Genetic Improvement of Energy Usage is only as Reliable as the Measurements are Accurate

Saemundur Oskar Haraldsson

University of Stirling

April, 2015

John R. Woodward  Co-author
Edmund K. Burke  Supervisor

http://daase.cs.ucl.ac.uk/
Overview

1. Motivation
2. Genetic Improvement
3. Energy in Computation
4. Summary
Motivation

“If you can not measure it, you can not improve it.”

– Lord Kelvin
Motivation

“If you can not measure it, you can not improve it.”
– Lord Kelvin

Software energy conservation

Energy optimisation with Genetic Improvement (GI)

Measuring energy with intent on improving
Motivation

“If you can not measure it, you can not improve it.”
– Lord Kelvin

Software energy conservation
- Environmental
- Financial
- Hardware is only as efficient as the software driving it.

Energy optimisation with Genetic Improvement (GI)

Measuring energy with intent on improving
Motivation

“If you can not measure it, you can not improve it.”

– Lord Kelvin

Software energy conservation

- Environmental
- Financial
- Hardware is only as efficient as the software driving it.

Energy optimisation with Genetic Improvement (GI)

- Unintuitive for manual improvements [4].
- Improving software is Multi-objective.

Measuring energy with intent on improving
Motivation

“If you can not measure it, you can not improve it.”

– Lord Kelvin

Software energy conservation

- Environmental
- Financial

Hardware is only as efficient as the software driving it.

Energy optimisation with Genetic Improvement (GI)

- Unintuitive for manual improvements [4].
- Improving software is Multi-objective.

Measuring energy with intent on improving

- Energy measurements are complicated [2].
- Be wary of overly simple surrogates [3].
Genetic Improvement

Automatic software adjustment operating directly on the source code, treating it as the genetic material [1].

- Improving the software by readjusting the source code.
- Works well on multiple objectives [5].
- Improvements are based on fitness evaluations.
Genetic Improvement

Automatic software adjustment operating directly on the source code, treating it as the genetic material [1].

- Improving the software by readjusting the source code.
- Works well on multiple objectives [5].
- Improvements are based on fitness evaluations.

1 Image courtesy of foto76 at FreeDigitalPhotos.net
Electricity as the current drawn and voltage over time.  
4 levels to optimize on:
Energy in Computation

Electricity as the current drawn and voltage over time.
4 levels to optimize on:

- Hardware optimisation
Energy in Computation

Electricity as the current drawn and voltage over time.
4 levels to optimize on:

1. Hardware optimisation
2. Optimizing the OS or kernel.
Energy in Computation

Electricity as the current drawn and voltage over time. 4 levels to optimize on:

1. Hardware optimisation
2. Optimizing the OS or kernel.
3. Minimizing the amount of computing used for a particular task.
Energy in Computation

Electricity as the current drawn and voltage over time. 4 levels to optimize on:

1. Hardware optimisation
2. Optimizing the OS or kernel.
3. Minimizing the amount of computing used for a particular task.
4. End user specific energy conservations
Measuring energy in computation

Physical measurements

- The whole system.
- Each hardware component.

Alternatives.

- Simulation
- Timing, CPU counts, Memory access, etc.
Measuring energy in computation

Physical measurements

- The whole system.
- Each hardware component.
- Not suitable for after launch adaptations.

Alternatives.

- Simulation
- Timing, CPU counts, Memory access, etc.
Measuring energy in computation

Physical measurements

- The whole system.
- Each hardware component.
- Not suitable for after launch adaptations.

Alternatives.

- Simulation
- Timing, CPU counts, Memory access, etc.
- Be vary of overly simple alternatives.

2

---

2 Image courtesy of TAW4 at FreeDigitalPhotos.net
## Things to consider

<table>
<thead>
<tr>
<th>Genetic Improvement</th>
<th>Useful for “green” optimisation of software</th>
</tr>
</thead>
<tbody>
<tr>
<td>Physical Measurements</td>
<td>Accurate if properly applied but impractical</td>
</tr>
<tr>
<td>Alternative Measurements</td>
<td>Have to be implemented on case by case basis.</td>
</tr>
</tbody>
</table>

Beware of overly simple surrogates

**Conclusion**
## Things to consider

<table>
<thead>
<tr>
<th>Genetic Improvement</th>
<th>Useful for “green” optimisation of software</th>
</tr>
</thead>
<tbody>
<tr>
<td>Physical Measurements</td>
<td>Accurate if properly applied but impractical</td>
</tr>
<tr>
<td>Alternative Measurements</td>
<td>Have to be implemented on case by case basis.</td>
</tr>
<tr>
<td><strong>Beware</strong></td>
<td>of overly simple surrogates</td>
</tr>
<tr>
<td>Conclusion</td>
<td>If you can not measure it <em>accurately</em>, you can not improve it <em>reliably</em></td>
</tr>
<tr>
<td>Motivation</td>
<td>Genetic Improvement</td>
</tr>
<tr>
<td>------------</td>
<td>---------------------</td>
</tr>
<tr>
<td>6/7</td>
<td>Thanks</td>
</tr>
</tbody>
</table>


