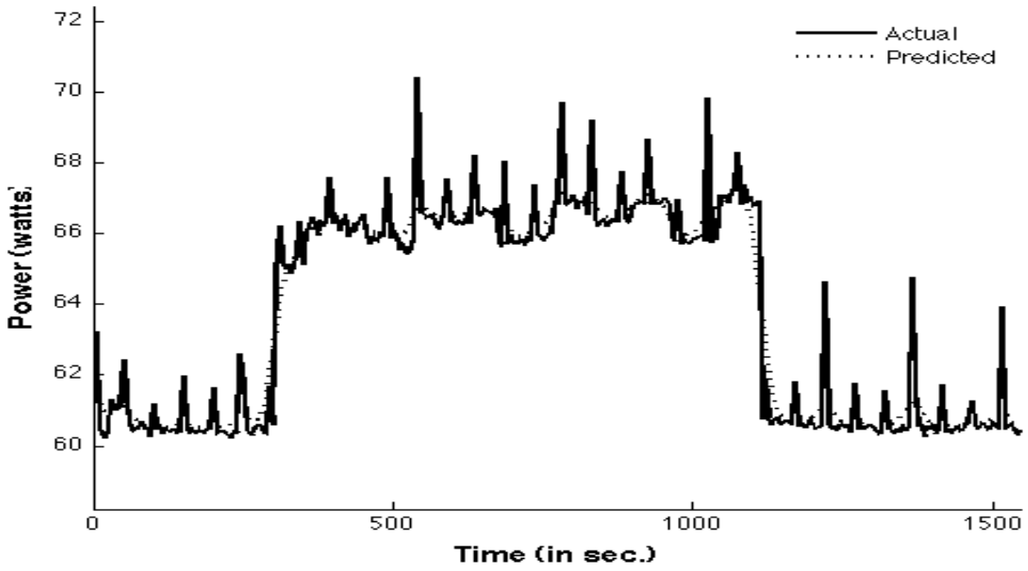
Evaluation Benchmarks

Integer Benchmark		
astar	C++	Path Finding
gobmk	C	Artificial Intelligence: Go
FP Benchmarks		
calculix	C++/F90	Structural Mechanics
zeusmp	F90	Computational Fluid Dynamics

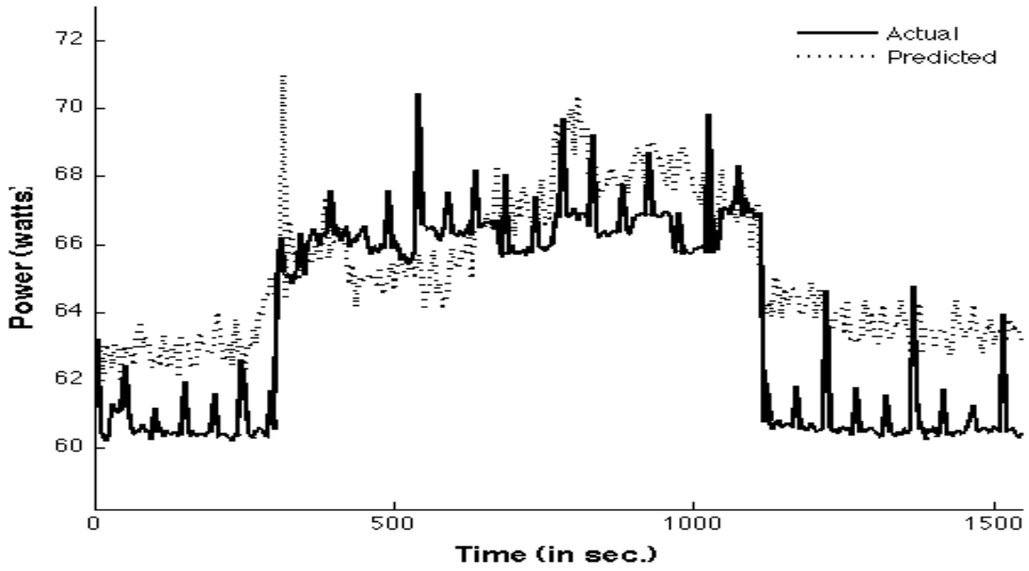
The Center for Advanced Computer Studies

UNIVERSITY OF LOUISIANA Lafayette

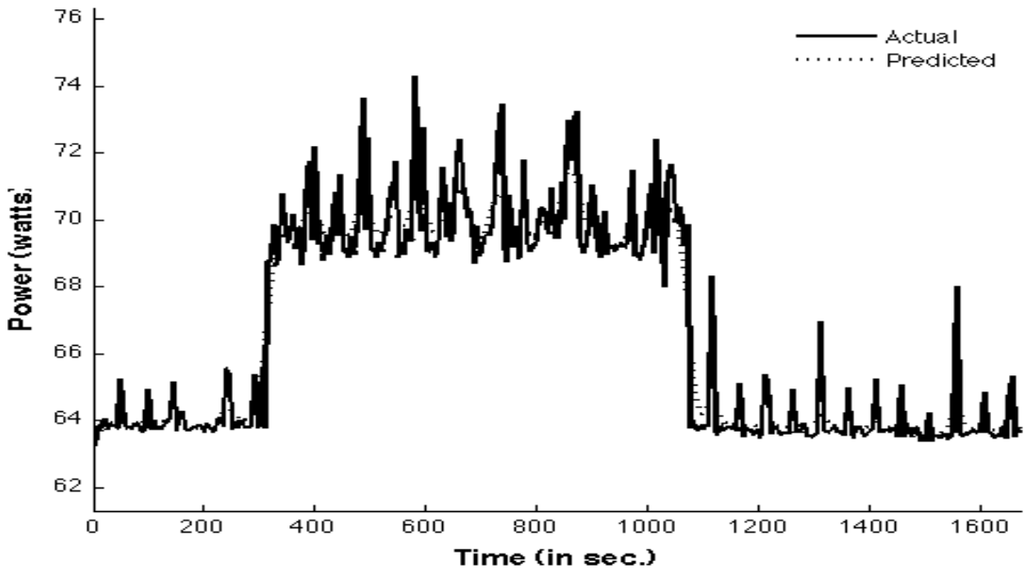
Results: AMD Opteron f10h



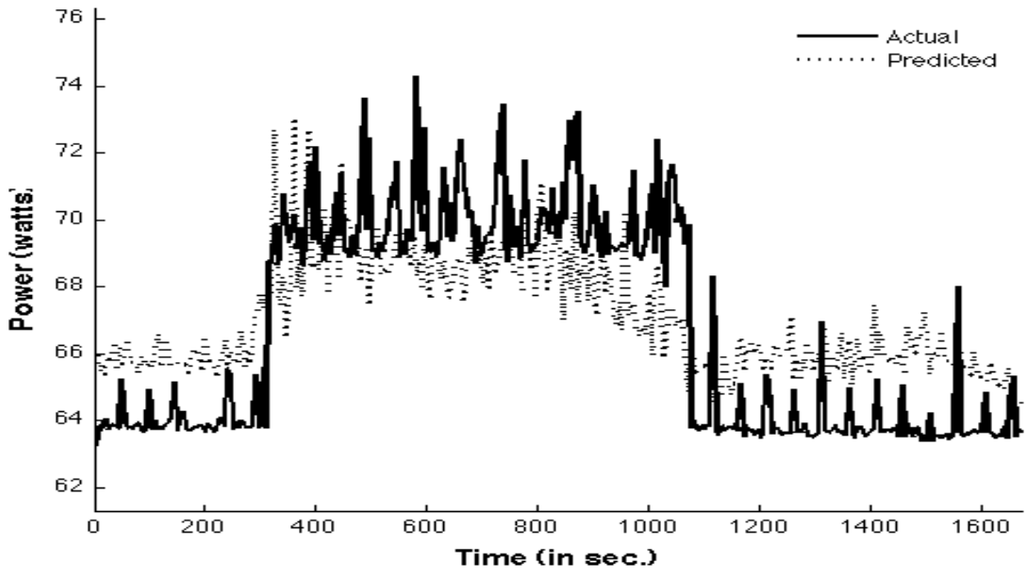
(a) Astar/CAP.



