Performance and energy issues in modern processors

E. Calore A. Gabbana S. F. Schifano R. Tripiccione

INFN Ferrara and Università degli Studi di Ferrara, Italy

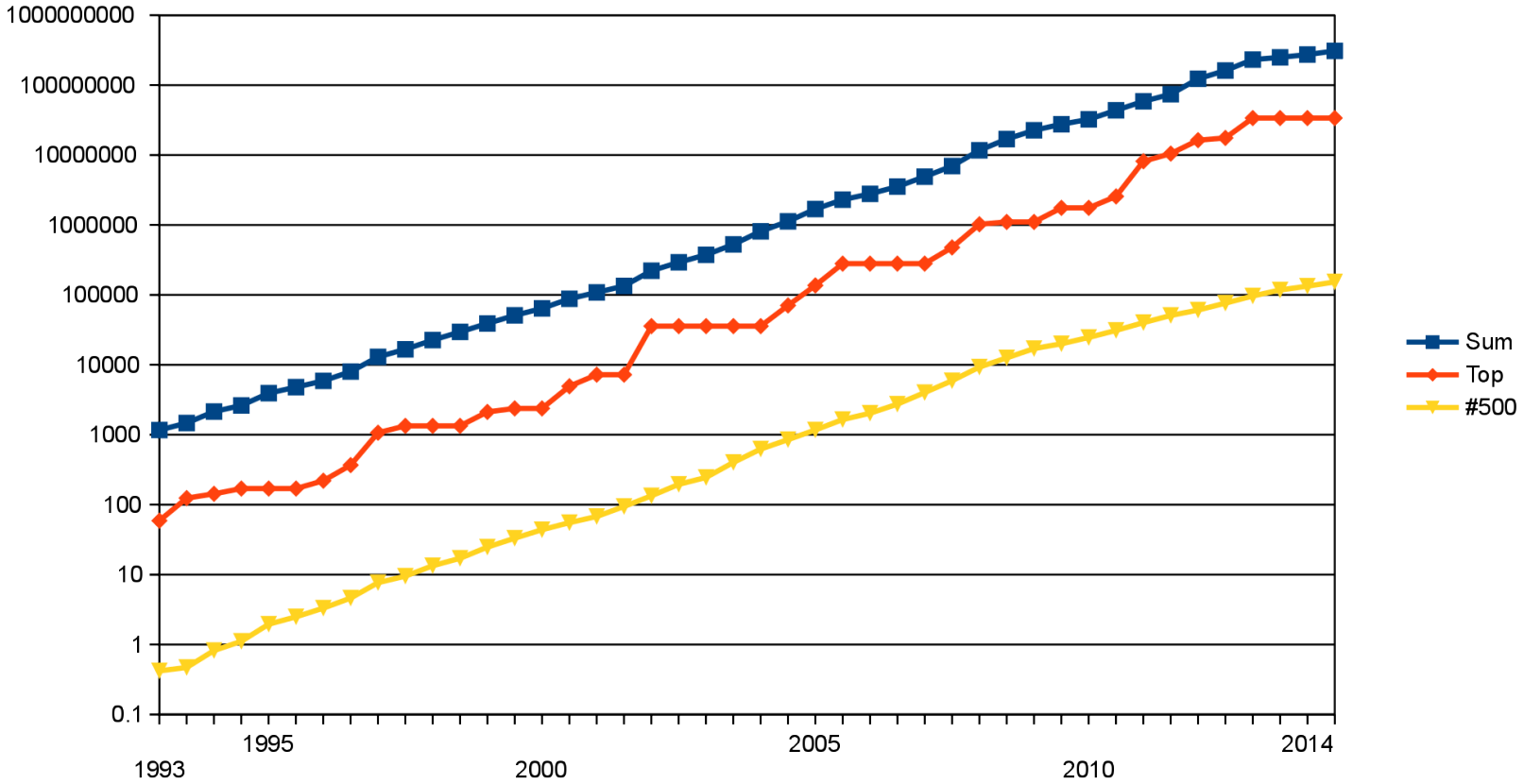
TORUS workshop

Ferrara, June 6th 2016

Outline

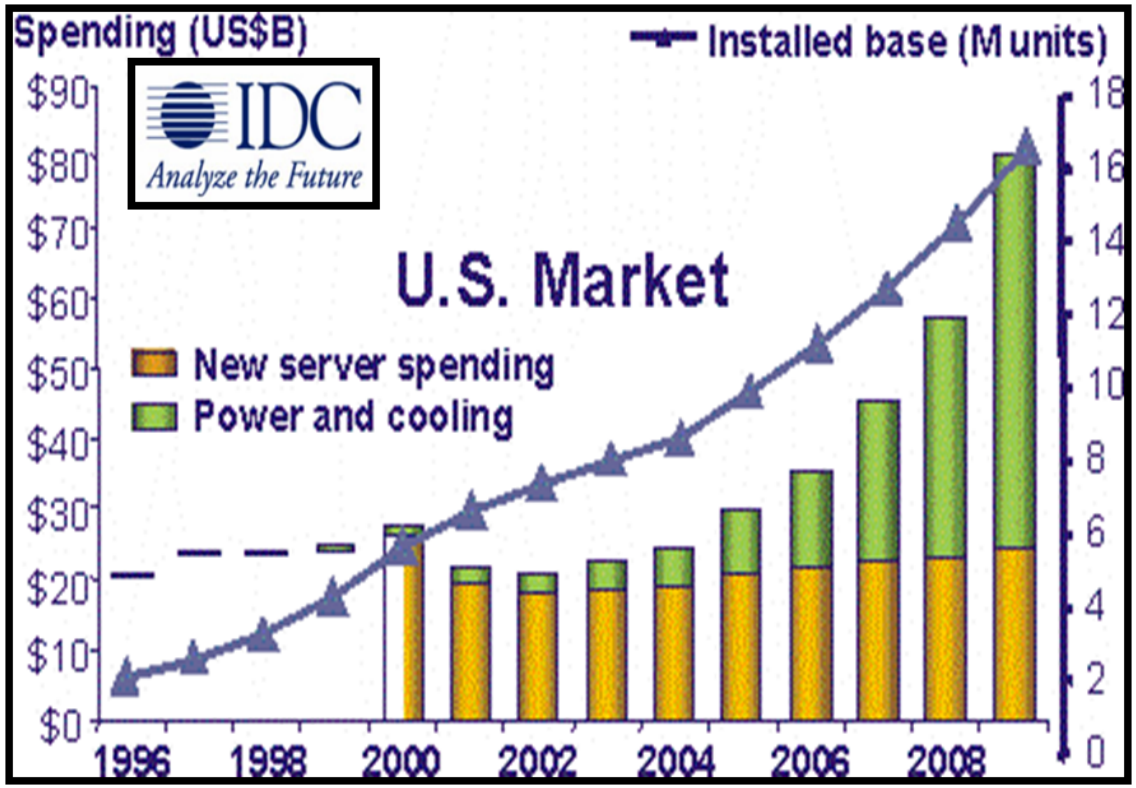
Outline

High Performance Computing is about performance



Constant increase in Performance (GFLOP/s) over time in the TOP 500

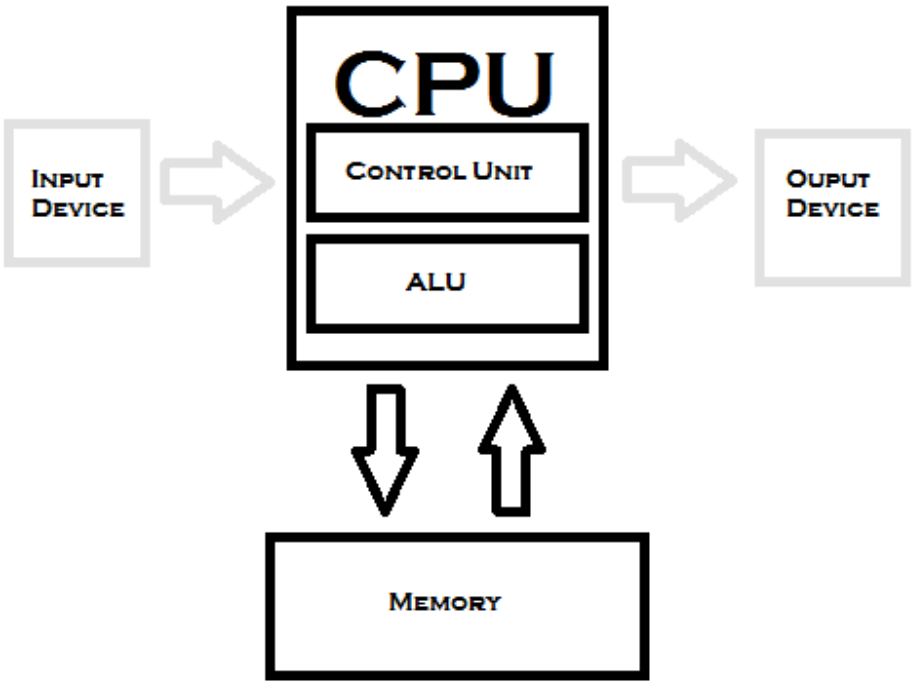
But energy is becoming more and more important



