Chaotic Attractor Prediction for Server Run-time Energy Consumption

Adam Lewis, Nian-Feng Tzeng, Soumik Ghosh
The Center for Advanced Computer Studies
The University of Louisiana at Lafayette

The Center for Advanced Computer Studies

UNIVERSITY OF LOUISIANA Lafayette

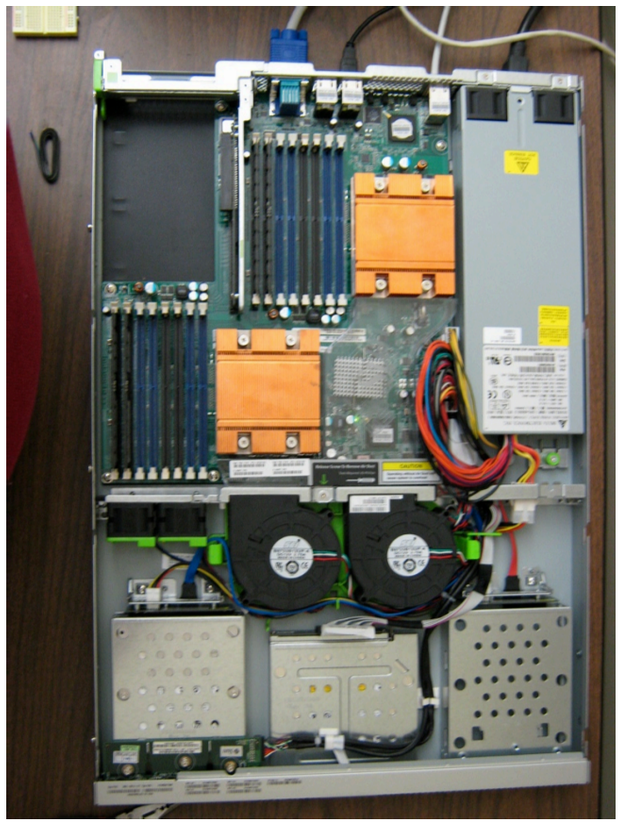
Agenda

- Introduction and problem statement
- CAP: prediction with chaotic time series
- Results
- Future work and next steps
- Summary

The Center for Advanced Computer Studies

UNIVERSITY OF LOUISIANA Lafayette

Full-system Power Models



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

$$\begin{aligned} E_{system} &= \alpha_0(E_{proc} + E_{mem}) \\ &+ \alpha_1 E_{em} \\ &+ \alpha_2 E_{board} \\ &+ \alpha_3 E_{hdd} \end{aligned}$$

[Lewis 2008]

- Continuous model ⇨ time series approximation
- Linear regression

Linear methods - A good idea?

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

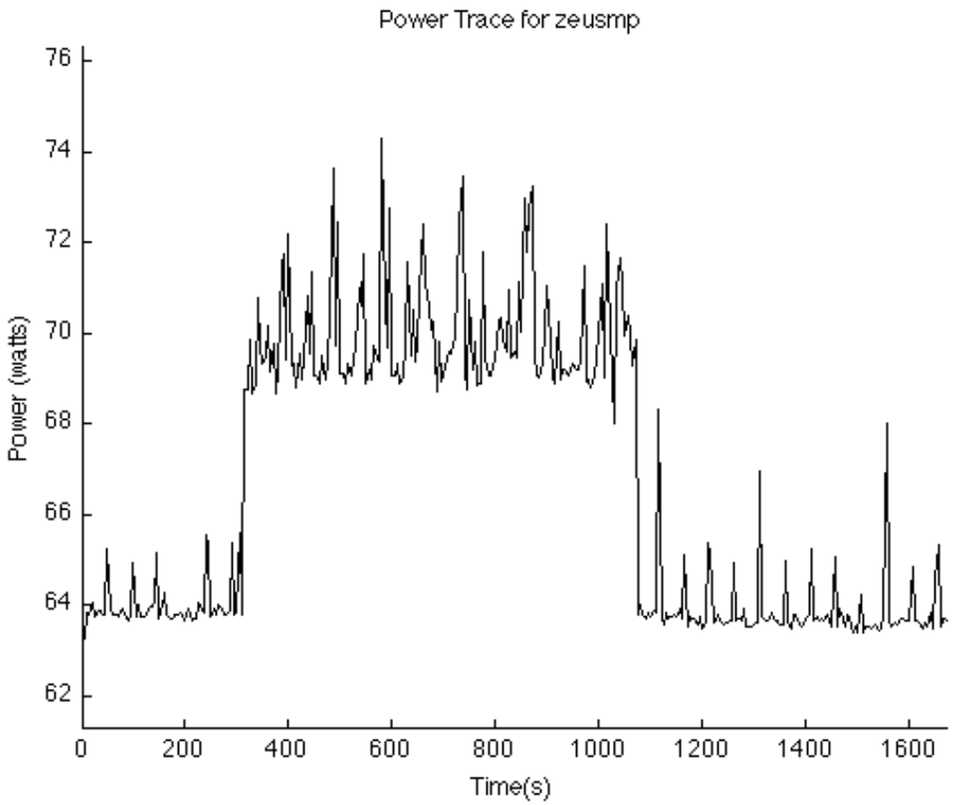
	AR		
Benchmark	**Avg Err %**	**Max Err %**	**RMSE**
astar	3.1%	8.9%	2.26
games	2.2%	9.3%	2.06
gobmk	1.7%	9.0%	2.30
zeusmp	2.8%	8.1%	2.14