(b) Astar/AR(1)).



(c) Zeusmp/CAP.

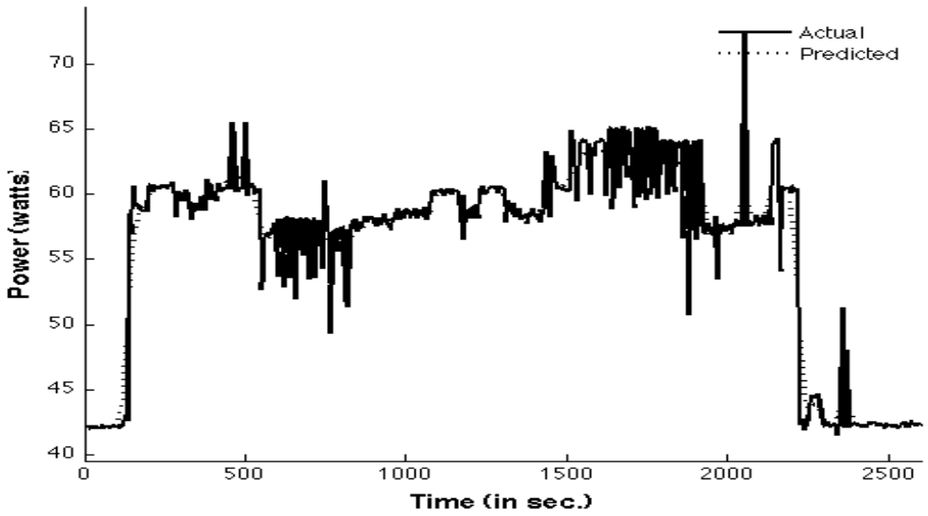


(d) Zeusmp/AR(1).

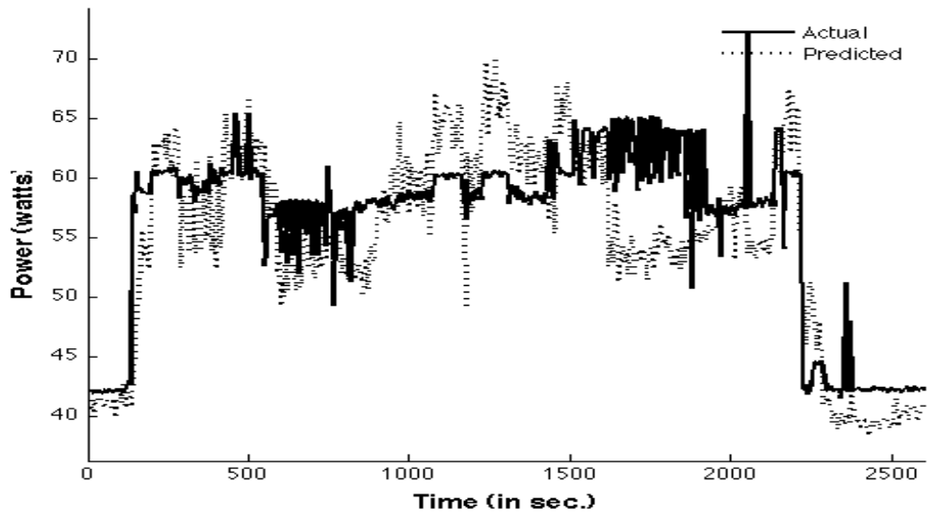
The Center for Advanced Computer Studies

UNIVERSITY OF LOUISIANA Lafayette

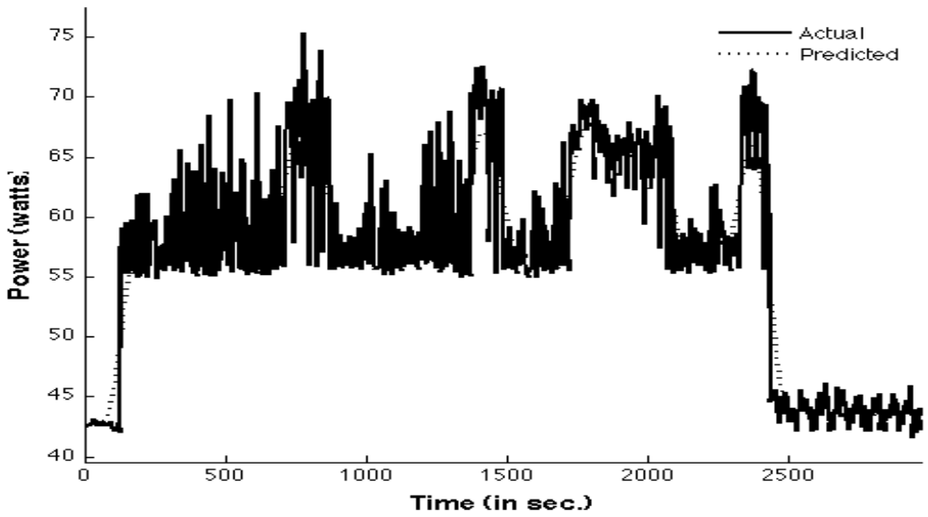
Results: Intel Nehalem



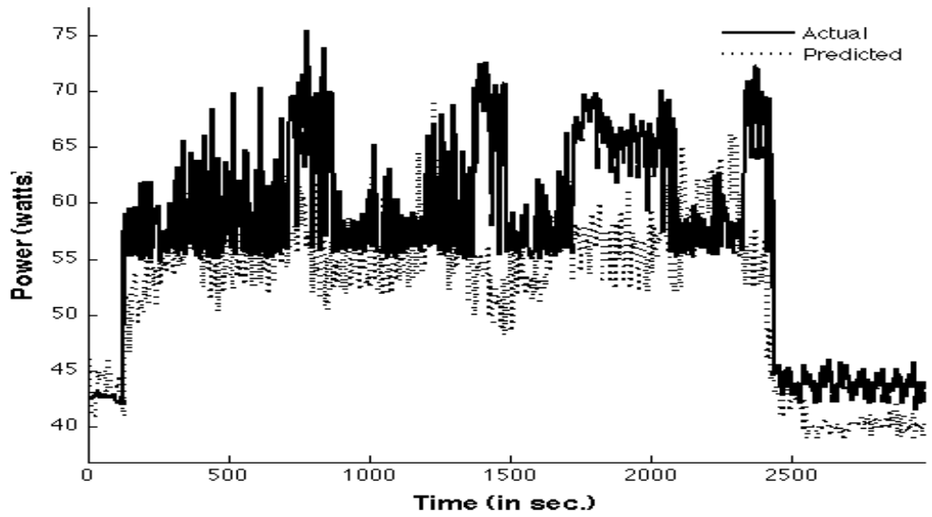
(a) Astar/CAP.



(b) Astar/AR(1)).



(c) Zeusmp/CAP.

