

# Simulation of Power Consumption of Energy Efficient Cluster Hardware

EnA-HPC Conference 2010, Hamburg

**Timo Minartz**, Julian M. Kunkel, Thomas Ludwig

`minartz@informatik.uni-hamburg.de`

Scientific Computing  
Department of Informatics  
University of Hamburg

16-09-2010

# Motivation

## Why simulate what you can measure?

- Measurement devices with high accuracy are expensive
- Power meters are not easy to install into high-density racks
- Component-based measurement is not really possible with today's hardware
- Evaluation of different hard- and software characteristics possible
- Reproducible results

- 1 Goals and methodology
- 2 Model
- 3 Model input values
- 4 Evaluation
- 5 Conclusions

## Goals

- Estimate Energy-to-Solution (ETS) for given hard- and software
- Comparison of (simulated) energy consumption of the application with different power saving strategies
- Strategies can include rearrangements of the code
  - e.g. delay of network activity, I/O activities...
- Calculate minimal ETS with energy-proportional components
  - i.e. energy consumption is proportional to utilization
  - Upper bound for any energy saving strategy
- Integration in existing simulation environment *PIOSimHD*

## Methodology

- Periodic tracing of utilization of components
  - Processor, memory, network and I/O-subsystems
- Replay trace file and estimate energy consumption
- Model considers future utilization and control energy saving mechanisms
- Comparison of the estimated and measured energy consumption
- Assessing power estimation for realistic program traces

1 Goals and methodology

**2 Model**

3 Model input values

4 Evaluation

5 Conclusions

# Model

## Approach

- Estimation of power consumption for each component for each timestep
- Linear interpolation of power consumption
  - Based on minimal and maximal consumption values
- If idle, activate energy saving mechanism with different look-ahead strategies
  - ACPI model for the energy consumption and duration of the component's state change

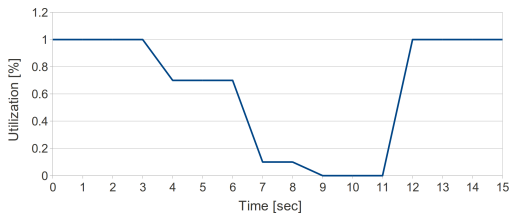
## Model (2)

### Strategies

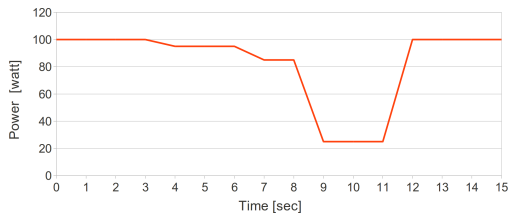
- Simple Strategy
  - Energy consumption without usage of explicit energy saving mechanism
- Optimal Strategy
  - Energy consumption with usage of low power state (0% utilization  $\Rightarrow$  low power state)
- Approach Strategy
  - Aggregate load to gain phases with zero utilization if possible
- Multiple State Strategy
  - Different power consumption for different utilization levels
    - e.g. P-states of the processor



# Example: Optimal Strategy



(a) Utilization



(b) Power consumption

- 1 Goals and methodology
- 2 Model
- 3 Model input values**
- 4 Evaluation
- 5 Conclusions

# Determining model input values

## Component power consumption

- Modeled components:
  - CPU, memory, disk, NIC, power supply
- Implementation of a micro benchmark to utilize the components for about 100 %
- Disassembling of one node to get 0 % utilization power consumption
- Cross-reference with data sheets
- Approximate power consumption for each component can be calculated

# Cluster tracing environment

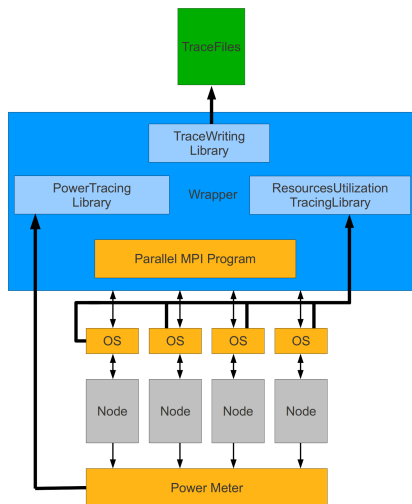


Figure: Cluster tracing environment

# Test environment

## Hard- and software

- 4 nodes (not power aware)
  - Dual Socket XEON (2003)
  - 1 Gigabyte RAM
  - Gigabit Ethernet Network
  - NFS, PVFS2
  - Ubuntu 8.04
- Each node connected to a channel of LMG 450 power meter

