Event-Driven Processor Power Management

Jan H. Schönherr
Jan Richling
Communication and Operating Systems
Technische Universität Berlin

Matthias Werner
Operating Systems Group
Chemnitz University of Technology

Gero Mühl
Architecture of Application Systems
University of Rostock
Overview

- State of the art
- Current problems
- Proposed solution
- Evaluation
- Summary
Saving Energy: State of the Art

> When nothing has to be done
  > Switch off components
  > ACPI defines C-states for processors
    > C0: working
    > C1..Cn: sleep-states
    > Managed by OS idle routine

> When only a little bit has to be done
  > Reduce performance of components
  > ACPI defines P-states for processors
    > Frequency/voltage combinations
    > P0: high performance
    > P1..Pn: lower performance
    > Managed by OS frequency governor

<table>
<thead>
<tr>
<th></th>
<th>MHz</th>
<th>TDP</th>
</tr>
</thead>
<tbody>
<tr>
<td>P0</td>
<td>2600</td>
<td>115.0 W</td>
</tr>
<tr>
<td>P1</td>
<td>2100</td>
<td>93.2 W</td>
</tr>
<tr>
<td>P2</td>
<td>1700</td>
<td>80.8 W</td>
</tr>
<tr>
<td>P3</td>
<td>1400</td>
<td>71.5 W</td>
</tr>
<tr>
<td>P4</td>
<td>800</td>
<td>53.2 W</td>
</tr>
</tbody>
</table>

P-states of an AMD Opteron 8435
Current Approach: Time-Driven Governor

- Example: Linux kernel compilation

- Severe performance degradation
- Increased energy consumption

<table>
<thead>
<tr>
<th>Governor</th>
<th>Energy</th>
</tr>
</thead>
<tbody>
<tr>
<td>Performance</td>
<td>329 Wh</td>
</tr>
<tr>
<td>Ondemand (10 ms)</td>
<td>368 Wh</td>
</tr>
<tr>
<td>Ondemand (200 ms)</td>
<td>475 Wh</td>
</tr>
</tbody>
</table>

Energy consumption for a parallelism degree of 1
Analyzing Problems: Problem 1

> Governor determines load by polling

> Incorrect results in case of
  > small activity phases/short running processes, and
  > a low overall degree of parallelism

→ No increase of frequency
→ Performance degradation
Analyzing Problems: Problem 2

> Governor reduces frequency of idling cores
> Scheduler favors idle cores

> New load is scheduled on cores with low frequency
  > in case of a low overall degree of parallelism

→ Performance degradation
Problem: Separation of Concerns

> Clear separation between
  > Scheduling (which task when and where) → Scheduler
  > Energy control (clock frequencies of cores) → Governor

> But
  > The functions mentioned above depend on each other in reality!
  > Overhead: Governor reconstructs information already available to scheduler

> Energy consumption is a non-functional property
  > Separation of concerns may raise problems
Approach: Integration

> Non-functional interdependency between scheduling and frequency control
  → Integration instead of separation
  → Combination of scheduler and governor

> Two-step approach
  1. Event-driven frequency control by scheduler

  2. Adaption of scheduler (future work)
     > Scheduling in space according to power states of cores
     > Concentration of load
Event-driven Frequency Control

> Load changes trigger frequency transitions
> Purely event-driven: No polling, no consideration of load

> Input events: State change from idle to load or vice versa
> Output events: State change from low to high frequency or vice versa

> Optimization
  > Delays for transitions
  > Minimum time to stay in high or low frequency
Implementation

> Based on current Linux kernel (2.6.32-rc5)

> Scheduler interface
  > Scheduler notifies CPUFreq framework whenever load changes

> New scheduler governor
  > Initiates P-state transitions based on load changes

> Modified CPUFreq driver
  > Driver is now used in atomic contexts
  > Must be (sleep-)lock-free

```c
on_load_change() {
    if( (is_idle && is_low) ||
        (is_load && is_high) ) {
        delete_timer();
    } else {
        if( is_due(next()) )
            initiate_transition();
        else
            add_timer(initiate_transition, next());
    }
}
```
Pseudo code of scheduler governor
Evaluation: Hardware