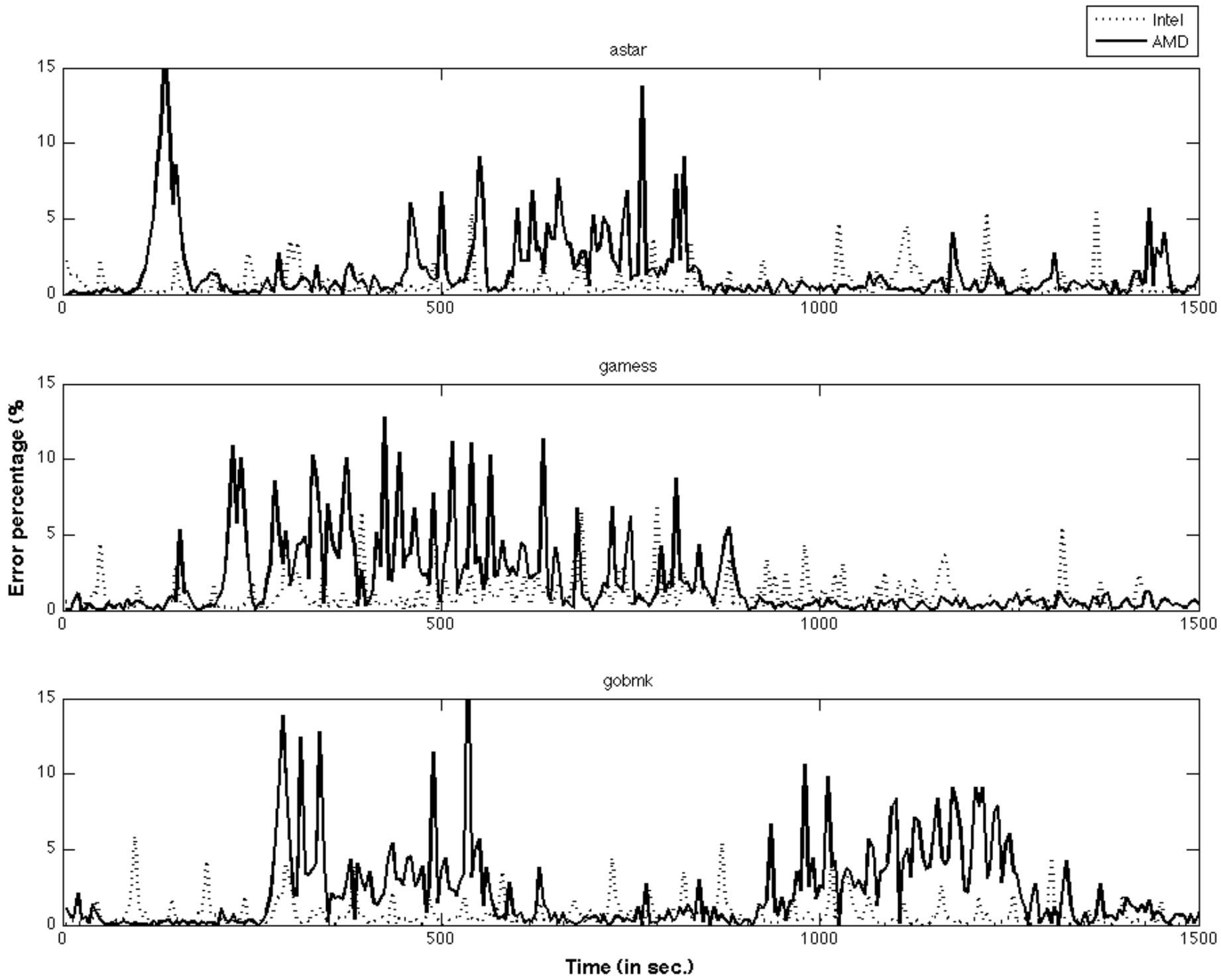


(d) Zeusmp/AR(1).

The Center for Advanced Computer Studies

UNIVERSITY OF LOUISIANA Lafayette

Results: Error - Other Benchmarks



The Center for Advanced Computer Studies

UNIVERSITY OF LOUISIANA Lafayette

Observations and Analysis

- Where does maximum error occur?
- Choice of performance counters
  - Difference in behavior between processors?
  - The right set of performance counters
- Benchmark selection

Thermal Aware Scheduling

The Center for Advanced Computer Studies

UNIVERSITY OF LOUISIANA Lafayette

Thermal Extensions to System Model

1. An application A is composed of p threads with data set D_A with d_i data per processor

2. The energy consumed by an application for a given workload W

$$E_A(A, D_A, t) = \sum_{i=1}^{p} W(\tau_i, d_i, t_i)$$

3. Which is used to compute the *Thermal Efficiency of Application*

$$\Theta(A, D_A, T, t) = \frac{E_A(A, D_A, t)}{\lim_{T \to T_{th}} J_e(D_A, \Psi_{cp})(T - T_{nominal})}$$

4. That is used to compute *Cost of Performance per Unit Power*

$$C_\theta(A, D_A, T, t) = \frac{\Theta(A, D_A, t)}{E_A(A, D_A, t)}$$

The Center for Advanced Computer Studies

UNIVERSITY OF LOUISIANA Lafayette

Extending CAP for Thermal Prediction

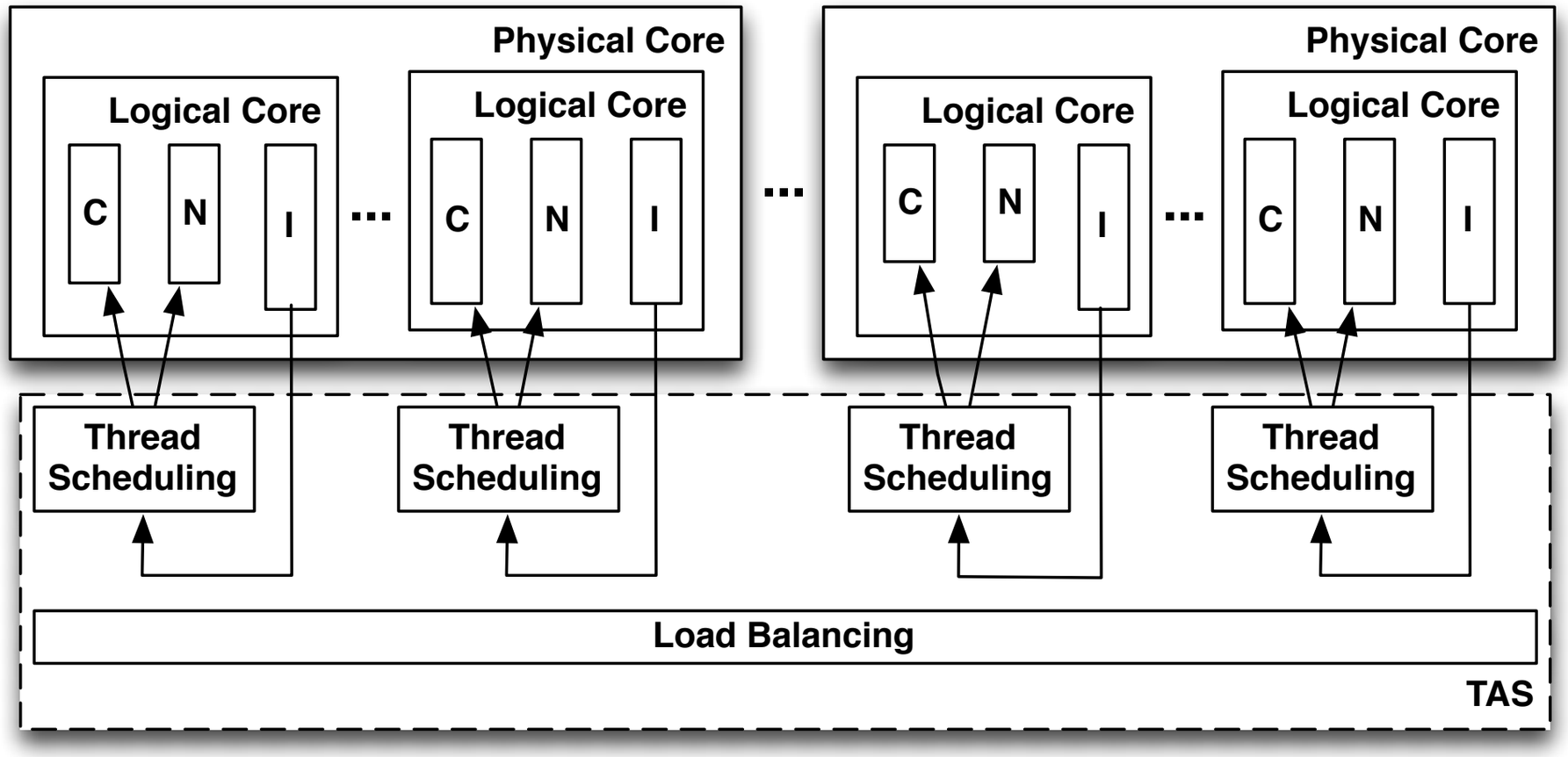
- Thermal Chaotic Attractor Predictor (tCAP)
  - Extends CAP to thermal domain
  - Created and used in similar manner to CAP
  - Matching tCAP for each thermal metric

