On The Implications of Language Constructs for Concurrent Execution in the Energy Efficiency of Multicore Applications

Gustavo Pinto, Fernando Castor - {ghlp, castor}@cin.ufpe.br

The Problem

The performance of the existing constructs for concurrent execution is reasonably well-understood. But, little is known about the energy-efficiency of these techniques.

"Race to idle": Is a common belief that faster applications will also consume less energy.

But... it is not necessarily true!

- **Factors analyzed:**
  - **Internal factors**
    - **Concurrent constructs**
    - **Number of threads**
    - **Resource usage**
  - **External factors**
    - **Clock frequency**
    - **JVM providers**

The Benchmarks

- **N-Queens: CPU-bound**
- **Mandelbrot: CPU-bound**
- **LargestImage: IO-bound**
- **Knucleotide: 23% doing IO**

The Study

4 Benchmark Apps * **4 Variants (3 conc. + 1 seq.)** * **3 JVMs** * **Number. of Threads** * **2 Clock Frequencies**

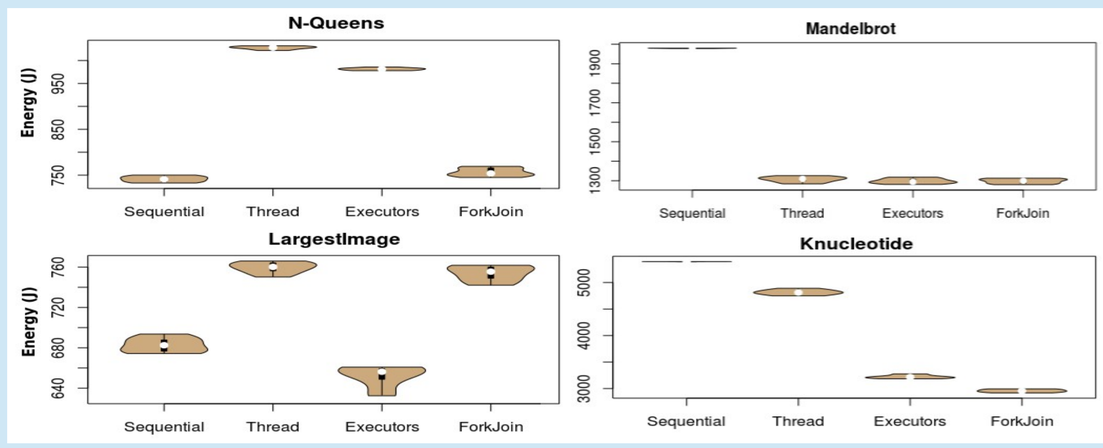
Intel(R) Xeon(R), 2.13 GHz, 4 cores/8 threads 16Gb of memory
Linux 64-bit, kernel 3.0.0.-31-server.

The Conclusions

- Some factors create variations, but some others do not.
  - Do
    - Nature of the problem
    - Concurrent programming construct
  - Do not
    - JVM implementation
    - CPU clock frequency
- We also found out that, for concurrent software, faster does not *always* mean greener

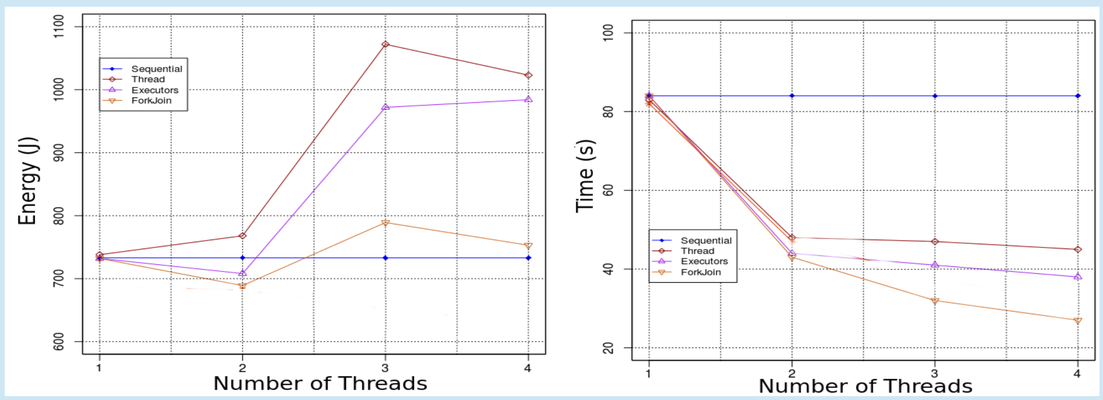
Some Results

Varying Concurrent Construct



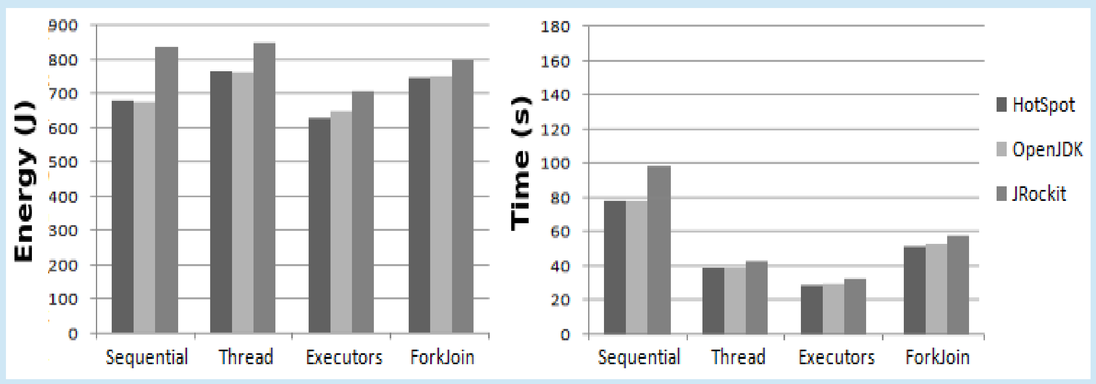
Different concurrent constructs could produce unexpected results.

Varying number of Threads: N-Queens



Improvements in performance do not necessarily mean less energy consumed.

Varying JVM providers: LargestImage



Different JVM could increase in more than 10% of the energy consumed!

Future work

- To conduct a broader-scoped study.
- The results of this new study will provide input for us to derive a catalog energy code smell for concurrent software.
- Then we plan to proceed with the design of refactoring catalog that will enable application programmer to safely restructure their applications to use less energy

STUDENT RESEARCH COMPETITION

This work is partially supported by National Institute of Science and Technology for Software Engineering (INES), by CNPq (grant 573964/2008-4) and FACEPE (grant APQ- 1037-1.03/08).
Gustavo is supported by CAPES.

INES

VIRTUS IMPAVIDA
UFPE