# Node power consumption

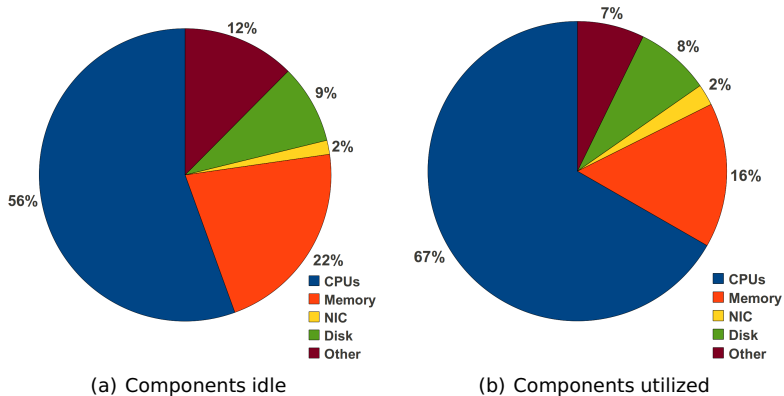
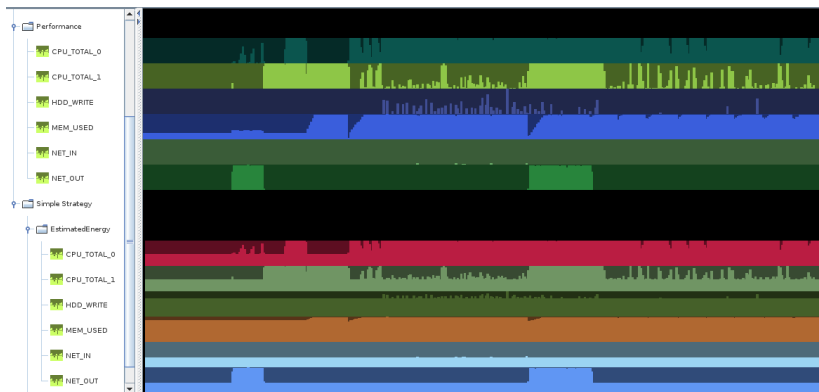


Figure: Distribution of power consumption (without power supply overhead)

# Sunshot screenshot



**Figure:** Component based utilization and estimated power consumption for the micro benchmark

- 1 Goals and methodology
- 2 Model
- 3 Model input values
- 4 Evaluation**
- 5 Conclusions



# Benchmark application

## **partdiff-par** (PDE-Solver)

- Supports parallel I/O
- The computation to communication ratio is flexible based on the input values for the boundary values of the matrix
- The component utilization depends on the check pointing frequency and the computation to communication ratio
- **Partdiff-par** is a real application and no synthetic benchmark

# Evaluation

## Observations

- Deviance of simulated and measured ETS  $\leq 5\%$
- Savings with different strategies (Optimal and Approach) at average about 10% and 13% respectively
- With energy proportional devices average savings of about 32% possible

# Strategies

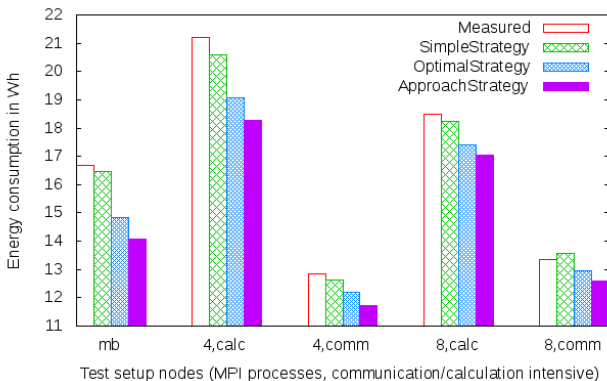


Figure: Energy consumption of various traces with different strategies

# Energy efficient sleeping

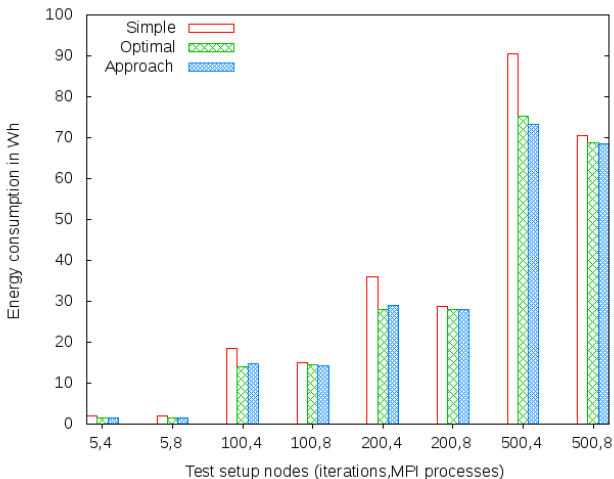


Figure: Comparison of four and eight calculating processors

# Energy proportional devices

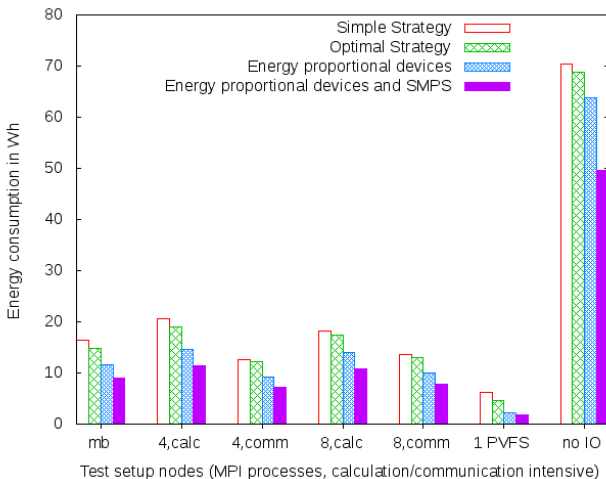


Figure: Energy consumption of energy proportional devices and SMPS

- 1 Goals and methodology
- 2 Model
- 3 Model input values
- 4 Evaluation
- 5 Conclusions**

## Conclusions

- Our prototype simulates power consumption and Energy-to-Solution with different hardware characteristics
- Different program configurations can be compared in terms of energy efficiency
- Monetary evaluation of hard- and software
- Payback period can be calculated
- Upper bound for power saving strategies can be determined

## Future work

- Trace usage of energy saving mechanism such as DVFS on our power-aware cluster
- Integrate ACPI power saving mechanism in the simulator inspired by the OS