HPC facilities may start to account for consumed energy instead of running time

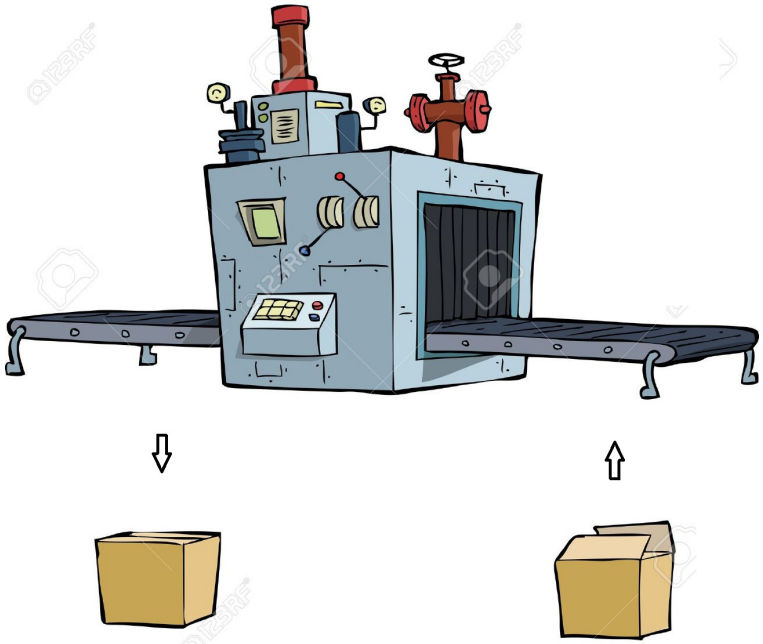
The von Neumann architecture

FLOP/s



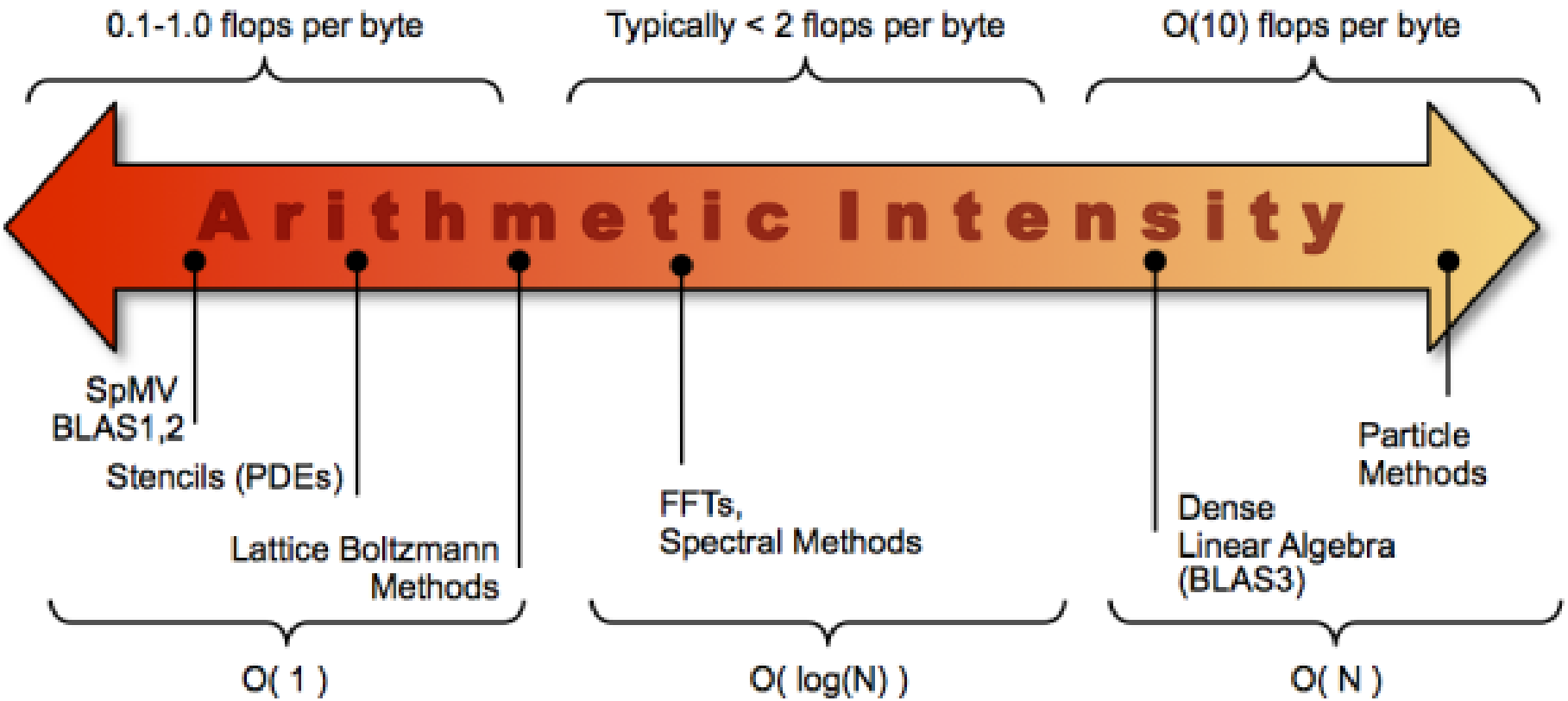
Byte/s

Manufacturing/s

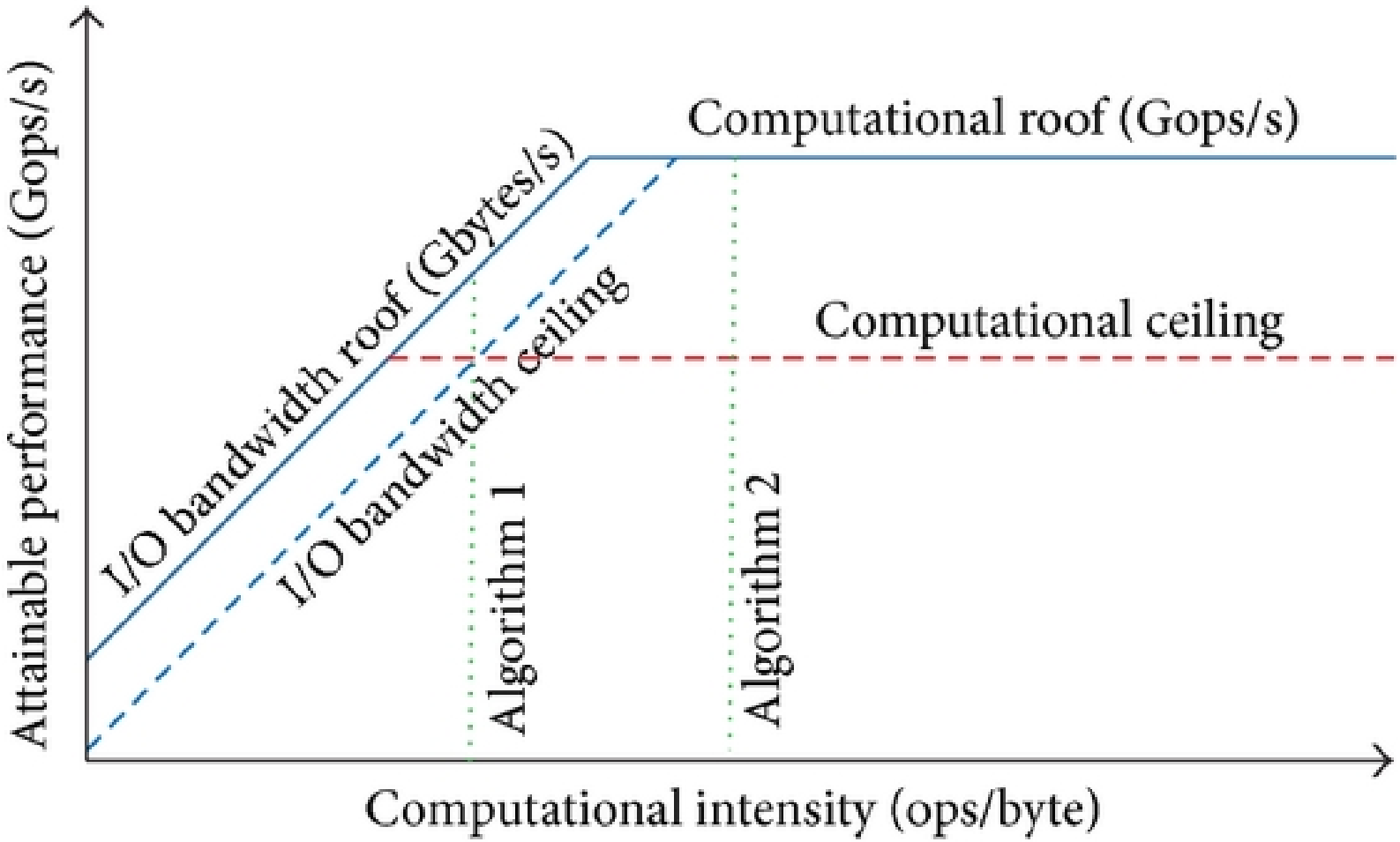


Load&Store/s

Arithmetic/computational Intensity



The Roofline Model



Williams, S., Waterman, A., & Patterson, D. (2009) "Roofline: an insightful visual performance model for multicore architectures" *Communications of the ACM*, 52(4), 65-76.

Towards energy optimizations...

To Optimize a code in HPC means:

- reduce the number of operations to the ones strictly needed
- try to match the operational intensity of the code to the capabilities of the given hardware
- and others details...

But the match is not always perfect... (memory or compute bound)

...can we change the hardware to match the software instead?

Yes, but we can only make it slower $\Rightarrow$ lowering the clock frequency

Towards energy optimizations...

To Optimize a code in HPC means:

- reduce the number of operations to the ones strictly needed
- try to match the operational intensity of the code to the capabilities of the given hardware
- and others details...

But the match is not always perfect... (memory or compute bound)

...can we change the hardware to match the software instead?

Yes, but we can only make it slower $\Rightarrow$ lowering the clock frequency

Towards energy optimizations...

To Optimize a code in HPC means:

- reduce the number of operations to the ones strictly needed
- try to match the operational intensity of the code to the capabilities of the given hardware
- and others details...

But the match is not always perfect... (memory or compute bound)

...can we change the hardware to match the software instead?

Yes, but we can only make it slower $\Rightarrow$ lowering the clock frequency

Towards energy optimizations...

To Optimize a code in HPC means:

- reduce the number of operations to the ones strictly needed
- try to match the operational intensity of the code to the capabilities of the given hardware
- and others details...

But the match is not always perfect... (memory or compute bound)

...can we change the hardware to match the software instead?

Yes, but we can only make it slower $\Rightarrow$ lowering the clock frequency

Towards energy optimizations...

To Optimize a code in HPC means:

- reduce the number of operations to the ones strictly needed
- try to match the operational intensity of the code to the capabilities of the given hardware
- and others details...

