Energy Conservation and Thermal Management in High-Performance Server Architectures

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

The Center for Advanced Computer Studies

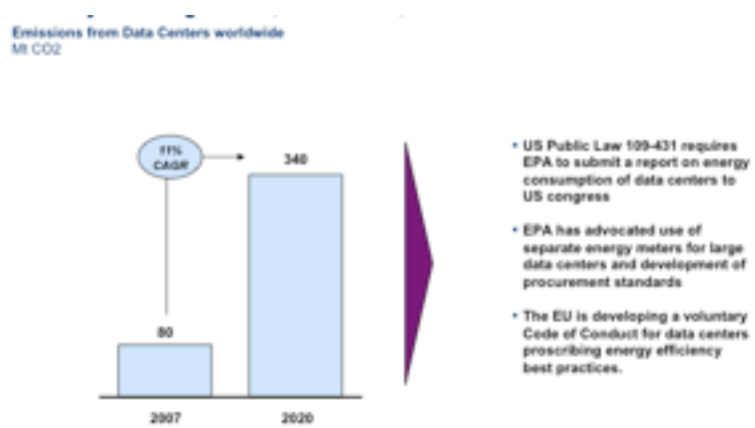
UNIVERSITY OF LOUISIANA Lafayette

Agenda

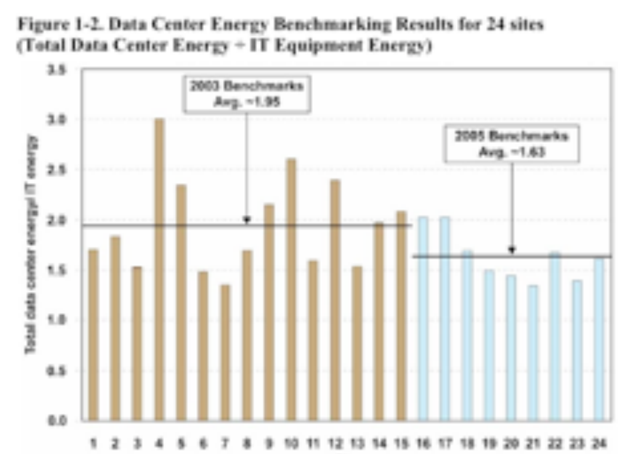
- Background and Related Work
- System Modeling
- Effective Prediction
- Initial Evaluation and Results
- Thermally-Aware Scheduling
- Status, Plans, and Summary

What does this picture tell us?

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



Source: McKinsey & Company 2008



Source: EPA 2008

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

Only ~50%
of power consumed
from IT equipment

The Center for Advanced Computer Studies

UNIVERSITY OF LOUISIANA Lafayette

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?

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

The Center for Advanced Computer Studies

UNIVERSITY OF LOUISIANA Lafayette

Proactively Avoid Thermal Emergencies

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

- Possible approaches
  - Heat-and-Run and related approaches
    [Gomaa2004] [Coskun2009] [Zhou2010]
  - Memory-resource focused approaches
    [Merkel2010]
  - Control-theoretic techniques

The Center for Advanced Computer Studies

UNIVERSITY OF LOUISIANA Lafayette

System Modeling

Model: Inputs & Components

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

$$E_{dc} = E_{system}$$

- Three DC voltage domains
  - 12Vdc, 5.5Vdc, 3.3Vdc
- 5.5V and 3.3V domains limited to 20% of rated voltage

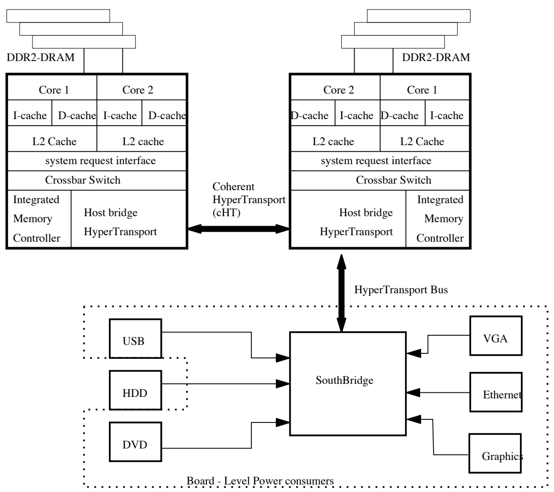
- Processor
- Memory
- Hard disk & storage devices
- Motherboard & peripherals
- Electrical & Electromechanical Components