Core Die Temp.	Hurst exp. (H)	Lyaponov exp. (λ)
Core 0	0.99	0.051
Core 1	0.98	0.019
Core 2	0.97	0.034
Core 3	0.95	0.040

Problem nature

- Scheduling...
  - in time: who runs next
  - in space: who runs where
- Optimization problem
  - Who runs next: least use of energy with best performance quality of service
  - Who runs where: best utilization of resources with least increase in processor and/or ambient temperature

A Thermal Aware Scheduler

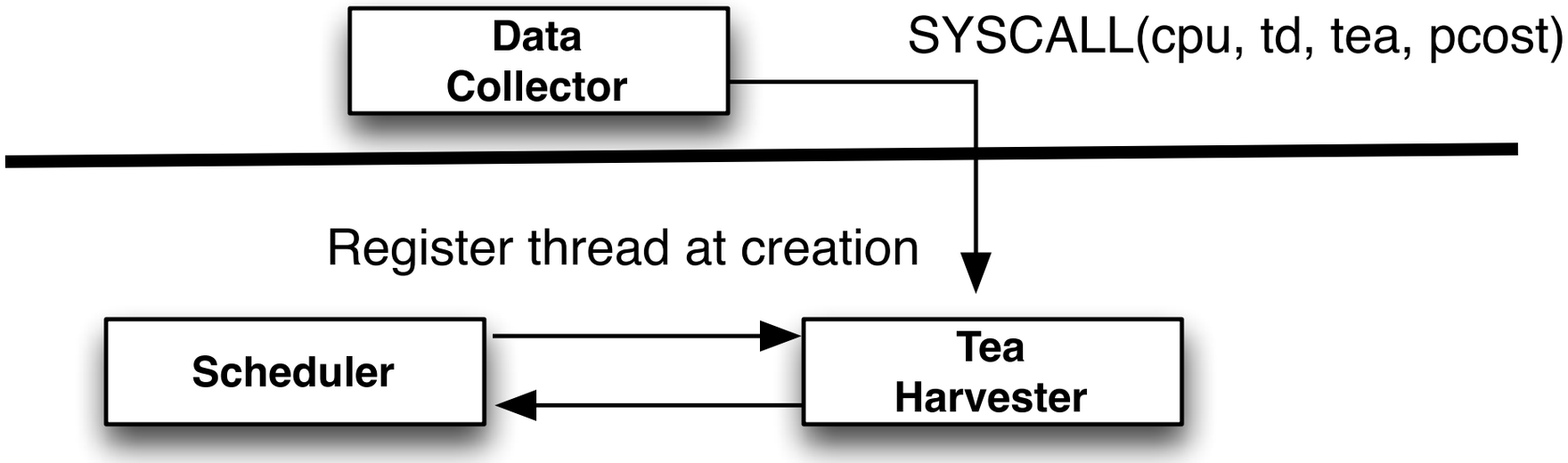


Extension of the ULE scheduler in FreeBSD

The Center for Advanced Computer Studies

UNIVERSITY OF LOUISIANA Lafayette

A Thermal Aware Scheduler



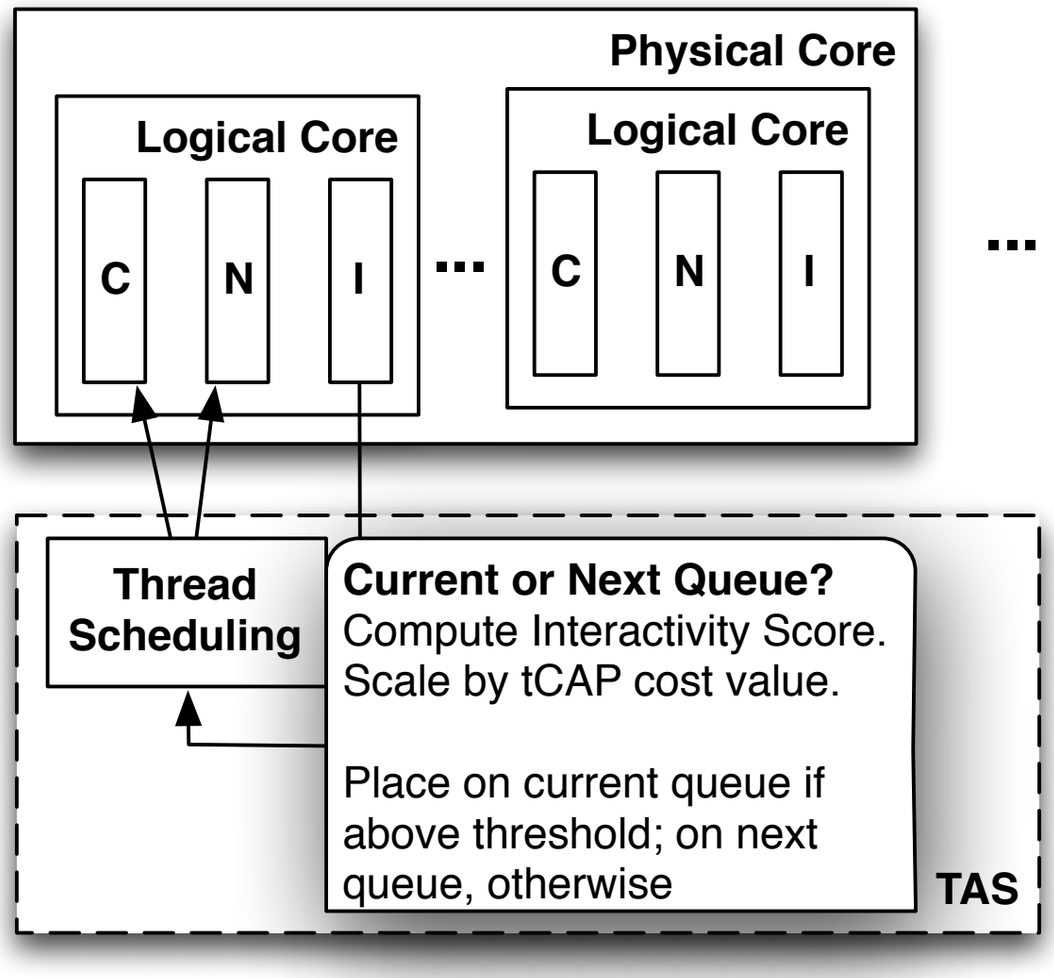
Hook function to get values

Separate the infrastructure for data collection and computing tCAP predictions

The Center for Advanced Computer Studies

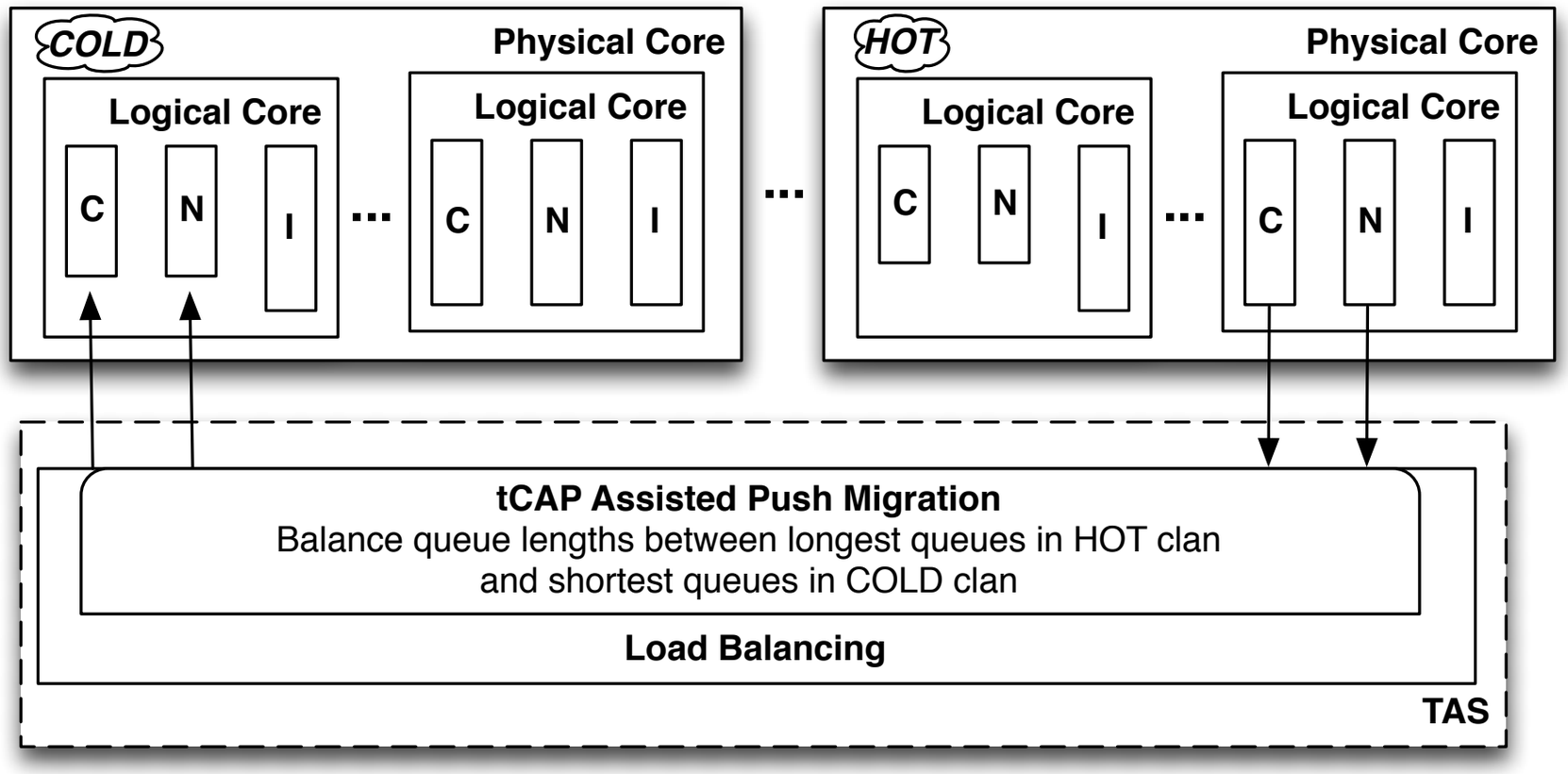
UNIVERSITY OF LOUISIANA Lafayette

Thread Scheduling