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

Linear, non-linear, and chaotic



Non-linear?

Noise?

Error?

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

Creating and using a CAP

- One time process for new hardware
  - Create a training set for the process
  - Use training set to reconstruct state space
    - Embed using Taken's Theorem
    - Nearest Neighbors
- Solve resulting linear least squares problem

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.

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

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

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:

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

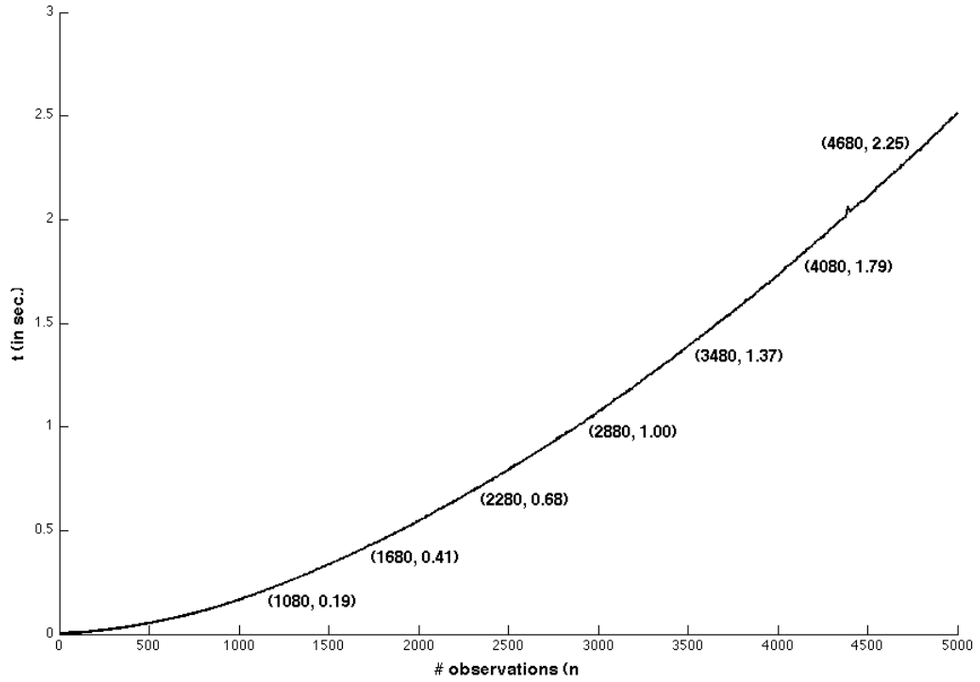
Forward prediction

- Explicit solution for our linear least squares problem:

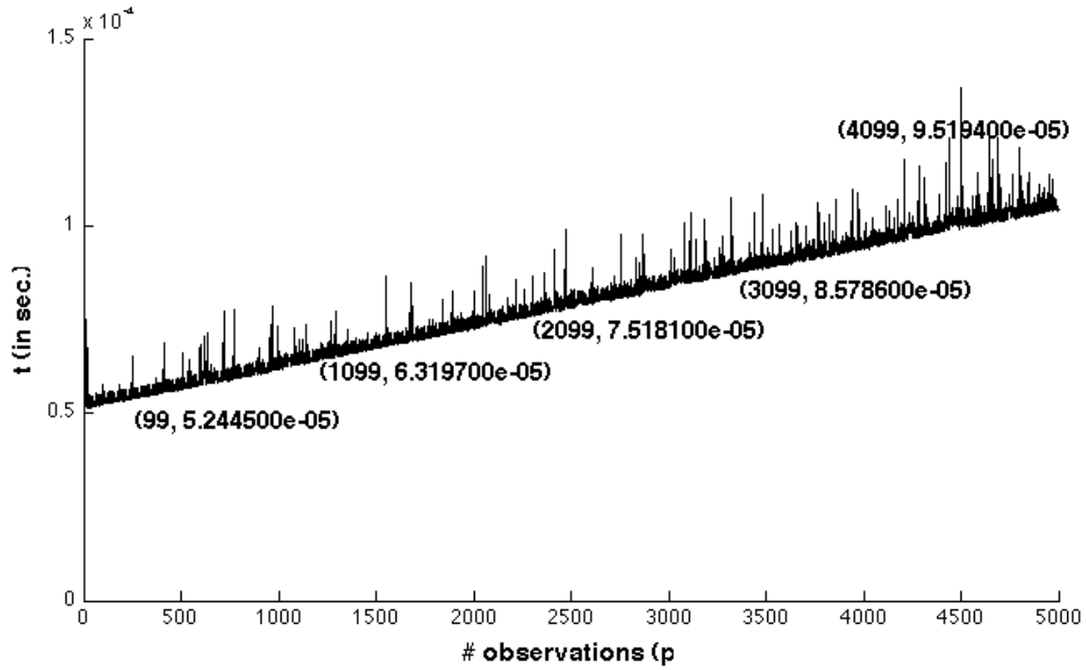
$$\hat{f}(x) = \frac{1}{n} \sum_{t=p+1}^{n+p} (s_2 - s_1 * (x - X_{t-1}))^2 * K_\beta((x - X_{t-1})/\beta)$$