> Quad AMD Opteron 8435
  > Latest generation server
  > Four sockets
  > 45 nm K10 hexa-core processors
    > codename Istanbul
  > Frequencies: 0.8, 1.4, 1.7, 2.1, and 2.6 GHz
  > Minimum idle consumption: 280W

> AMD Phenom 9950
  > Previous generation desktop
  > 65 nm K10 quad-core processor
    > codename Agena
  > Frequencies: 1.3 and 2.6 GHz
  > Minimum idle consumption: 84W
### Evaluation: Quad AMD Opteron 8435

<table>
<thead>
<tr>
<th>Governor</th>
<th>Run-Time</th>
<th>Power</th>
<th>Energy</th>
<th>Relative Energy</th>
<th>Relative EDP</th>
</tr>
</thead>
<tbody>
<tr>
<td>Performance</td>
<td>1:03:33 h</td>
<td>310 W</td>
<td>329 Wh</td>
<td>32 Wh</td>
<td>34 Whh</td>
</tr>
<tr>
<td>Scheduler</td>
<td>1:05:12 h</td>
<td>298 W</td>
<td>324 Wh</td>
<td>20 Wh</td>
<td>21 Whh</td>
</tr>
<tr>
<td>Ondemand (10ms)</td>
<td>1:14:58 h</td>
<td>294 W</td>
<td>368 Wh</td>
<td>18 Wh</td>
<td>23 Whh</td>
</tr>
<tr>
<td>Ondemand (200ms)</td>
<td>1:37:32 h</td>
<td>292 W</td>
<td>475 Wh</td>
<td>20 Wh</td>
<td>32 Whh</td>
</tr>
<tr>
<td>Powersave</td>
<td>3:09:06 h</td>
<td>284 W</td>
<td>895 Wh</td>
<td>13 Wh</td>
<td>40 Whh</td>
</tr>
</tbody>
</table>
### Evaluation: AMD Phenom 9950

#### Performance Governor (ratio)
- **Run-Time:** 1:13:39 h
- **Power:** 132 W
- **Energy:** 162 Wh
- **Relative Energy:** 59 Wh
- **Relative EDP:** 72 Whh

#### Scheduler Governor (ratio)
- **Run-Time:** 1:14:35 h
- **Power:** 120 W
- **Energy:** 149 Wh
- **Relative Energy:** 45 Wh
- **Relative EDP:** 55 Whh

#### Ondemand Governor, 10 ms (ratio)
- **Run-Time:** 1:19:43 h
- **Power:** 117 W
- **Energy:** 156 Wh
- **Relative Energy:** 44 Wh
- **Relative EDP:** 59 Whh

#### Ondemand Governor, 200 ms (ratio)
- **Run-Time:** 1:26:26 h
- **Power:** 116 W
- **Energy:** 167 Wh
- **Relative Energy:** 46 Wh
- **Relative EDP:** 66 Whh

#### Powersave
- **Run-Time:** 2:18:43 h
- **Power:** 107 W
- **Energy:** 246 Wh
- **Relative Energy:** 52 Wh
- **Relative EDP:** 120 Whh
Summary

- Integration of scheduler and frequency governor
- Event-driven frequency control

- Implementation based on Linux

- Reduced energy consumption
- No performance degradation
- Improved interactive behavior

- Promising foundation for future extensions
Future Work

> Event-driven governor
  > Further evaluation of parameters
  > Adaption of scheduling in space
    > Concentration of load
  > Energy efficiency by considering application behavior
  > Coordinated control of P-states and C-states

> Further approaches
  > Consideration of hardware-driven frequency scaling
    (Intel’s Turbo Boost Technology, AMD’s expected Core Performance Boost)
  > Consideration of SMT
Evaluation: Intel Core 2 Quad Q9400

> Intel Core 2 Quad Q9400
  > Previous generation desktop
  > 45 nm quad-core processor (codename Yorkfield)
  > Frequencies: 2.0, 2.33 and 2.67 GHz
Evaluation: Dual Intel Xeon X5570