- Fairness: who runs when
  - Start with existing decisions about "interactivity"
- Penalize threads with higher "thermal cost"

Load Balancing



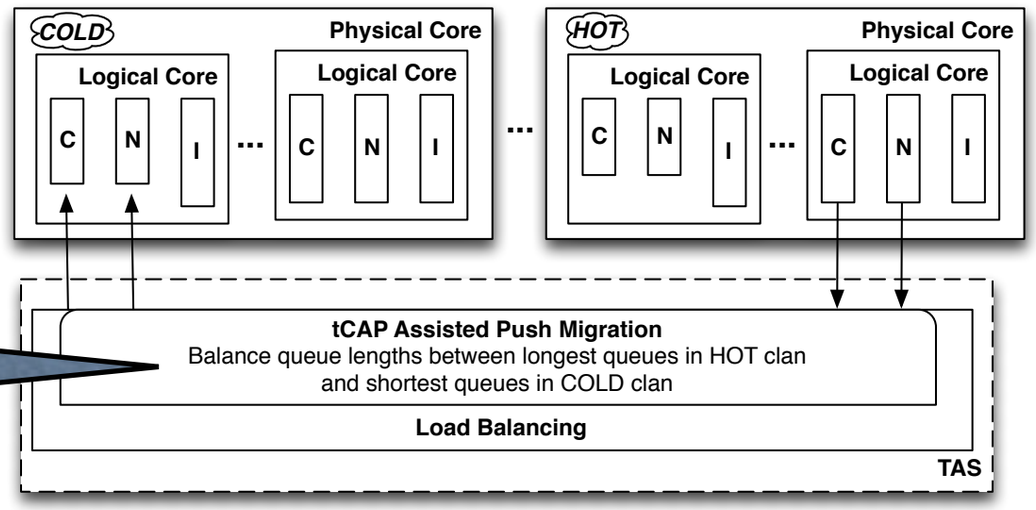
Accept less performance to provide time for overtaxed resources to recover

The Center for Advanced Computer Studies

UNIVERSITY OF LOUISIANA Lafayette

Load Balancing

BEGIN
  Determine if the logical core
    is in the Hot, Warm, or Cold clan.
  For each thread in the run queue
    BEGIN
      Use tCAP to estimate resulting
        temperature change if the thread
        is executed.
    END
  Determine least loaded core in the
      Cold clan.
  Migrate the thread with worst impact
      on temperature to logical core in
      the Cold clan most suitable from
      the speed standpoint.
END



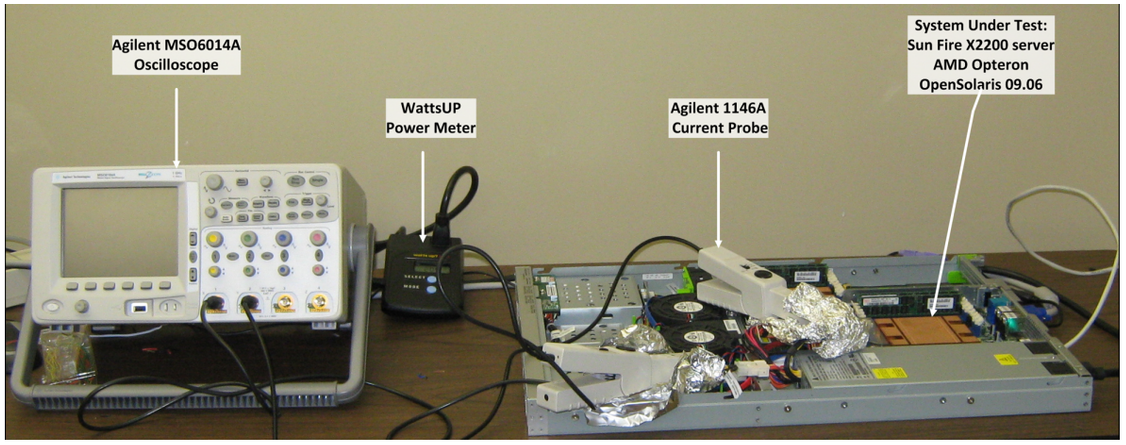
- Most suitable from a speed standpoint?
  - Shared last level
  - Same processor
  - Other processor