But the match is not always perfect... (memory or compute bound)

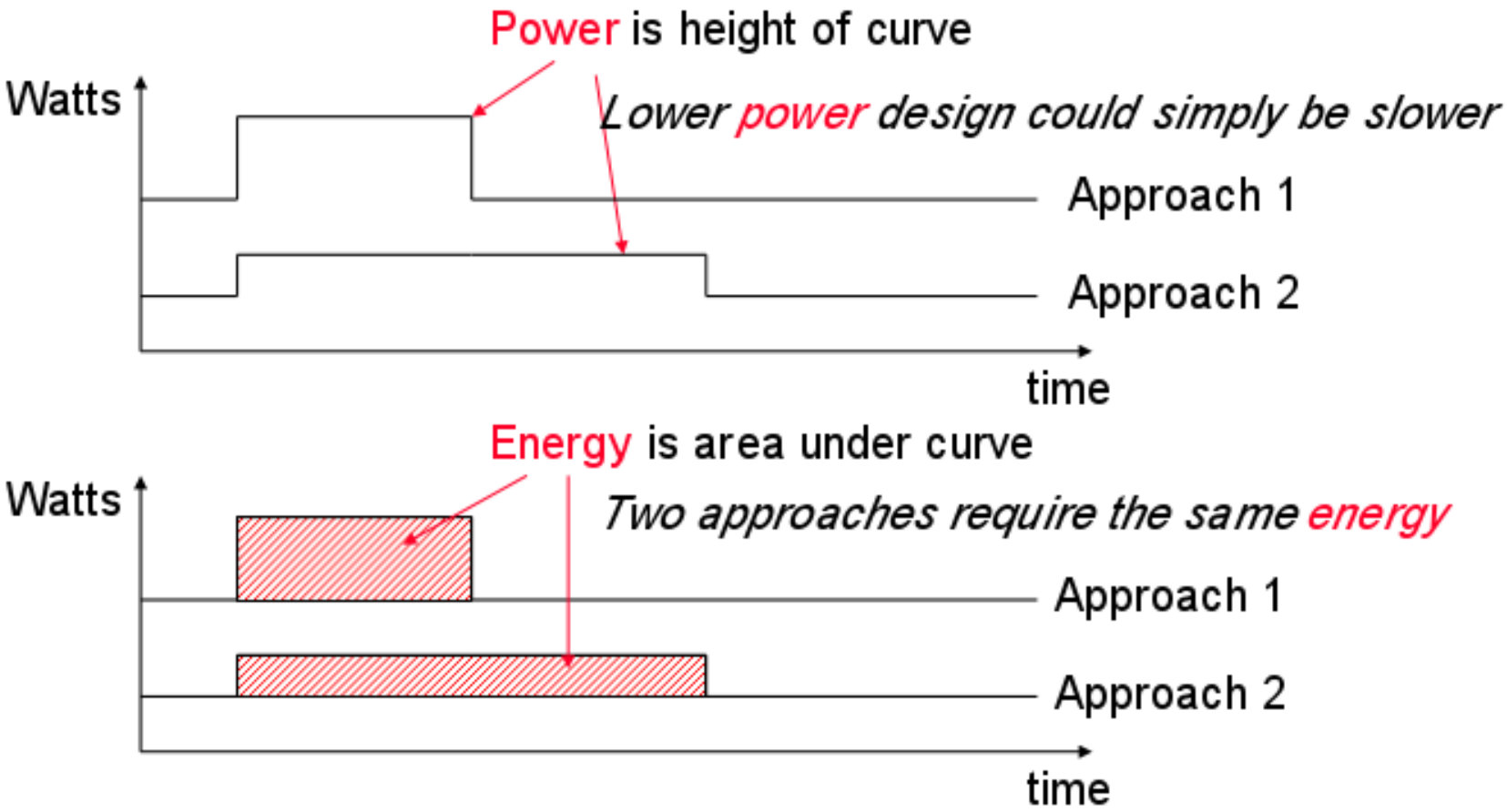
...can we change the hardware to match the software instead?