- Dual Intel Xeon X5570
  - Latest generation server (Nehalem architecture)
  - Two sockets
  - 45 nm quad-core processors (codename Gainestown)
  - Frequencies: 1.6 – 2.93 GHz in 133 MHz steps, 2/2/3/3 turbo
Current Approach: Time-Driven Governor

- Example: Linux kernel compilation

> Severe performance degradation
> Increased energy consumption

<table>
<thead>
<tr>
<th>Governor</th>
<th>Energy</th>
</tr>
</thead>
<tbody>
<tr>
<td>Performance</td>
<td>329 Wh</td>
</tr>
<tr>
<td>Ondemand (10 ms)</td>
<td>368 Wh</td>
</tr>
<tr>
<td>Ondemand (200 ms)</td>
<td>475 Wh</td>
</tr>
</tbody>
</table>

Energy consumption for a parallelism degree of 1
Evaluation: Quad AMD Opteron 8435

<table>
<thead>
<tr>
<th>Governor</th>
<th>Run-Time</th>
<th>Power</th>
<th>Energy</th>
<th>Relative Energy</th>
<th>Relative EDP</th>
</tr>
</thead>
<tbody>
<tr>
<td>Performance</td>
<td>1:03:33 h</td>
<td>310 W</td>
<td>329 Wh</td>
<td>32 Wh</td>
<td>34 Whh</td>
</tr>
<tr>
<td>Scheduler</td>
<td>1:05:12 h</td>
<td>298 W</td>
<td>324 Wh</td>
<td>20 Wh</td>
<td>21 Whh</td>
</tr>
<tr>
<td>Ondemand (10ms)</td>
<td>1:14:58 h</td>
<td>294 W</td>
<td>368 Wh</td>
<td>18 Wh</td>
<td>23 Whh</td>
</tr>
<tr>
<td>Ondemand (200ms)</td>
<td>1:37:32 h</td>
<td>292 W</td>
<td>475 Wh</td>
<td>20 Wh</td>
<td>32 Whh</td>
</tr>
<tr>
<td>Powersave</td>
<td>3:09:06 h</td>
<td>284 W</td>
<td>895 Wh</td>
<td>13 Wh</td>
<td>40 Whh</td>
</tr>
</tbody>
</table>
### Evaluation: AMD Phenom 9950

<table>
<thead>
<tr>
<th>Governor</th>
<th>Run-Time</th>
<th>Power</th>
<th>Energy</th>
<th>Relative Energy</th>
<th>Relative EDP</th>
</tr>
</thead>
<tbody>
<tr>
<td>Performance</td>
<td>1:13:39 h</td>
<td>132 W</td>
<td>162 Wh</td>
<td>59 Wh</td>
<td>72 Whh</td>
</tr>
<tr>
<td>Scheduler</td>
<td>1:14:35 h</td>
<td>120 W</td>
<td>149 Wh</td>
<td>45 Wh</td>
<td>55 Whh</td>
</tr>
<tr>
<td>Ondemand (10ms)</td>
<td>1:19:43 h</td>
<td>117 W</td>
<td>156 Wh</td>
<td>44 Wh</td>
<td>59 Whh</td>
</tr>
<tr>
<td>Ondemand (200ms)</td>
<td>1:26:26 h</td>
<td>116 W</td>
<td>167 Wh</td>
<td>46 Wh</td>
<td>66 Whh</td>
</tr>
<tr>
<td>Powersave</td>
<td>2:18:43 h</td>
<td>107 W</td>
<td>246 Wh</td>
<td>52 Wh</td>
<td>120 Whh</td>
</tr>
</tbody>
</table>
Evaluation: Intel Core 2 Quad Q9400

> Intel Core 2 Quad Q9400
  > Previous generation desktop
  > 45 nm quad-core processor (code-name Yorkfield)
  > Frequencies: 2.0, 2.33 and 2.67 GHz
Evaluation: Dual Intel Xeon X5570

- Dual Intel Xeon X5570
  - Latest generation server (Nehalem architecture)
  - Two sockets
  - 45 nm quad-core processors (code-name Gainestown)
  - Frequencies: 1.6 – 2.93 GHz in 133 MHz steps, 2/2/3/3 turbo