Thermal Aware Scheduler: Evaluation and Results

The Center for Advanced Computer Studies

UNIVERSITY OF LOUISIANA Lafayette

Evaluation Environment



	Dell Precision 490
CPU	Intel Xeon 5300 (Woodcrest)
CPU L2 cache	4MB
Memory	8GB, DDR2 667Mhz with ECC
Internal disk	500GB
Network	1x1000Mbps
Video	NVIDA Quadro FX3400

Training Benchmarks

Two extremes

- Idle
- Stressed

Evaluation Benchmarks

spec
CPU2006

PARSEC
['pär-"sek] A unit of measure

The Center for Advanced Computer Studies

UNIVERSITY OF LOUISIANA Lafayette

CPU-bound Benchmarks

SPEC CPU2006 benchmarks
Each benchmark is single-threaded
Combine into synthetic workloads
Multiple selection criteria

Workload Mix	**Benchmarks**
A	`namd`, `namd`, `hmmer`, `mcf`
B	`milc`, `milc`, `namd`,`hmmer`
C	`milc`, `mcf`,`mcf`, `hmmer`

Benchmark	**Inst Count (Billions)**	**Branches**	**Loads**	**Stores**
namd	2.483	4.28%	35.45%	8.83%
hmmer	3,363	7.08%	47.36%	17.68%
mcf	327	21.17%	37.99%	10.55%
milc	937	1.51%	40.15%	12.98%

CPU-bound Benchmarks

Workload Mix	Core die temperature reduction				Runtime increase
	Core 0	Core 1	Core 2	Core 3	
A	2.8°C	1.0°C	1.7°C	2.0°C	2.1%
B	1.0°C	0.8°C	2.0°C	1.1°C	2.9%
C	3.1°C	3.3°C	3.0°C	2.9°C	2.5%

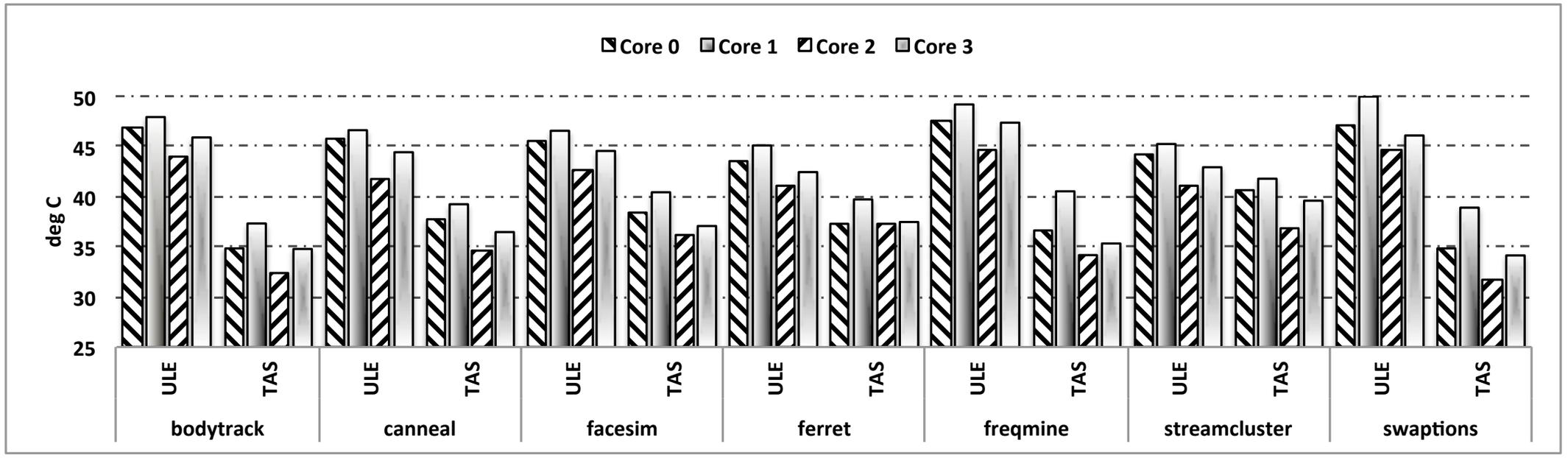
- Threads tend to pin themselves to particular cores
- In-line with prior work but
  - More aware of cache affinity
  - Other methods simulation based

Parallel Benchmarks

bodytrack	Computer vision image tracking application
canneal	Simulated annealing chip routing cost computation
facesim	Physical simulation of facial behavior
ferret	Content-based similarity search
freqmine	Simulate FP-growth method for Frequent Itemset Mining
streamcluster	Online clustering algorithm for data mining
swaptions	Simulated pricing of portfolio options

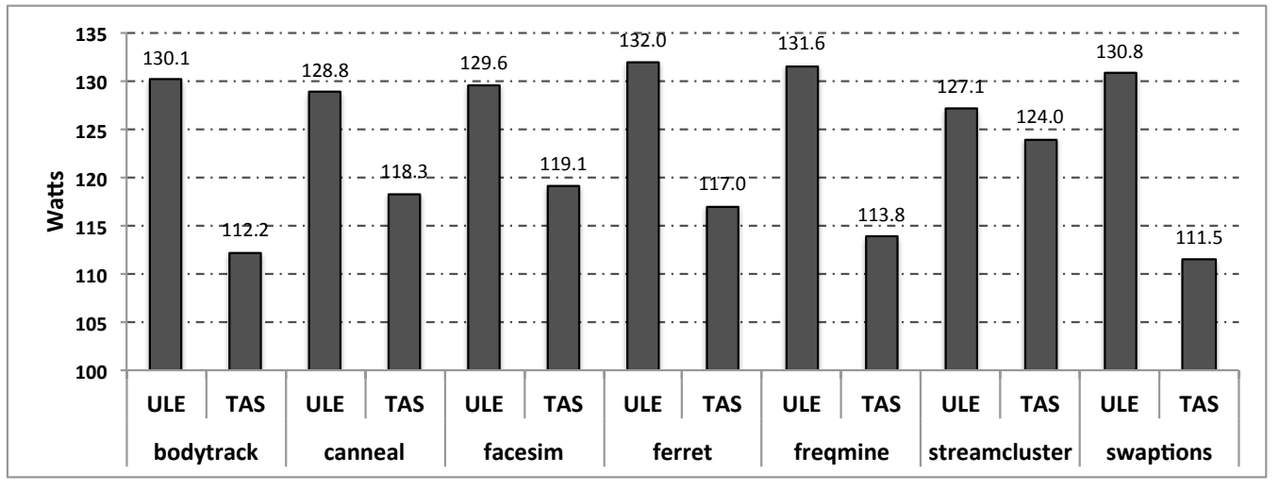
- Better represents server applications than CPU2006
- Natively parallel, more threads for scheduling, more opportunities for TAS