The Center for Advanced Computer Studies

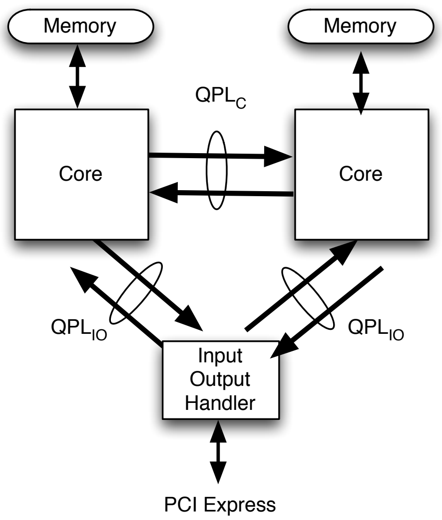
UNIVERSITY OF LOUISIANA Lafayette

Model: Processor

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



AMD Opteron



Intel Nehalem

- Processor as black box
  - Power = f(workload)
  - Manifests as heat

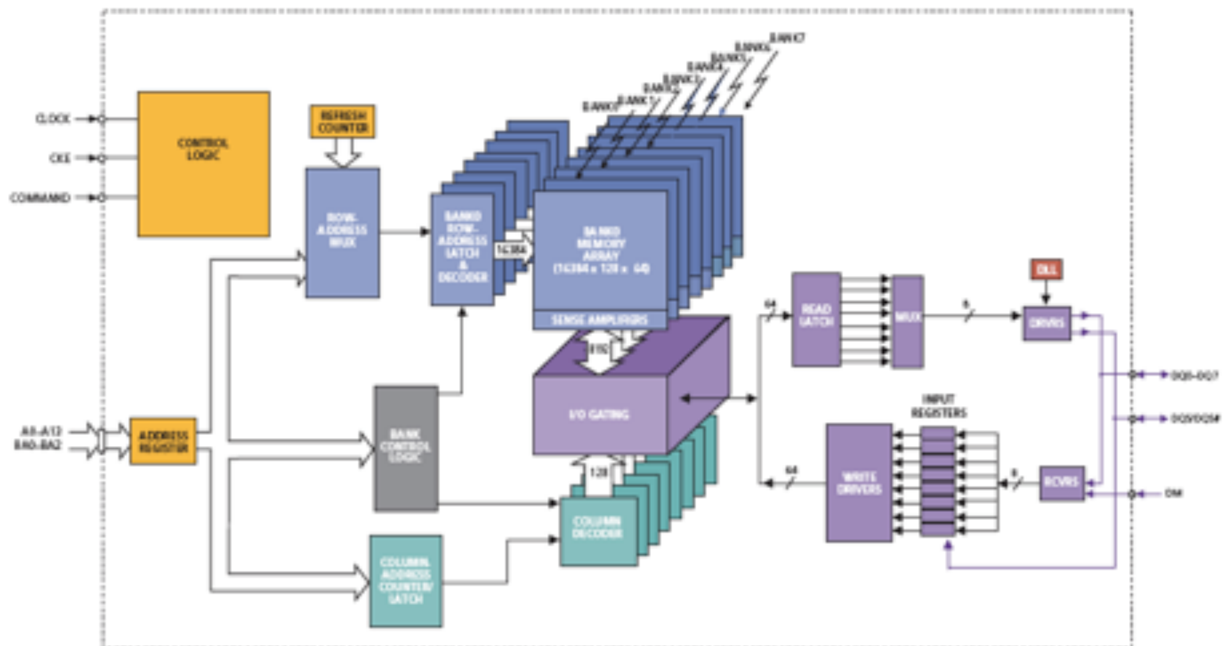
- Bus transactions
  - Reflects amount of data processed
- Die temperature
  - Computation per core
- Processor & system metrics

The Center for Advanced Computer Studies

UNIVERSITY OF LOUISIANA Lafayette

Model: Memory

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



- DRAM Read/Write power + background power = known quantities
- Performance counters exist for measuring the count of highest level cache miss and bus transactions
- Combine these to compute the energy consumed

The Center for Advanced Computer Studies

UNIVERSITY OF LOUISIANA Lafayette

Model: Storage

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

Parameter	Value
Interface	Serial ATA
Capacity	250 GB
Rotational speed	7200 rpm
Power (spin up)	5.25 W (max)
Power (Random read, write)	9.4 W (typical)
Power (Silent read, write)	7 W (typical)
Power (idle)	5 W (typical)
Power (low RPM idle)	2.3 W (typical for 4500 RPM)
Power (standby)	0.8 W (typical)
Power (sleep)	0.6 W (typical)

The Center for Advanced Computer Studies

UNIVERSITY OF LOUISIANA Lafayette

Model: Board

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

- System components that support the operation of the machine
  - Typically in the 5.5Vdc and 3.3Vdc power domains
  - Measured by current probe

The Center for Advanced Computer Studies

UNIVERSITY OF LOUISIANA Lafayette

Model: Electromechanical

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

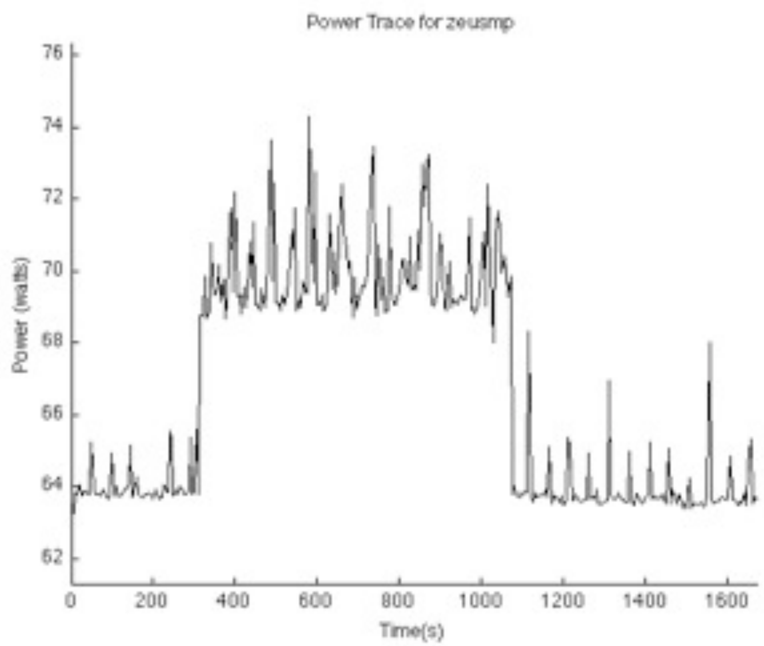
- Need to account for energy required to cool
  - No performance counters
  - Can measure power drawn by the fans
  - Derived from log data collected by OS

The Center for Advanced Computer Studies

UNIVERSITY OF LOUISIANA Lafayette

Effective Prediction

Linear AR Time Series - A good idea?