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

Time Complexity



n future observations



p past observations

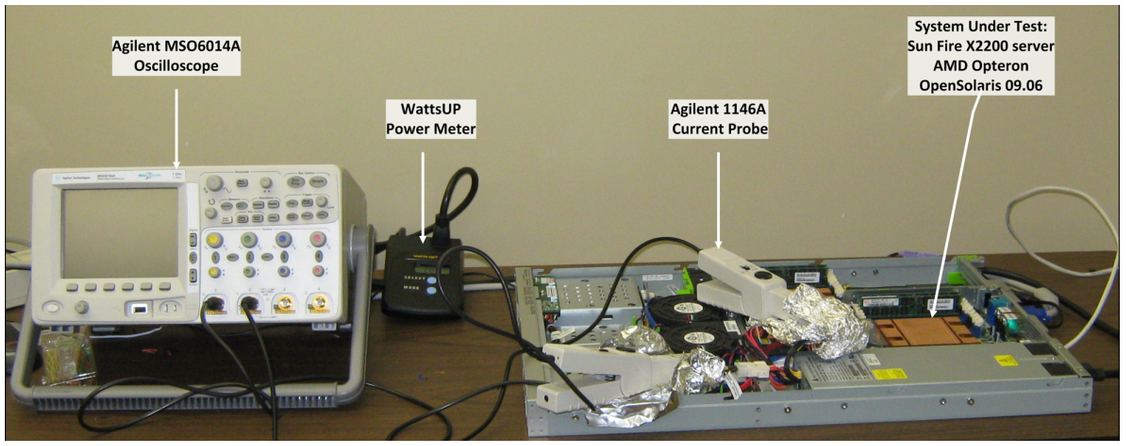
Creating a CAP: $O(n^2)$

Predicting with a CAP: $O(p)$

The Center for Advanced Computer Studies

UNIVERSITY OF LOUISIANA Lafayette

Evaluation



	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

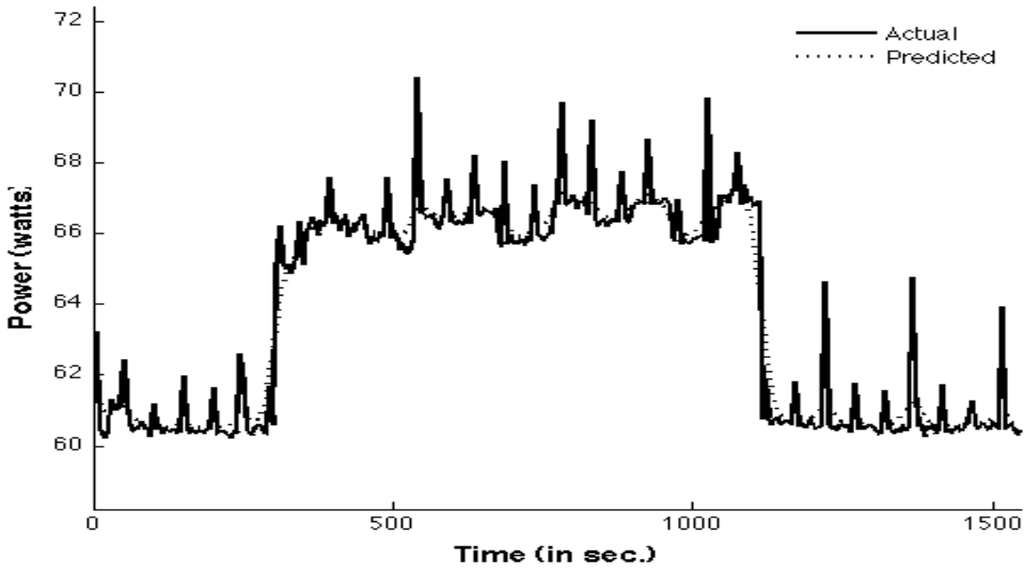
Evaluation Benchmarks

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

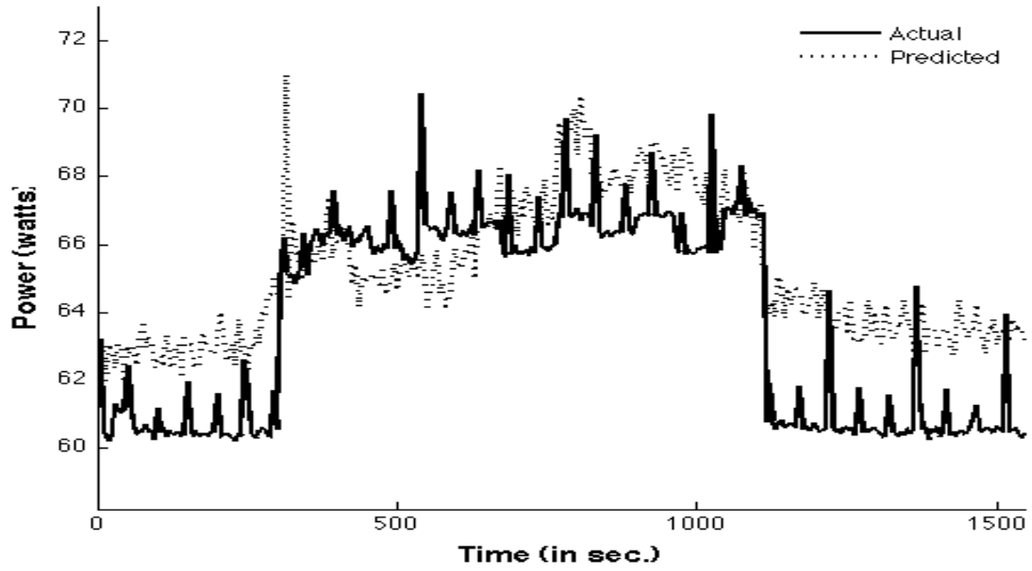
The Center for Advanced Computer Studies