Temperature

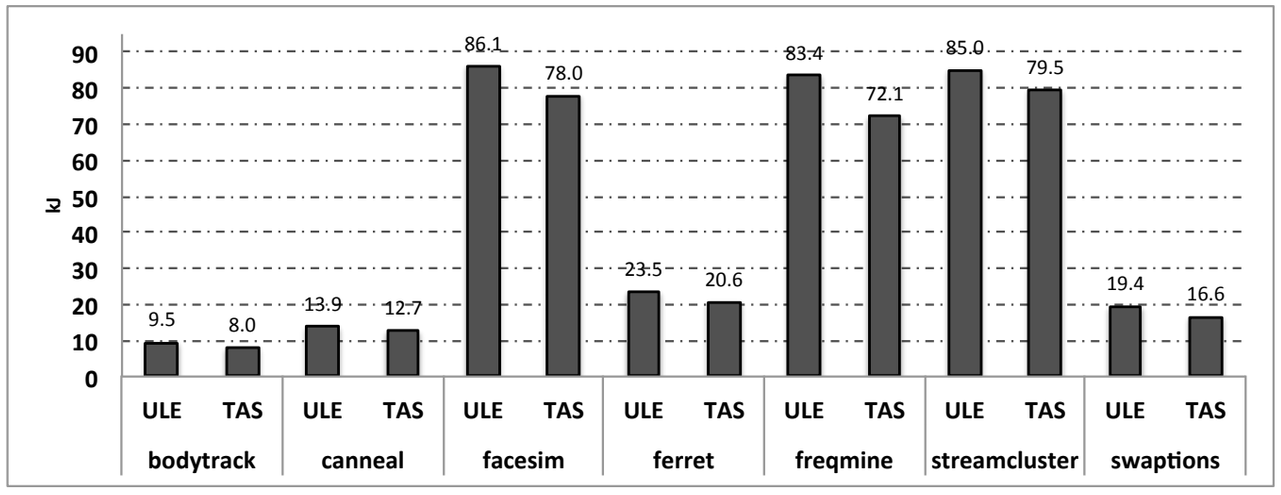


- Better results for benchmarks with smaller working sets
  - Lower cache requirements
  - Up to 12.8˚C improvement
- Benchmarks with streaming functions have much large working sets
  - Smaller temperature benefit
  - Average 3 to 6˚C improvement vs. ULE

Power and Energy



- Average power dissipation difference of ~3W to ~21W over all benchmarks

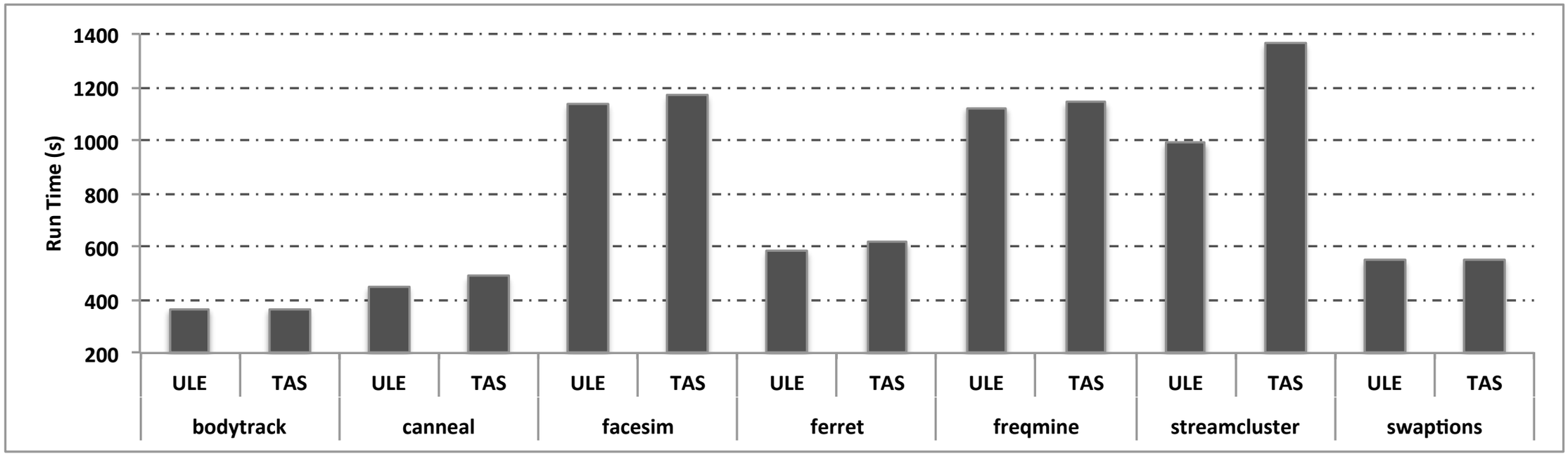


- Energy savings from 1.5kJ to 8.1kJ over all benchmarks

The Center for Advanced Computer Studies

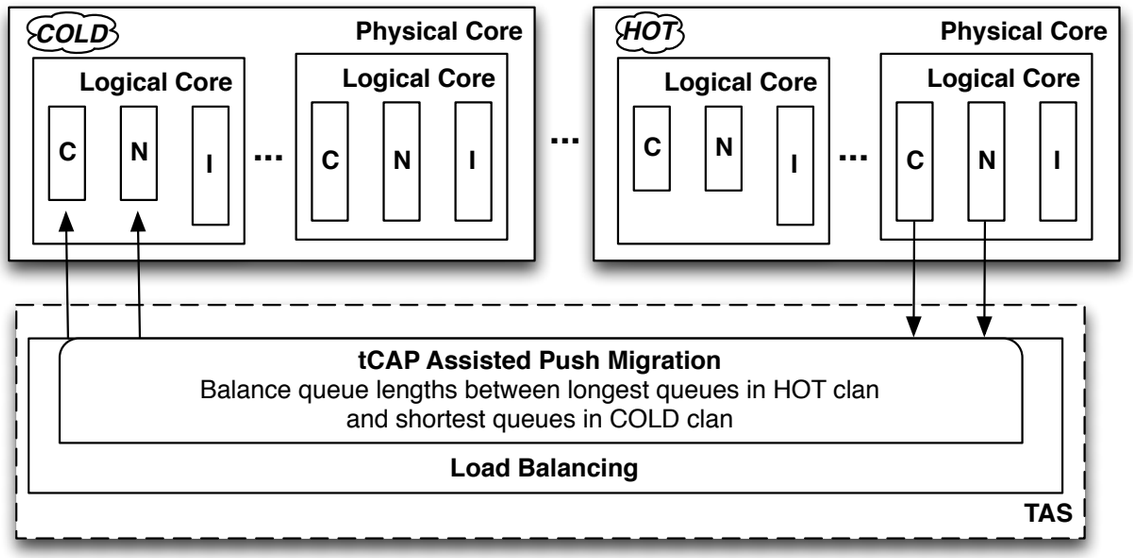
UNIVERSITY OF LOUISIANA Lafayette

Performance



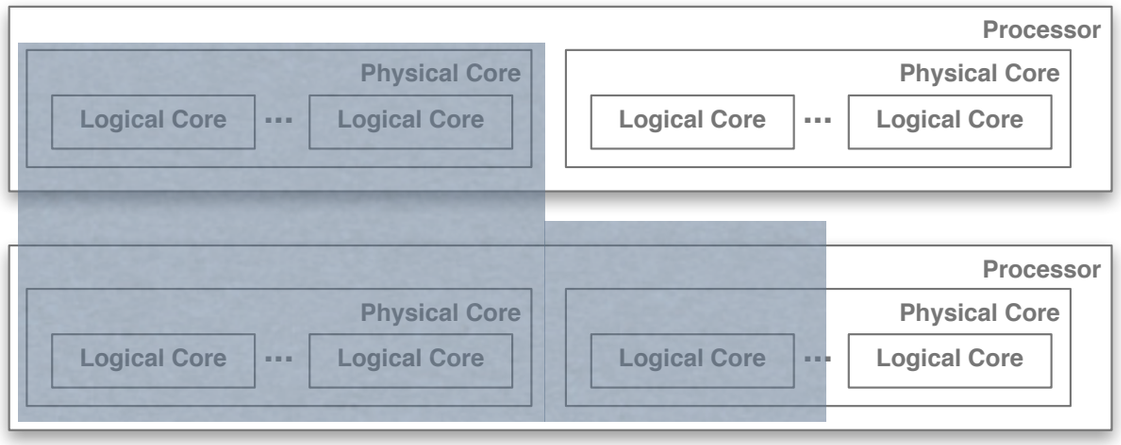
- Negligible performance degradation compared to ULE (no more than 3.3%)
- Behavior of streamcluster benchmark
  - Load balancing and cache affinity

Load Balancing and Cache Affinity



Where does the workload migrate?