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

	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

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 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

Kernel weighting

1.

$$K(x) = (2\pi)^{-\frac{m}{2}} exp(-\|x\|^2/2)$$

$$K_\beta(x) = \frac{1}{\beta} K(\frac{x}{\beta})$$

2.

$$\beta = \left(\frac{4}{3p}\right)^{\frac{1}{5}} \sigma$$

$$\bar{\sigma} = median(|x_i - \bar{\mu}|)/0.6745$$

3.

$$\hat{f}(x) = \frac{\sum_{t=p+1}^{n+p} O_p * K_\beta(X_{t-1} - x)}{\sum_{t=p+1}^{n+p} K_\beta(X_{t-1} - x)}$$

$$O_p = (X_{t-1}, \ldots, X_{t-p})^T$$

The Center for Advanced Computer Studies

UNIVERSITY OF LOUISIANA Lafayette

Forward prediction

- Start with a Taylor series expansion

$$\hat{f}(X) = \hat{f}(x) + \hat{f}'(x)^T (X - x)$$

- Find the coefficients of the polynomial by solving the linear least squares problem for a and b:

$$\sum_{t=p+1}^{n+p} \left[X_t - a - b^T (X_{t-1} - x)\right]^2 * K_\beta(X_{t-1} - x)$$

- Explicit solution for our linear least squares problem:

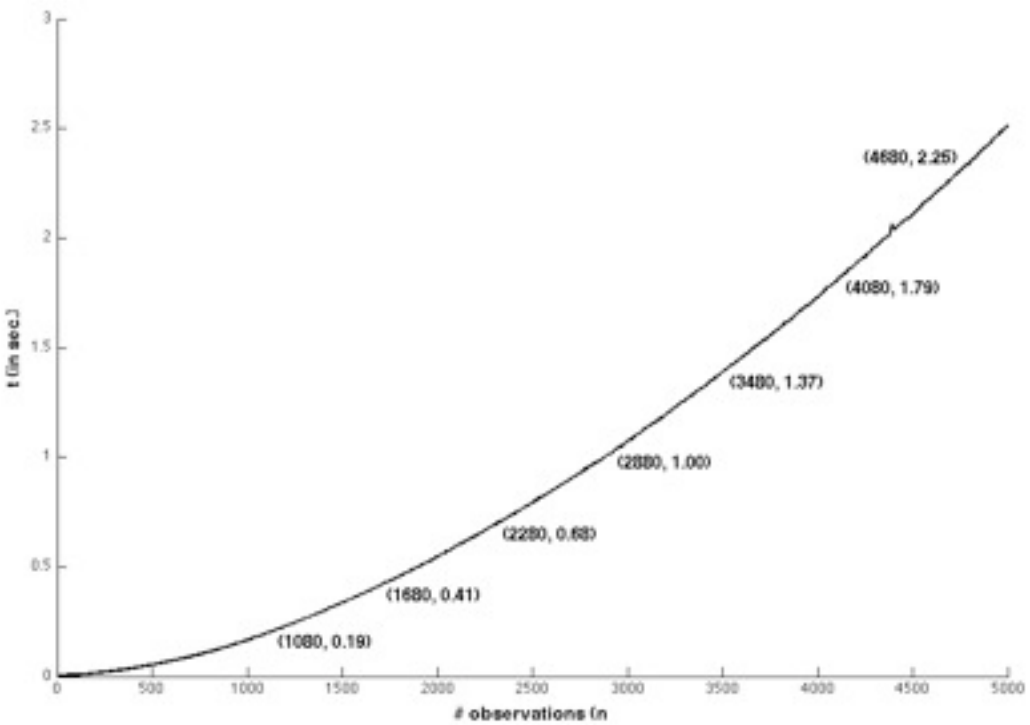
$$\hat{f}(x) = \frac{1}{n} \sum_{t=p+1}^{n+p} (s_2 - s_1 * (x - X_{t-1}))^2 * K_\beta((x - X_{t-1})/\beta)$$

$$s_i = \frac{1}{n} \sum_{t=p+1}^{n+p} (x - X_{t-1})^i * K_\beta((x - X_{t-1})/\beta)$$