UNIVERSITY OF LOUISIANA Lafayette

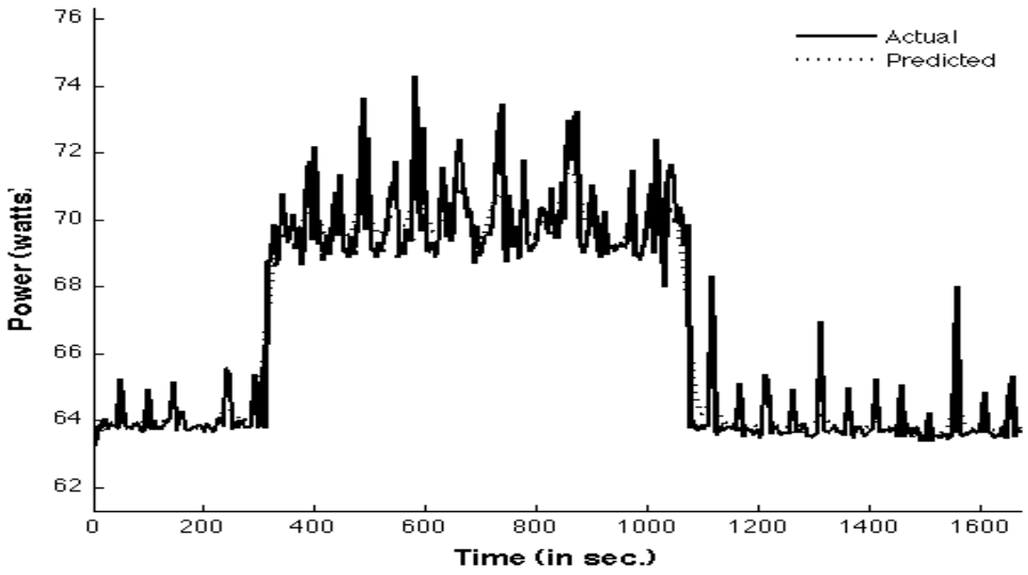
Results: AMD Opteron f10h



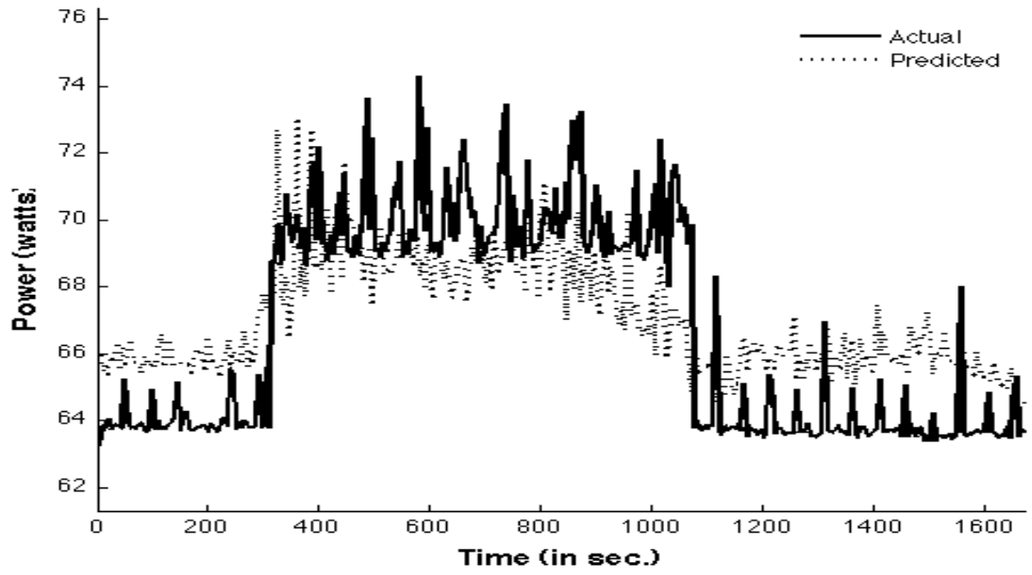
(a) Astar/CAP.



(b) Astar/AR(1)).



(c) Zeusmp/CAP.

