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

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.

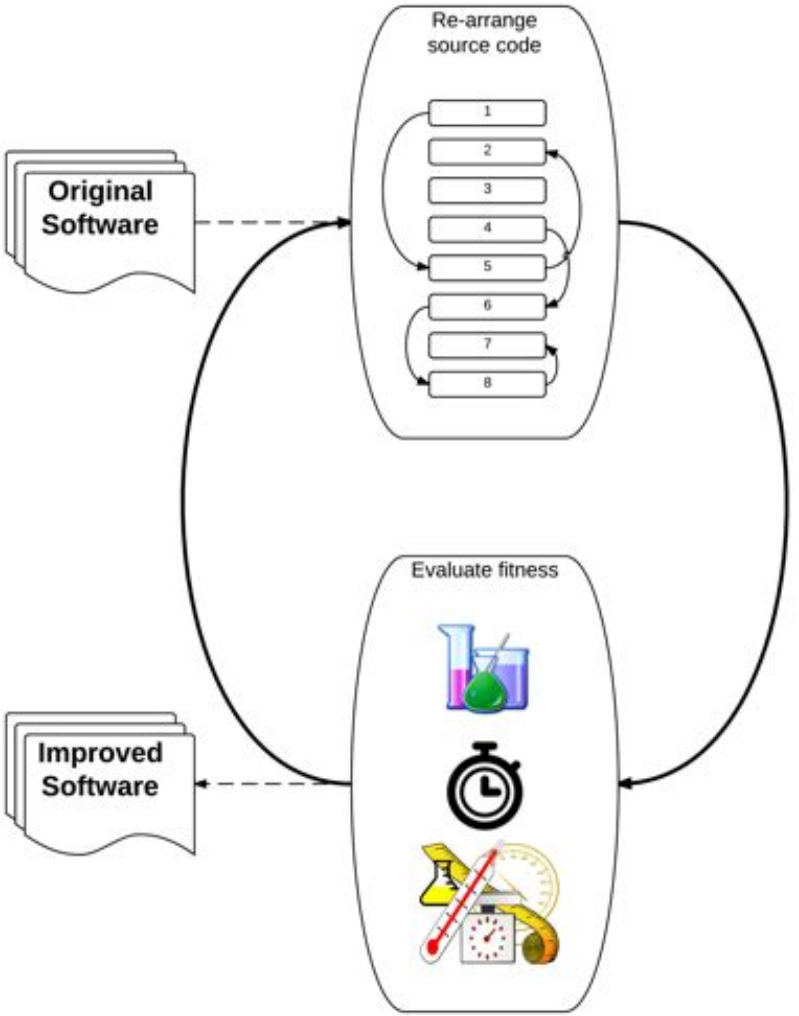
Measuring energy with intent on improving

- 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.



1

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

Energy in Computation

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

Energy in Computation

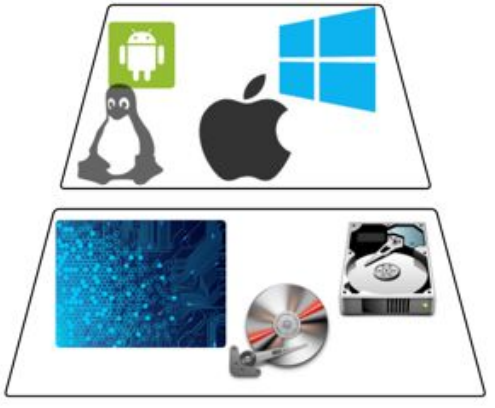
Electricity as the current drawn and voltage over time.
4 levels to optimize on:

1. Hardware optimisation

Energy in Computation

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

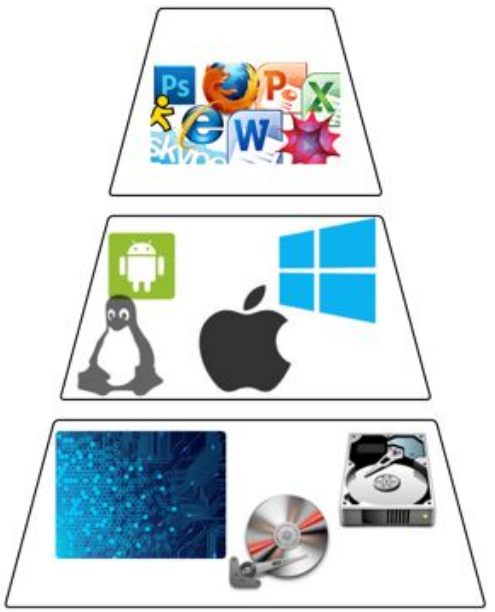
1. Hardware optimisation
2. Optimizing the OS or kernel.



Energy in Computation

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

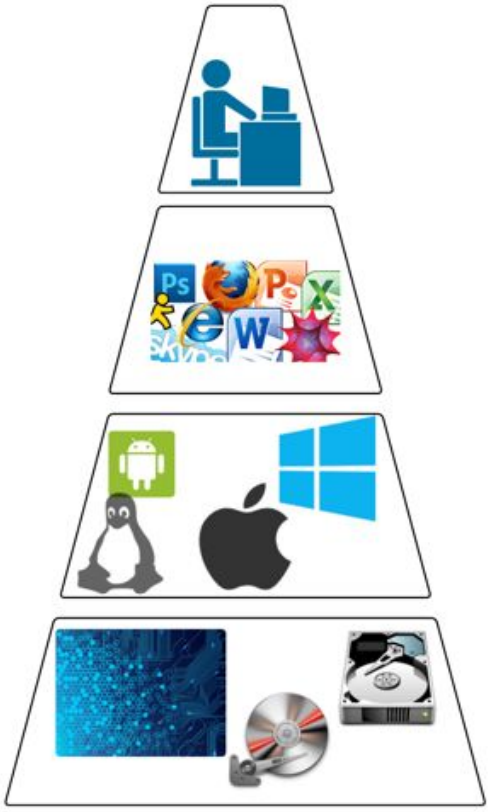
1. Hardware optimisation
2. Optimizing the OS or kernel.
3. Minimizing the amount of computing used for a particular task.



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

Genetic Improvement	Useful for “green” optimisation of software
Physical Measurements	Accurate if properly applied but impractical
Alternative Measurements	Have to be implemented on case by case basis. Beware of overly simple surrogates
Conclusion	

Things to consider

Genetic Improvement	Useful for “green” optimisation of software
Physical Measurements	Accurate if properly applied but impractical
Alternative Measurements	Have to be implemented on case by case basis. Beware of overly simple surrogates
Conclusion	If you can not measure it *accurately*, you can not improve it *reliably*

Thanks

Any questions?

References

[1] W. B. Langdon and M. Harman. Optimising Existing Software with Genetic Programming. *IEEE Transactions on Evolutionary Computation*, PP(99):1–18, 2014.

[2] Irene Manotas, Cagri Sahin, James Clause, Lori Pollock, and Kristina Winbladh. Investigating the impacts of web servers on web application energy usage. In *Green and Sustainable Software (GREENS), 2013 2nd International Workshop on*, pages 16–23. IEEE, 2013.

[3] Cagri Sahin, Furkan Cayci, Irene Lizeth Manotas Gutiérrez, James Clause, Fouad Kiamilev, Lori Pollock, and Kristina Winbladh. Initial explorations on design pattern energy usage. In *2012 First International Workshop on Green and Sustainable Software (GREENS)*, pages 55–61. IEEE, 2012.

[4] Cagri Sahin, Lori Pollock, and James Clause. How do code refactorings affect energy usage? In *Proceedings of the 8th ACM/IEEE International Symposium on Empirical Software Engineering and Measurement - ESEM '14*, pages 1–10. ACM, 2014.

[5] White, R., Clark, J., Jacob, J. and Poulding, S. Searching for Resource-Efficient Programs : Low-Power Pseudorandom Number Generators. In Keijzer, M, editor, *GECCO'08, 10th annual conference on Genetic and evolutionary computation*, pages 1775-1782, Atlanta, GA, USA, 2008.