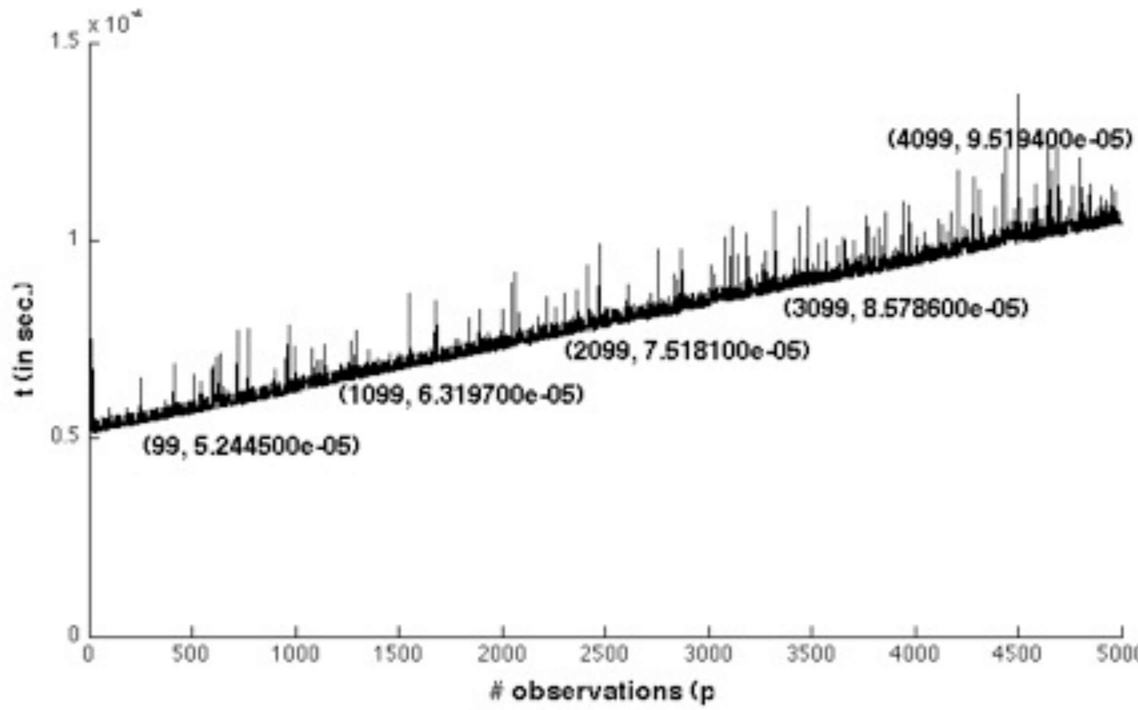
The Center for Advanced Computer Studies