Yes, but we can only make it slower $\Rightarrow$ lowering the clock frequency

Outline

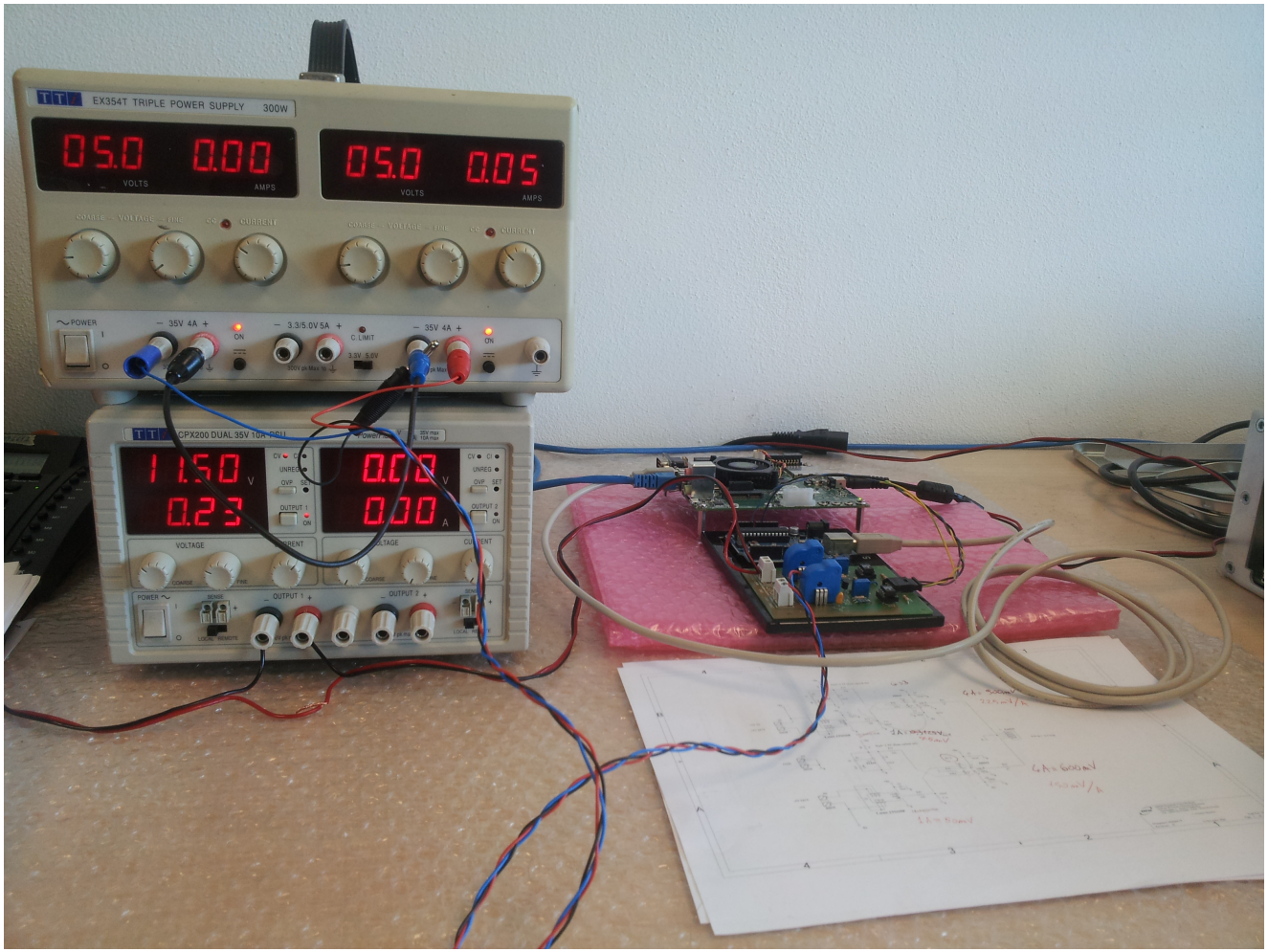
Power vs Energy

$$E_s = T_s \times P_{avg} \qquad P_{avg} = I_{avg} \times V$$



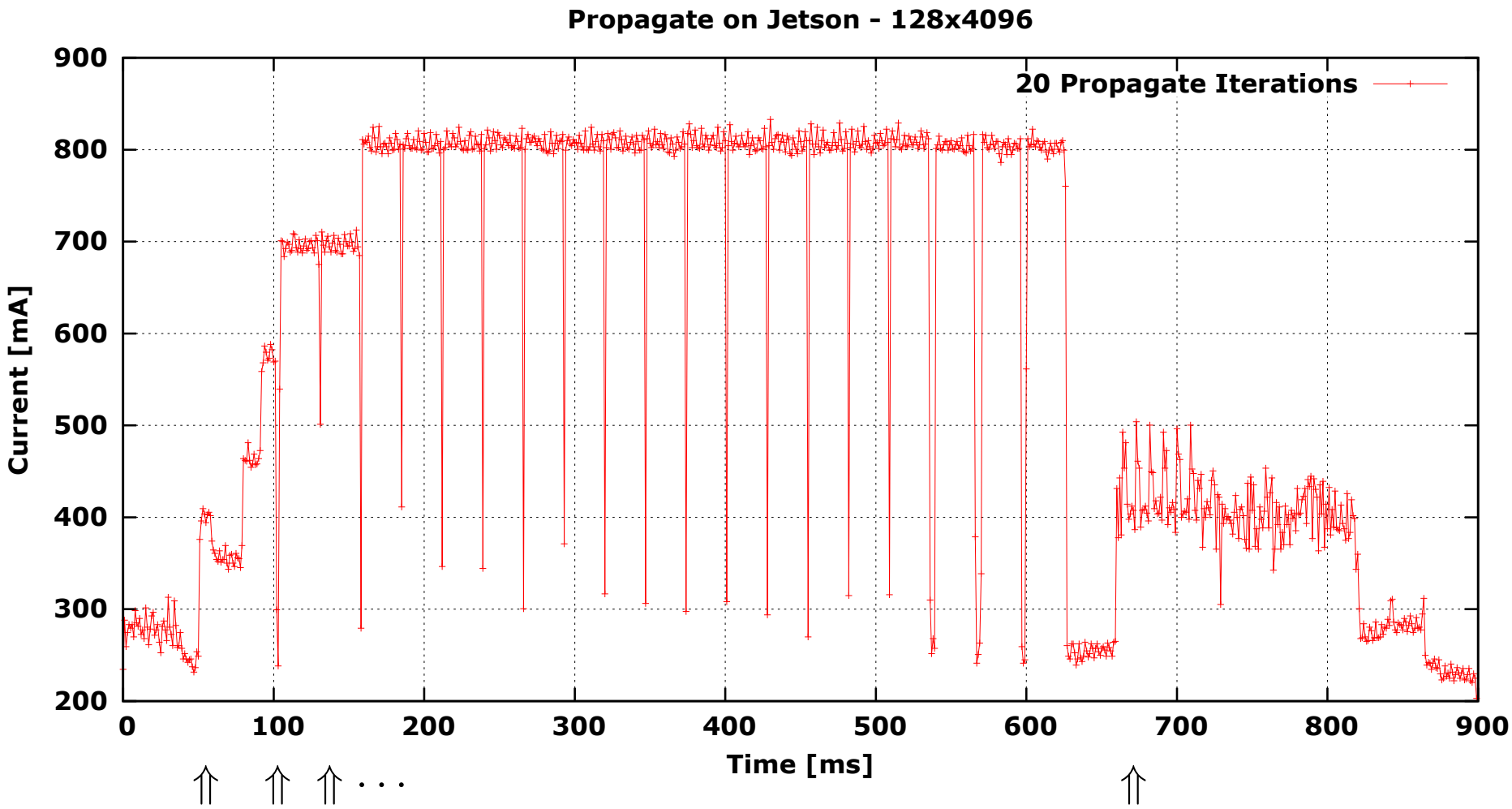
Setup to sample instantaneous current absorption

One current to voltage converter...
...plus an Arduino UNO (microcontroller + 10-bit ADC + Serial over USB)