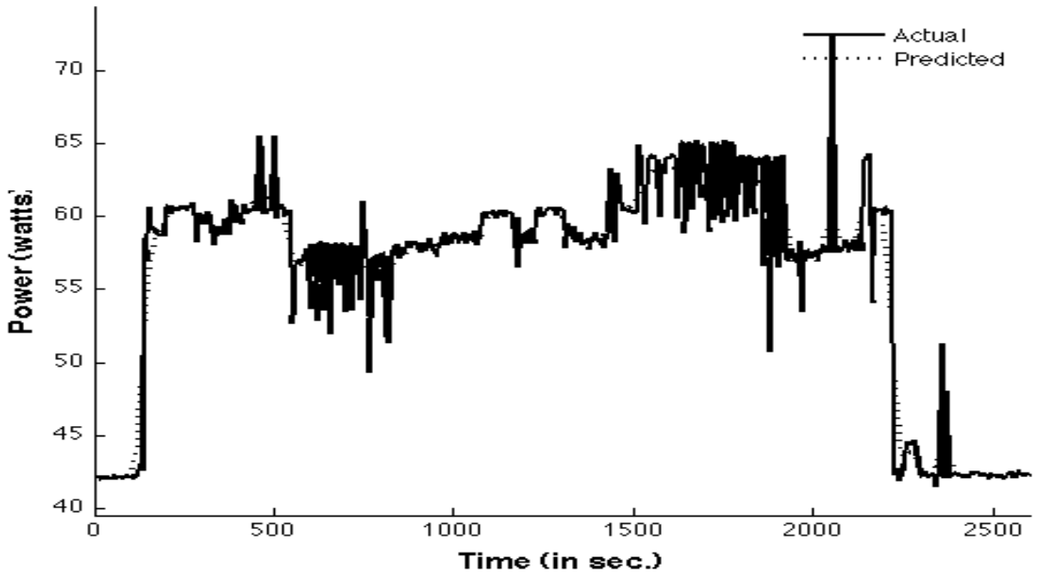


(d) Zeusmp/AR(1).

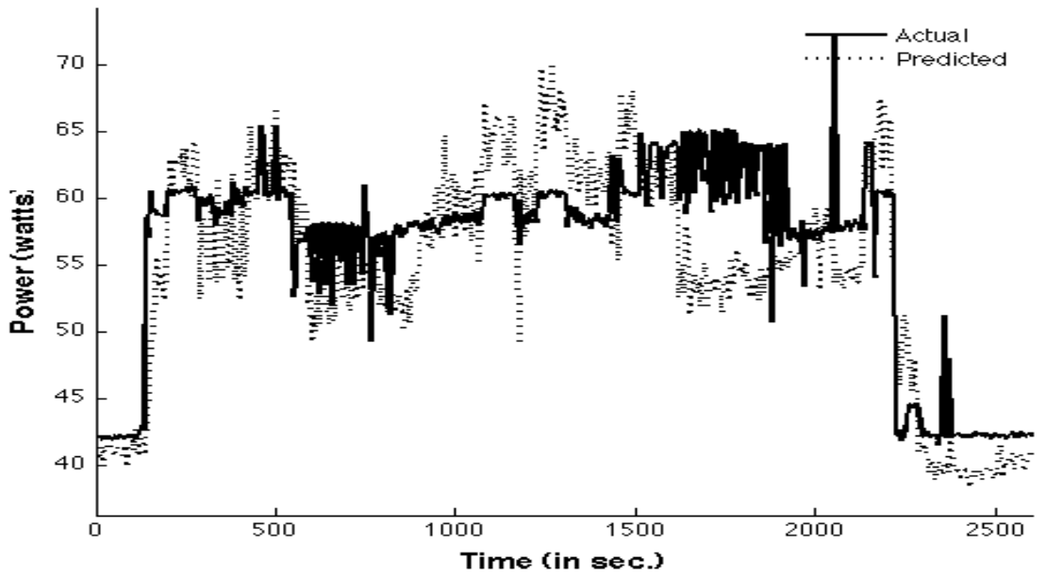
The Center for Advanced Computer Studies