UNIVERSITY OF LOUISIANA Lafayette

Time Complexity



n future observations



p past observations

Creating a CAP: $O(n^2)$

Predicting with a CAP: $O(p)$

The Center for Advanced Computer Studies

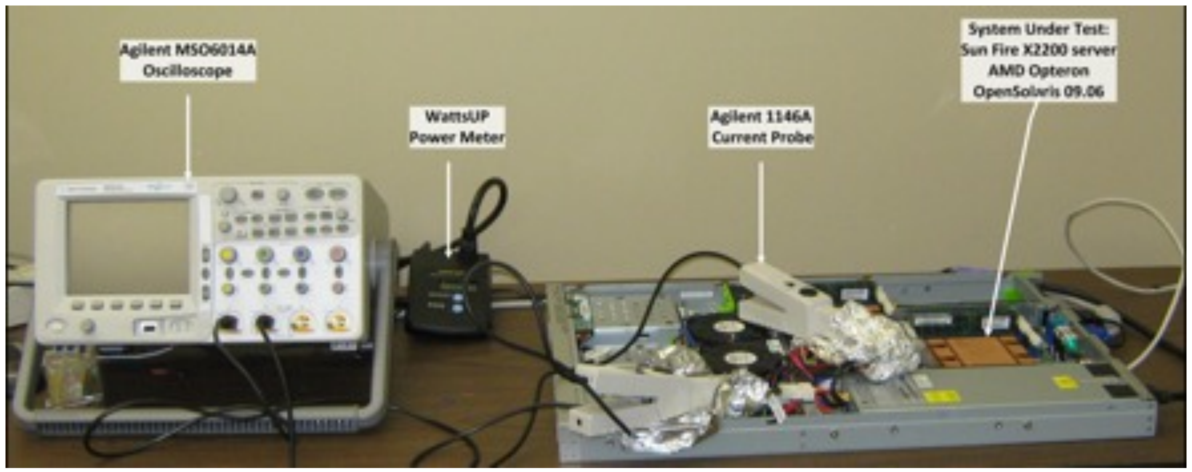
UNIVERSITY OF LOUISIANA Lafayette

Initial Evaluation and Results

The Center for Advanced Computer Studies

UNIVERSITY OF LOUISIANA Lafayette

Initial Evaluation and Results



	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

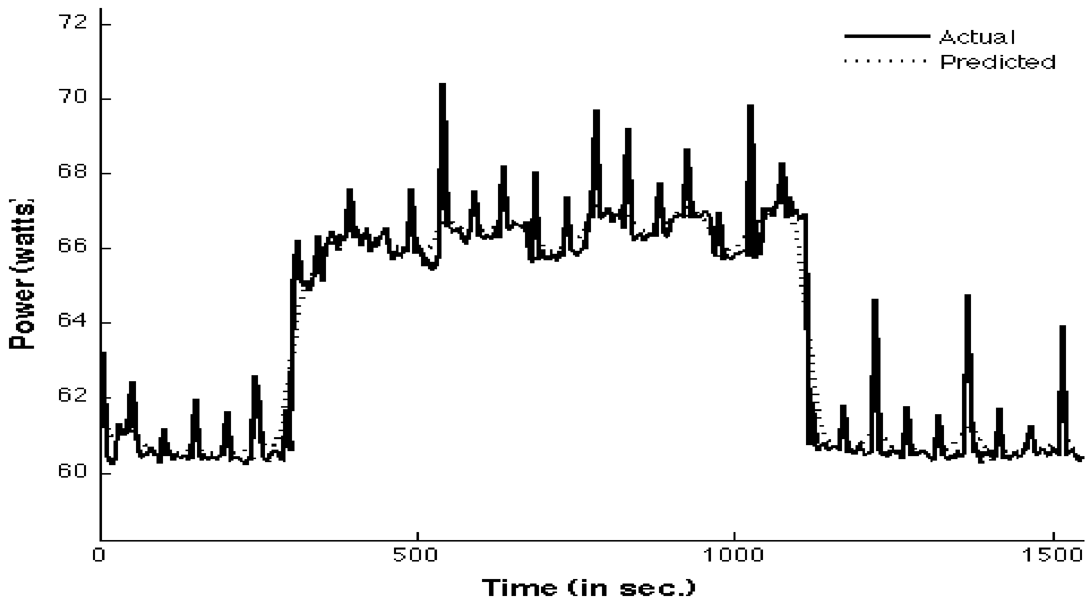
Evaluation Benchmarks

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

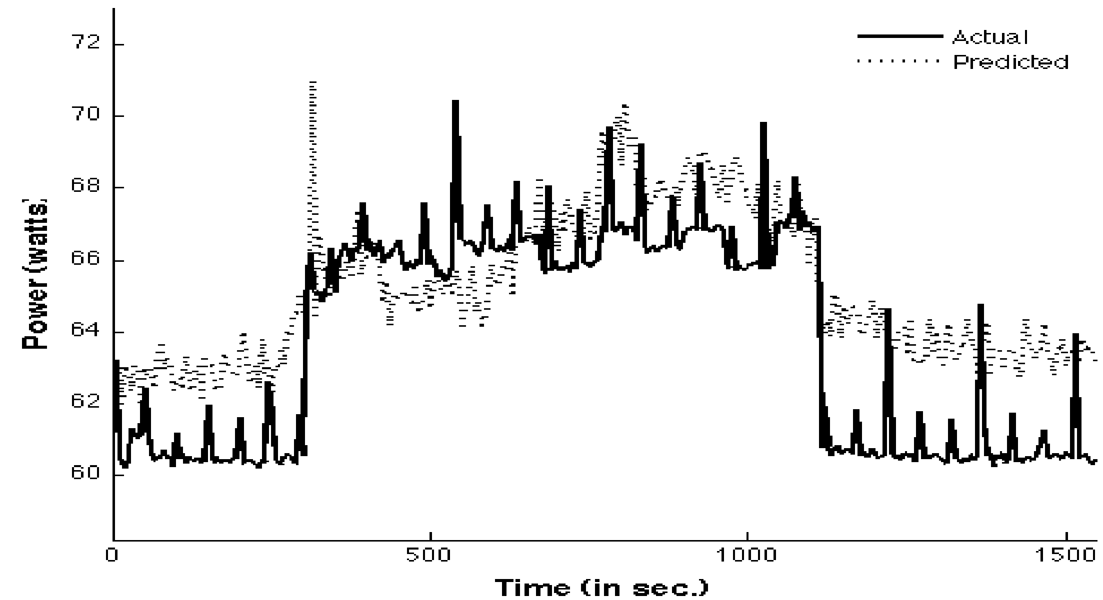
The Center for Advanced Computer Studies