How much is the load balancer aware of the processor's physical topology?



The Center for Advanced Computer Studies

UNIVERSITY OF LOUISIANA Lafayette

Summary

The Center for Advanced Computer Studies

UNIVERSITY OF LOUISIANA Lafayette

Current Status

A Full-System Energy Model + Effective Prediction → Thermal Aware Scheduling

- Development Complete
- Evaluation Complete
  - Intel + AMD processors
  - OpenSolaris (Solaris 11)
- Peer-reviewed
  - Journal: 1 (under review)
  - Conference/Workshop: 2

- Development complete
- Evaluation
  - Intel processor
  - FreeBSD
- Peer-reviewed
  - Journal: 1 (in-progress)
  - Conference/ Workshop: 2(1 under review)

The Center for Advanced Computer Studies

UNIVERSITY OF LOUISIANA Lafayette

Publications - Modeling and Prediction

Lewis, A., Tzeng, N.-F., and Ghosh, S. 2012. Run-Time Energy Consumption for Server Workloads based on Chaotic Time-Series Approximation. Under review for publication in ACM Transactions on Architecture and Code Optimization.

Lewis, A., Simon, J., and Tzeng, N.-F. 2010. Chaotic attractor prediction for server run-time energy consumption. Proc. of the 2010 Workshop on Power Aware Computing and Systems (Hotpower'10).

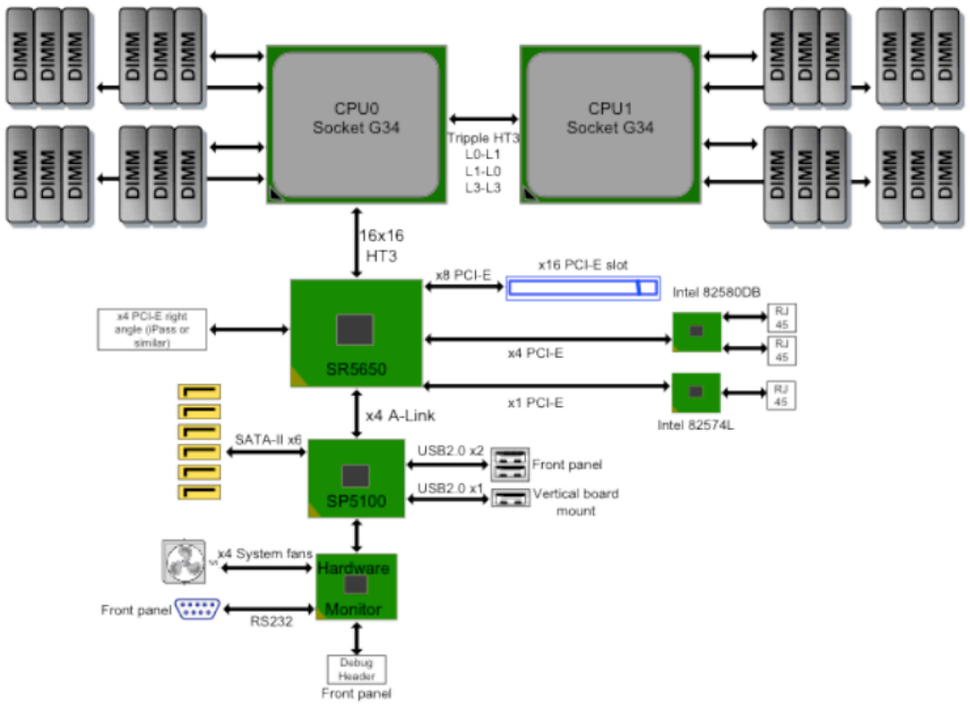
Lewis, A., Ghosh, S., and Tzeng, N.-F. 2008. Run-time energy consumption estimation based on workload in server systems. Proceedings of the 2008 conference on Power aware computing and systems.

Publications - Scheduling

Lewis, A., Ghosh, S., and Tzeng, N.-F. 2009. Thermal-Aware Scheduling for Real-Time Applications in Embedded Systems. Proceedings of the 2009 High Performance Embedded Computing Workshop.

Lewis, A. and Tzeng, N.-F. 2012. Thermal-Aware Scheduling in Multicore Systems Using Chaotic Attractor Predictors. Under review for inclusion in the Proc. of the 2012 Workshop on Energy Efficient Design.

Facebook & the Open Compute Project



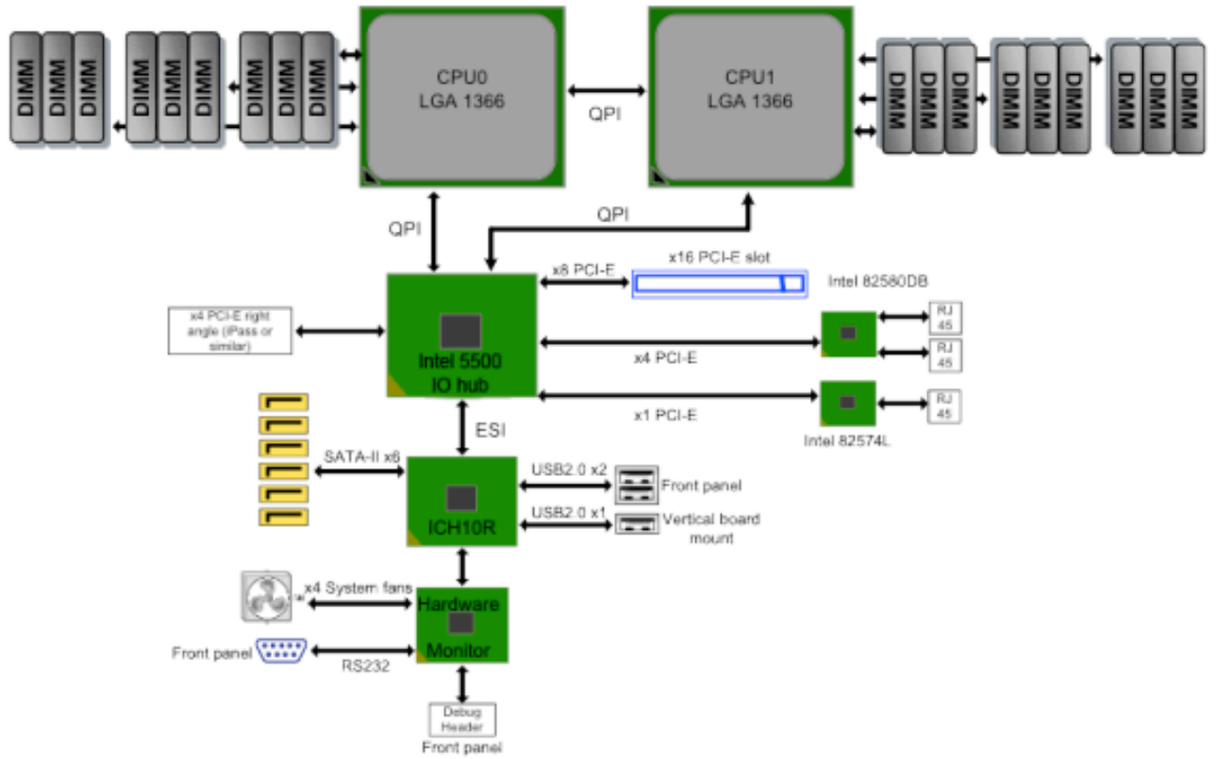
AMD Reference Design

A clean slate design as compared to state of the art is:

- 38% more efficient
- 24% less expensive

Source: Facebook

Intel Reference Design



Data center energy use (excluding small DCs, office IT equipment) equals

Electricity used by the entire U.S. transportation manufacturing industry (manufacture of automobiles, aircraft, trucks, and ships

2MW data center

=

4,600 typical U.S. cars