UNIVERSITY OF LOUISIANA Lafayette

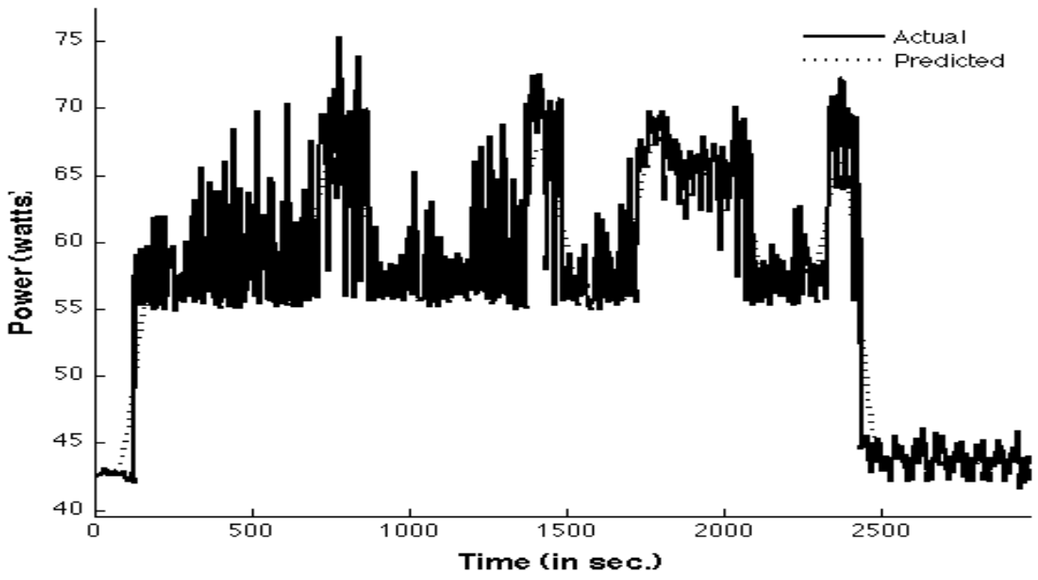
Results: Intel Nehalem



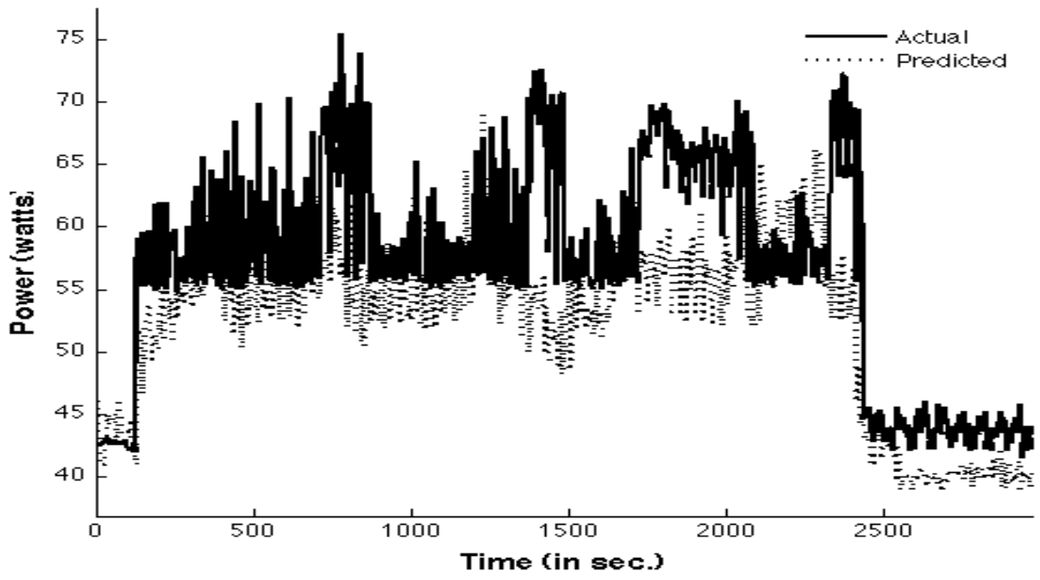
(a) Astar/CAP.



(b) Astar/AR(1)).