UNIVERSITY OF LOUISIANA Lafayette

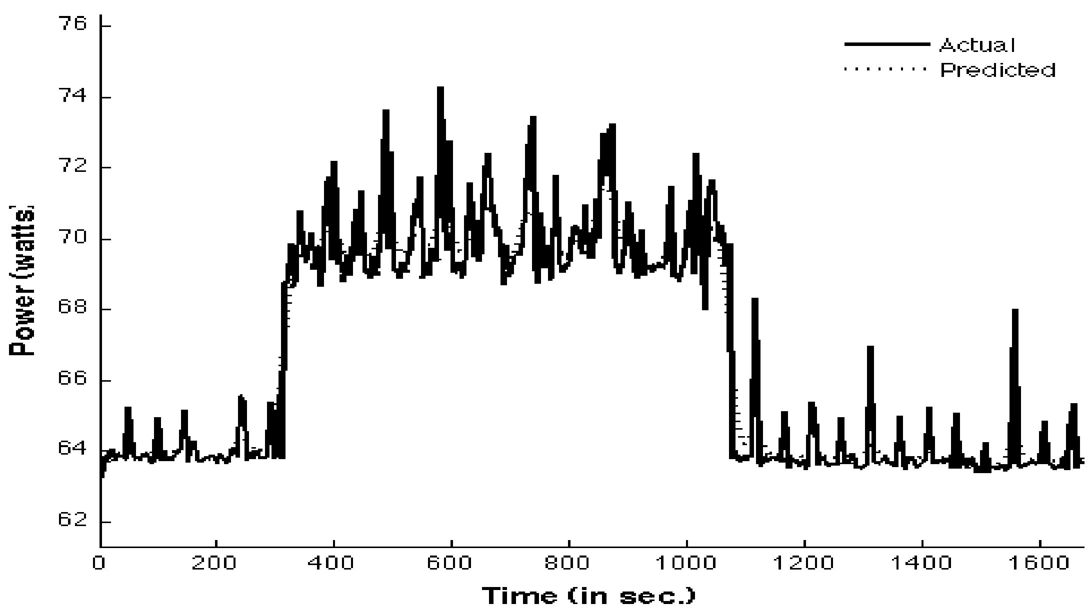
Results: AMD Opteron f10h



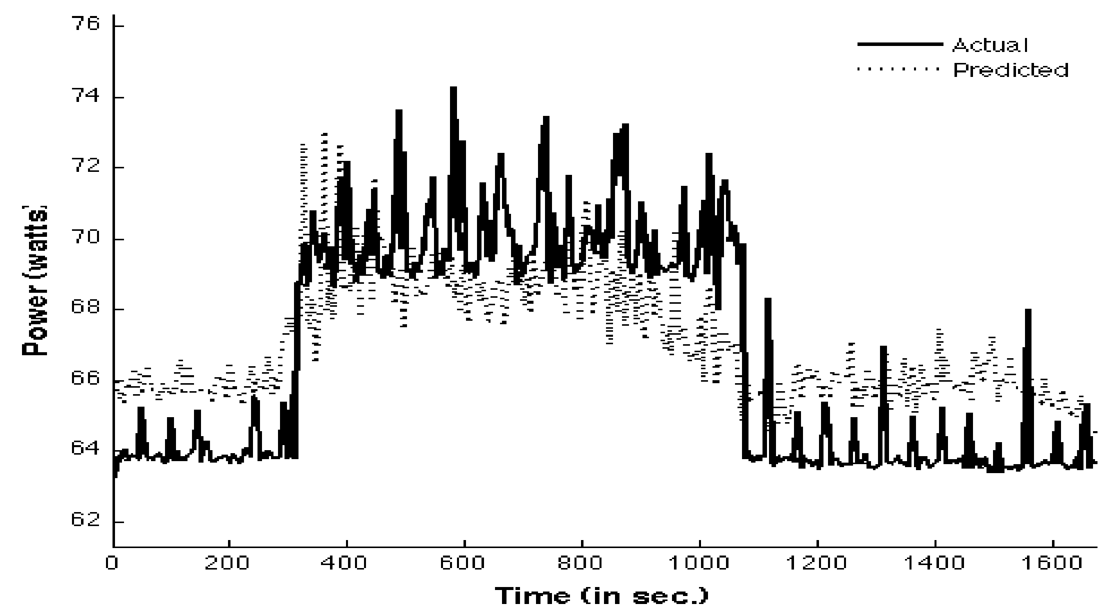
(a) Astar/CAP.



(b) Astar/AR(1)).



(c) Zeusmp/CAP.

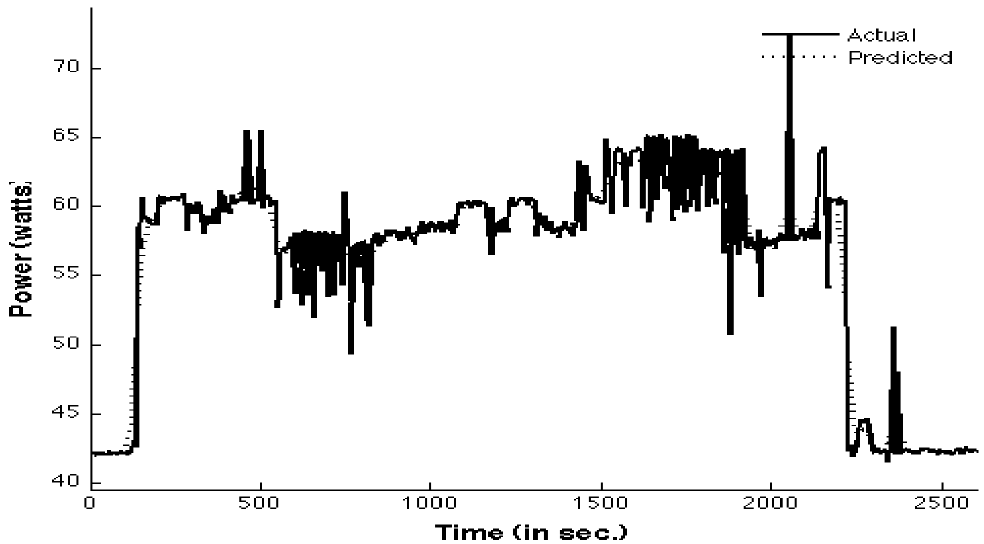


(d) Zeusmp/AR(1).

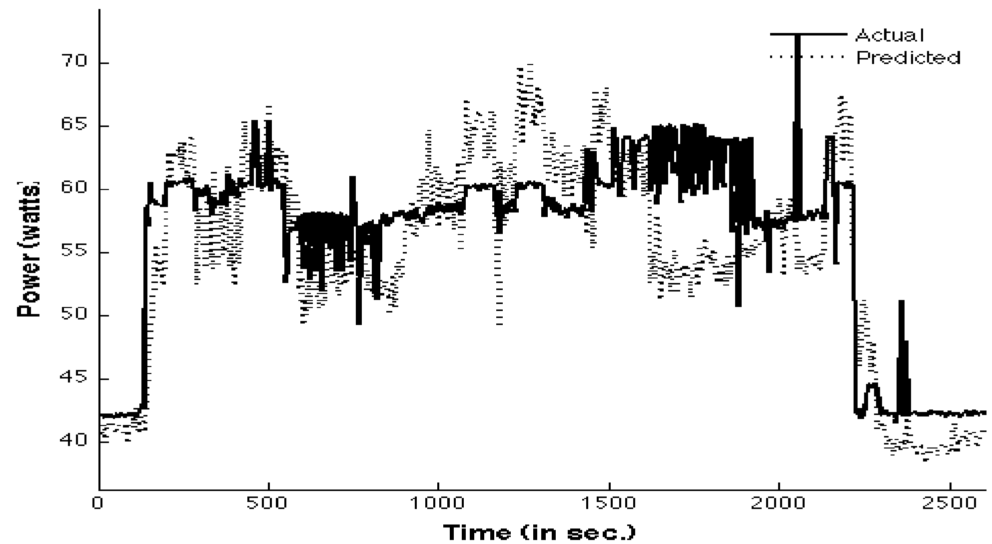
The Center for Advanced Computer Studies