Current to Voltage + Digitization with Arduino + USB Serial

Acquired data example with default frequency scaling



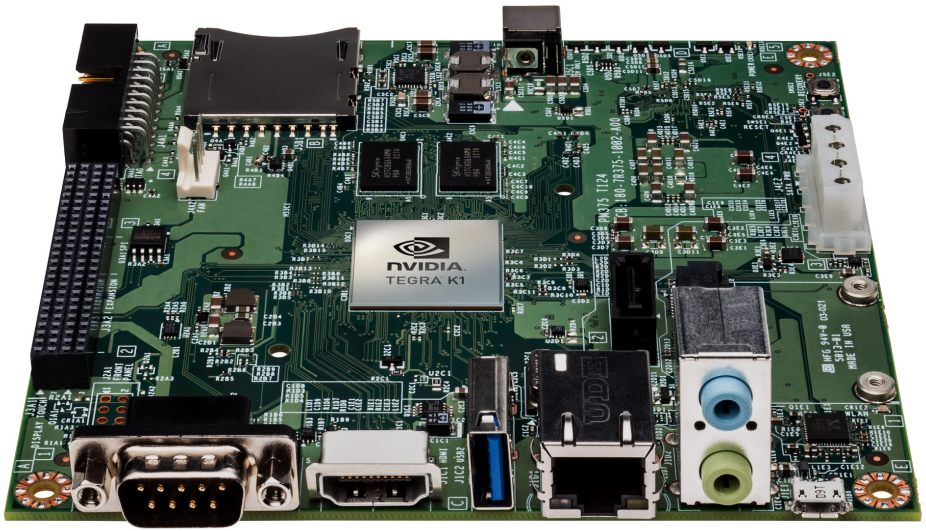
Iterations can be counted

This is a D2H transfer

Outline

Outline

NVIDIA Jetson TK1



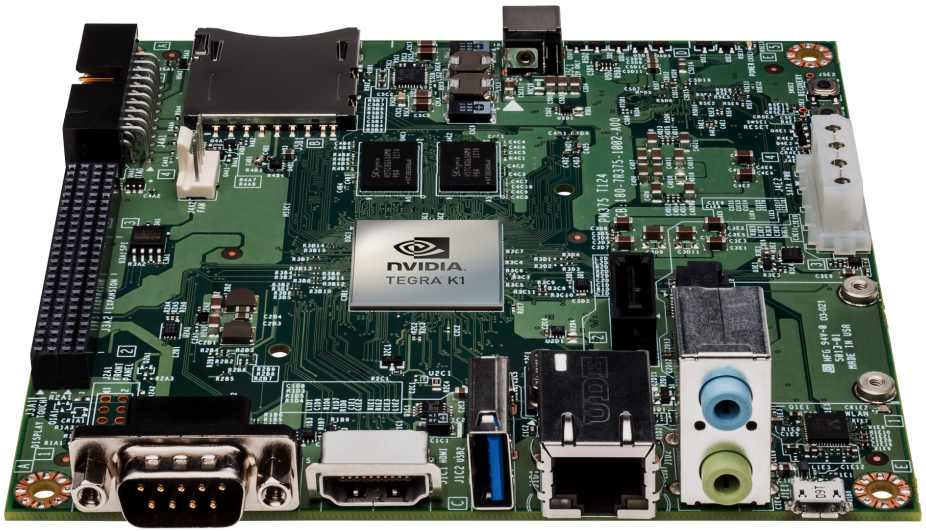
SoC: Tegra K1

- CPU: NVIDIA "4-Plus-1" 2.32GHz ARM quad-core Cortex-A15, with battery-saving shadow-core
- GPU: NVIDIA Kepler "GK20a" GPU with 192 SM3.2 CUDA cores