(c) Zeusmp/CAP.

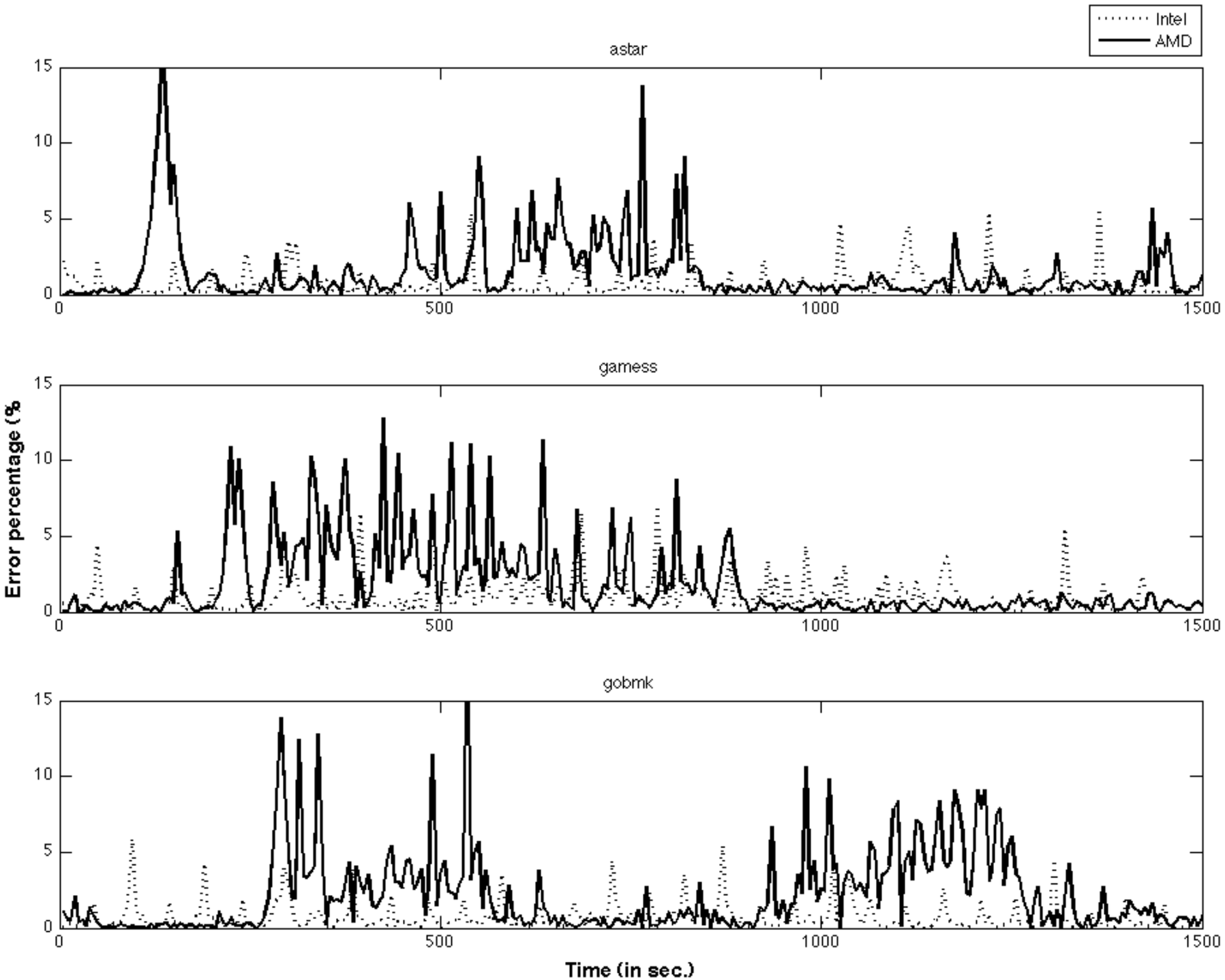


(d) Zeusmp/AR(1)

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

Conclusions

- Fast and accurate model
  - Addresses non-linearity
  - Addresses chaotic dynamics
- Future work
  - Other workloads
  - Other architectures

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

Questions?