UNIVERSITY OF LOUISIANA Lafayette

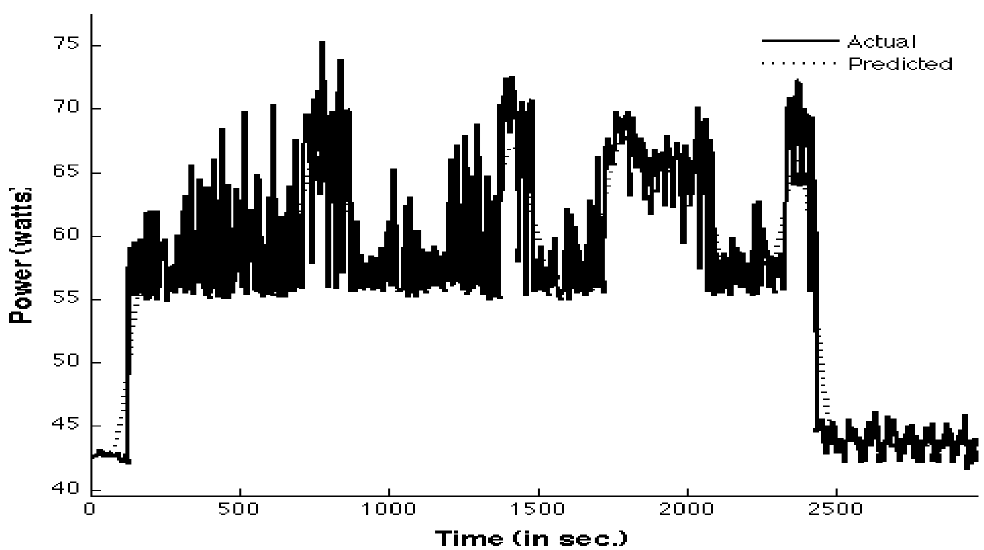
Results: Intel Nehalem



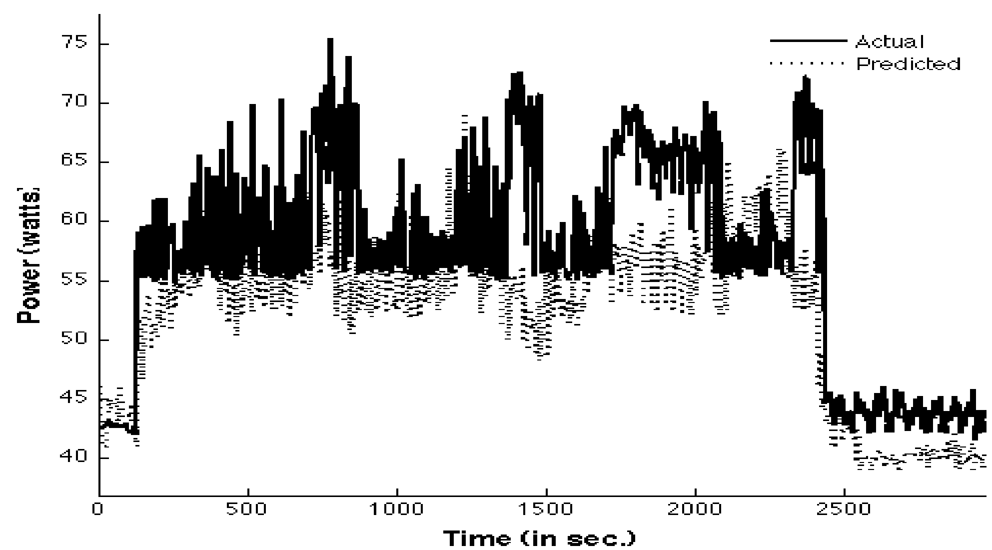
(a) Astar/CAP.



(b) Astar/AR(1)).



(c) Zeusmp/CAP.