Awarded for the Best Paper

7th Workshop on UnConventional High Performance Computing (UCHPC), Porto 2014

NVIDIA Jetson TK1



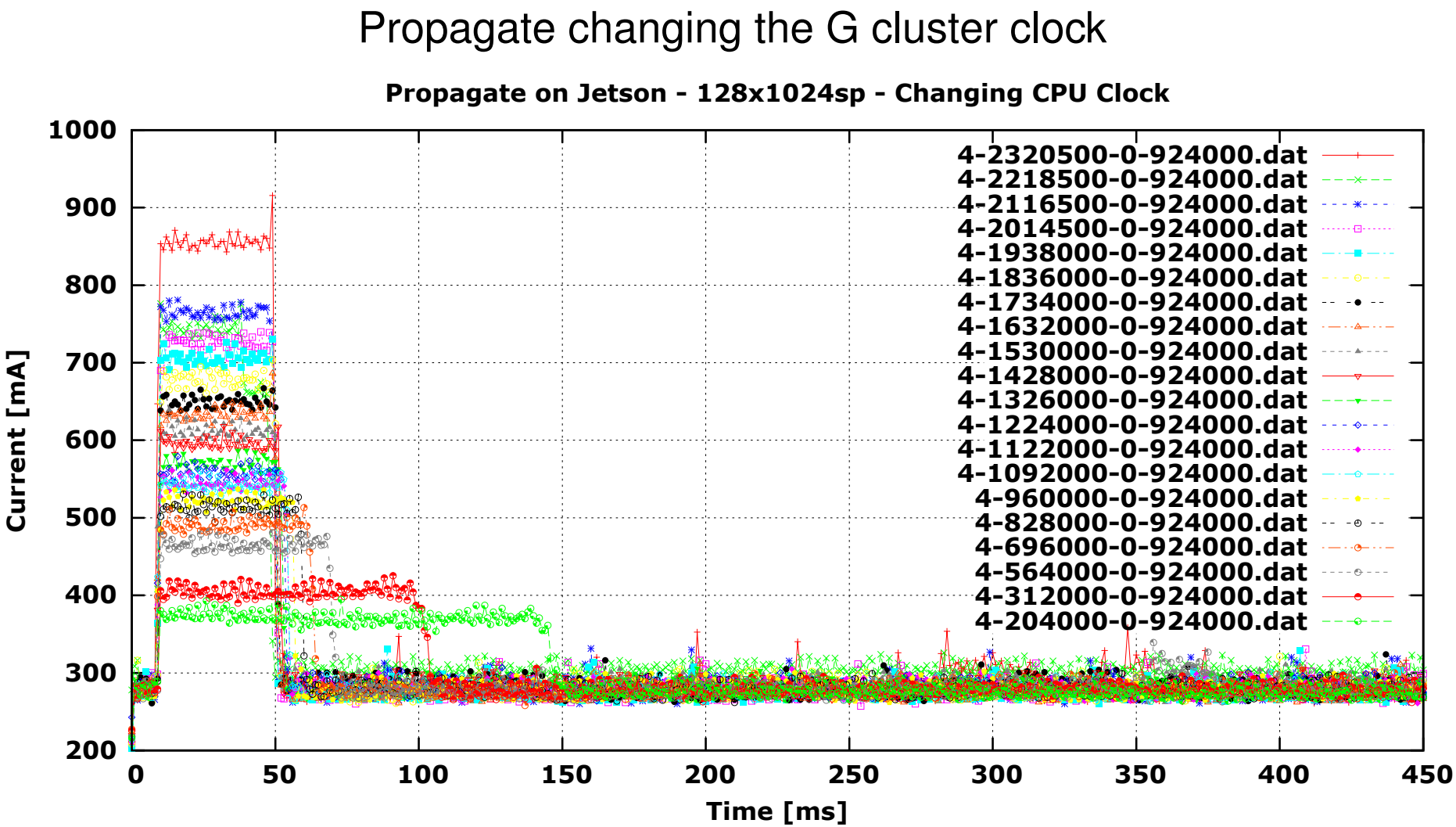
SoC: Tegra K1

- CPU: NVIDIA "4-Plus-1" 2.32GHz ARM quad-core Cortex-A15, with battery-saving shadow-core
- GPU: NVIDIA Kepler "GK20a" GPU with 192 SM3.2 CUDA cores

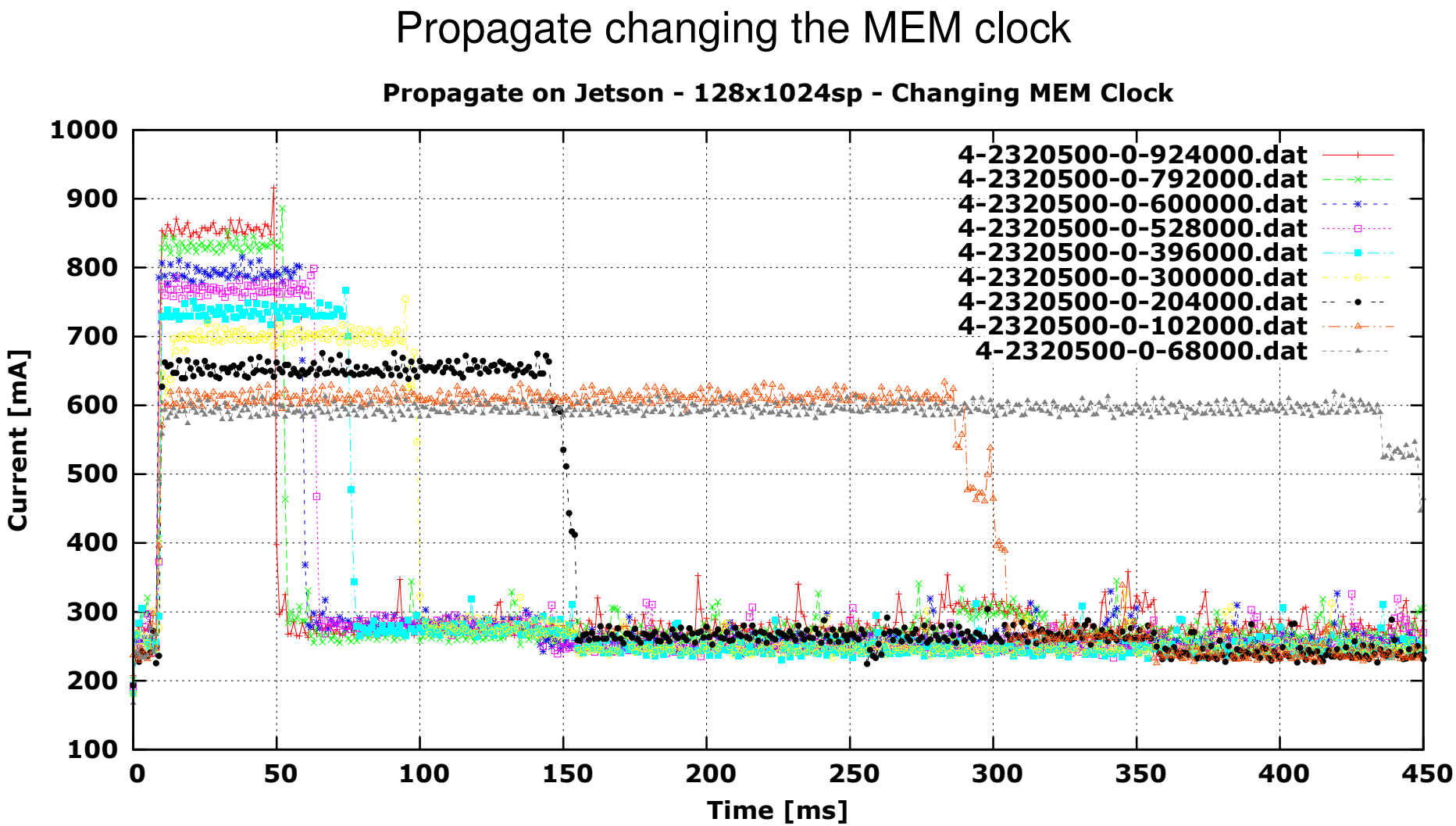
Awarded for the Best Paper

7th Workshop on UnConventional High Performance Computing (UCHPC), Porto 2014

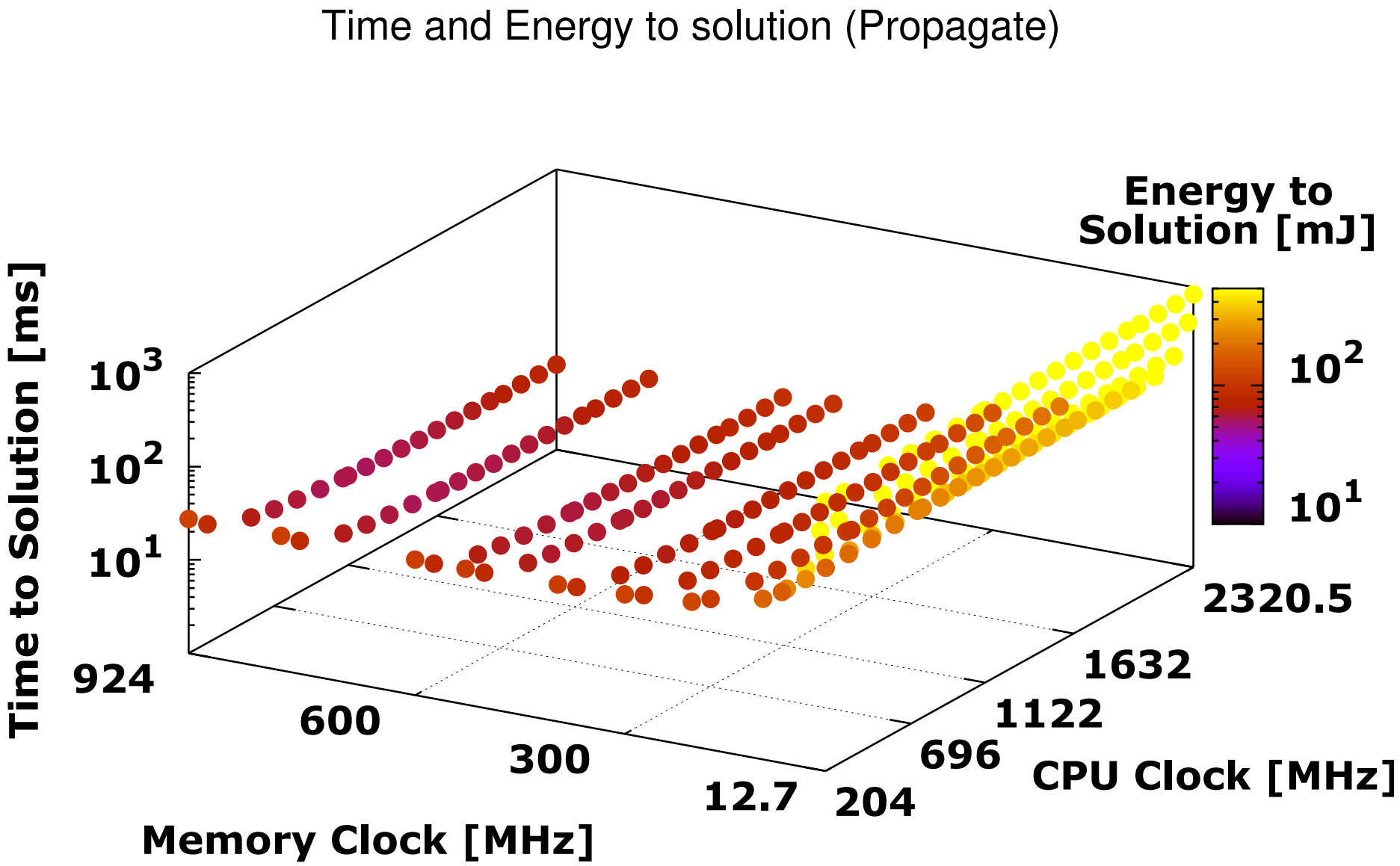
C with NEON intrinsics, on the Cortex A15