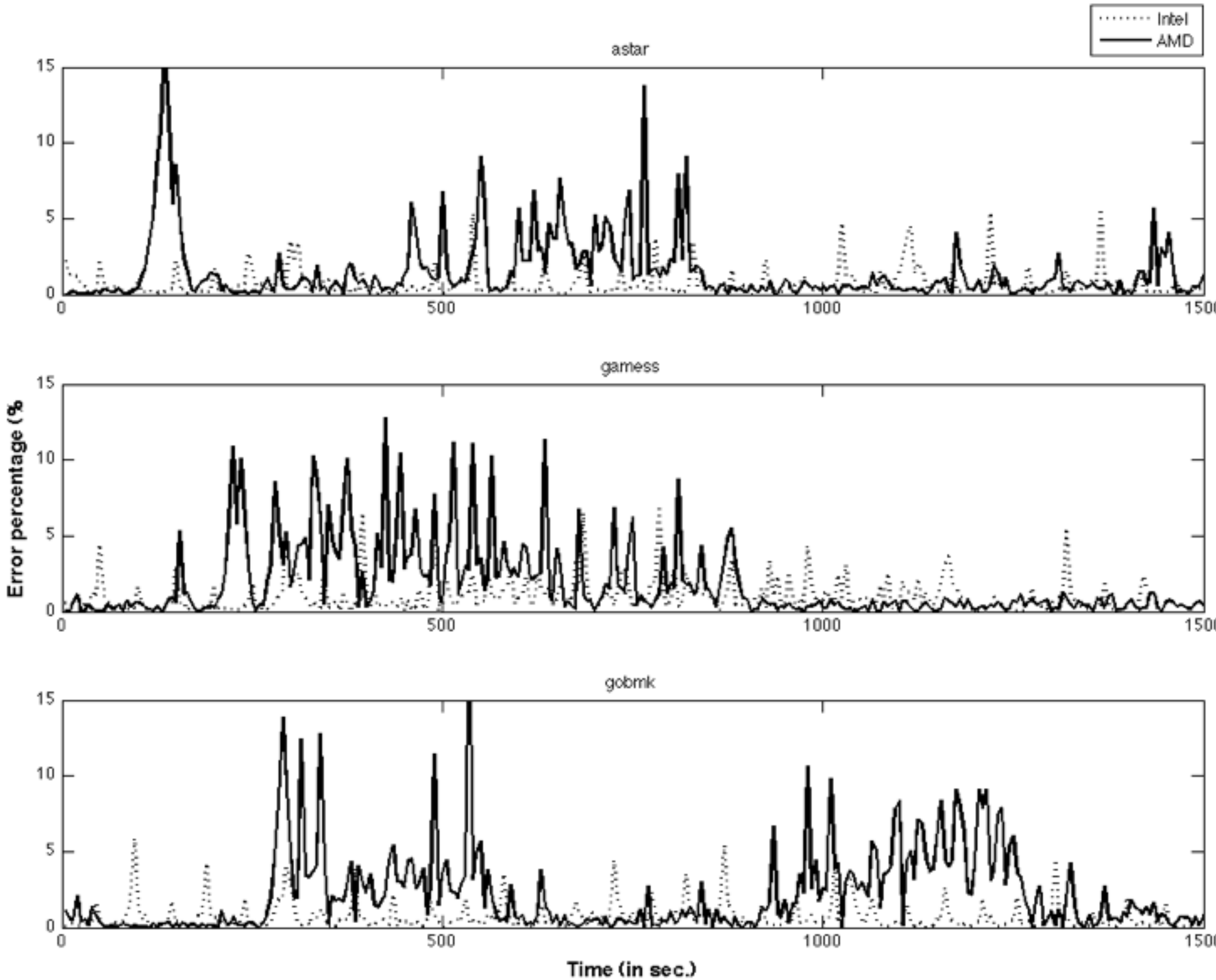


(d) Zeusmp/AR(1).

The Center for Advanced Computer Studies

UNIVERSITY OF LOUISIANA Lafayette

Results: Error - Other Benchmarks



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

Thermally-Aware Scheduling

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

The Center for Advanced Computer Studies

UNIVERSITY OF LOUISIANA Lafayette

Thermal Extensions to System Model

1. Applications have a length:

$$L(A, D_A, t)$$

and generate workload

$$U(A, D_A, t) = \lim_{n \to k_e} n \times W(p_i, d_i, t) \times L_n(A_n, D_{A_n}, t), 1 \leq i \leq p$$

2. For which we define *Thermal Equivalent of Application*

$$\Theta_A(A, D_A, T, t) = \frac{U(A, D_A, t)}{\lim\limits_{T \to T_{th}} J_e \times (T - T_{nominal})}$$

3. Which is used to generate *Thermal Efficiency to Completion*

$$\eta(A, D_A, T, t) = \frac{\Theta_A(A, D_A, T, t)}{\Theta_A(A_e, D_{A_e}, T_{me}, L_e)}$$

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

$$C_\theta(A, D_A, T, t) = \frac{\Theta_A(A, D_A, T, t)}{E_{sys}(A, D_A, t)}$$

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

Reducing Processor Temperatures

- Premise: Processor die temperature can be managed by controlling what threads execute over time
- Predict the next thread to run a logical CPU using TCAP for processor die temperature

Reducing Ambient Temperature

- Premise: Control system ambient temperature by managing load on logical CPUs so that overheated resources have time to recover
- Partition resources into categories based on predicted change in temperature
- Move workload from "HOT" resources towards "COLD" resources

Status, Plans, and Summary

The Center for Advanced Computer Studies

UNIVERSITY OF LOUISIANA Lafayette

Current Status

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

A Full-System Energy Model + Effective Prediction

- Development Complete
- Evaluation Complete
  - Intel + AMD processors
  - OpenSolaris (Solaris 11)
- Peer-reviewed
  - Conference/Workshop: 3
  - Journal: 1

Thermally-Aware Scheduling

- Design complete
- Prototype under development

The Center for Advanced Computer Studies

UNIVERSITY OF LOUISIANA Lafayette

Plan for Completion

ID	Task
1	Respond to review comments for [Lewis 2011]
2	Implement scheduler prototype in FreeBSD
3	Evaluate scheduler performance using parallel benchmarks
4	Document results and submit to archival journal
5	Create dissertation from Prospectus + output from previous task
6	Defend dissertation
7	Respond to comments from committee and Graduate School editor
8	Submit final version of document

Future Directions

- Extend beyond a single blade
  - Cluster, Grid, and Cloud Scheduling
  - MPI, OpenMP, and other environments
- Impact of operating system virtualization
- Extension of the thermal model in terms of the thermodynamics of computation

The Center for Advanced Computer Studies

UNIVERSITY OF LOUISIANA Lafayette

Questions?

The Center for Advanced Computer Studies

UNIVERSITY OF LOUISIANA Lafayette

Additional Material

The Center for Advanced Computer Studies

UNIVERSITY OF LOUISIANA Lafayette

This work was supported in part by the U.S. Department of Energy and by the Louisiana Board of Regents

The Center for Advanced Computer Studies

UNIVERSITY OF LOUISIANA Lafayette

Publications List

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.

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., Tzeng, N.-F., and Ghosh, S. 2011. Time series approximation of run-time energy consumption based on server workload. Under review for publication in ACM Transactions on Architecture and Code Optimization.