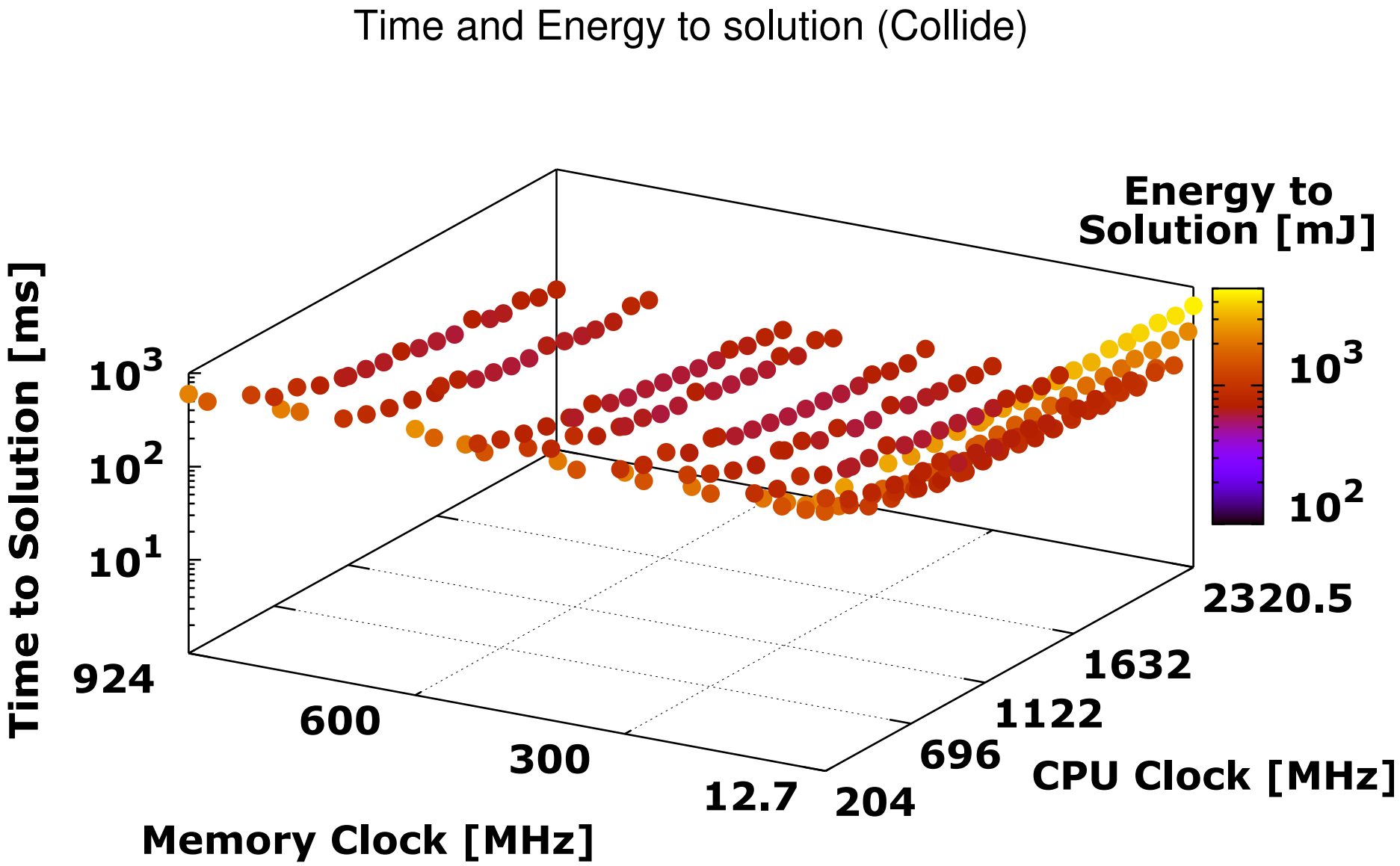
C with NEON intrinsics, on the Cortex A15



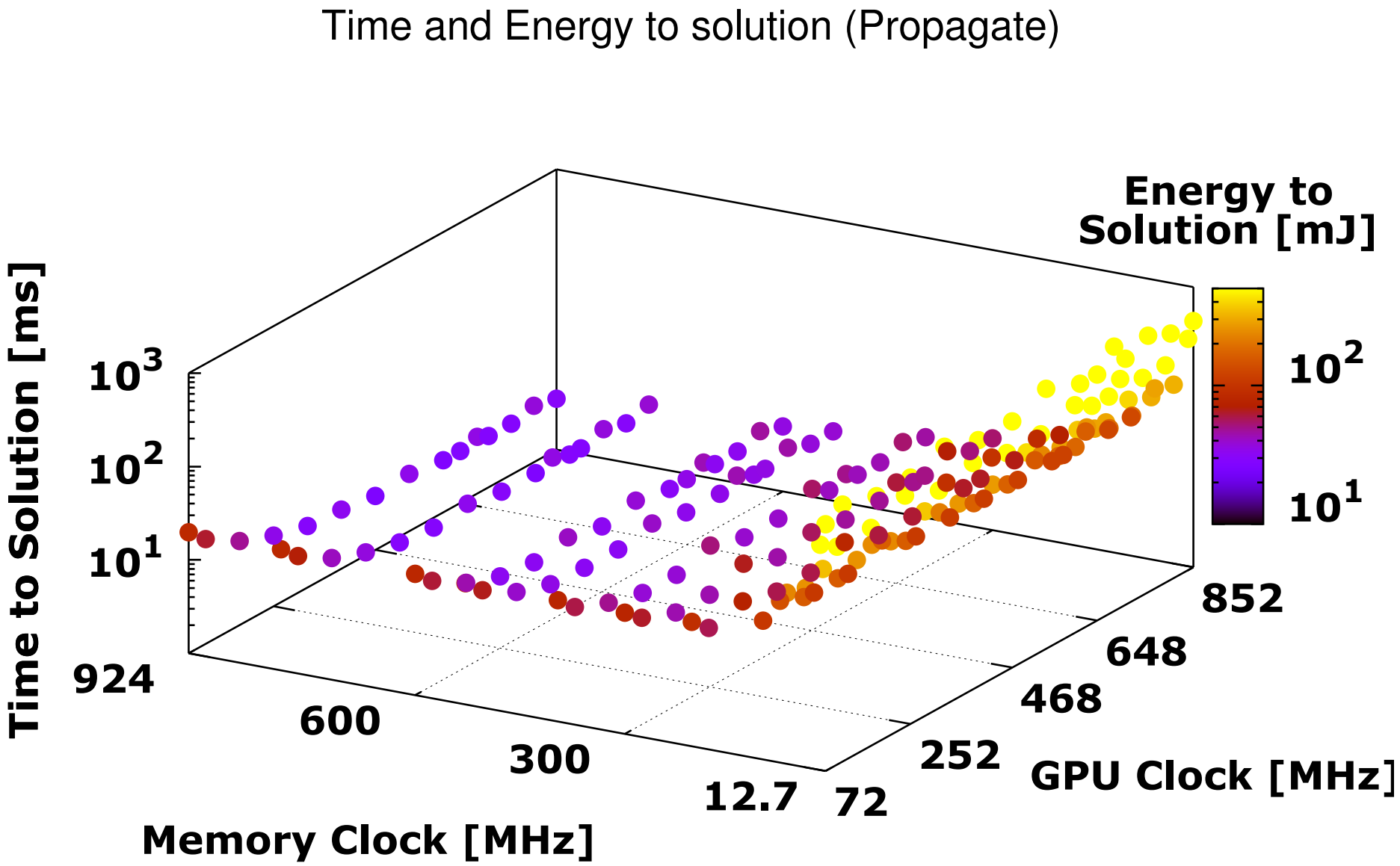
C with NEON intrinsics, on the Cortex A15



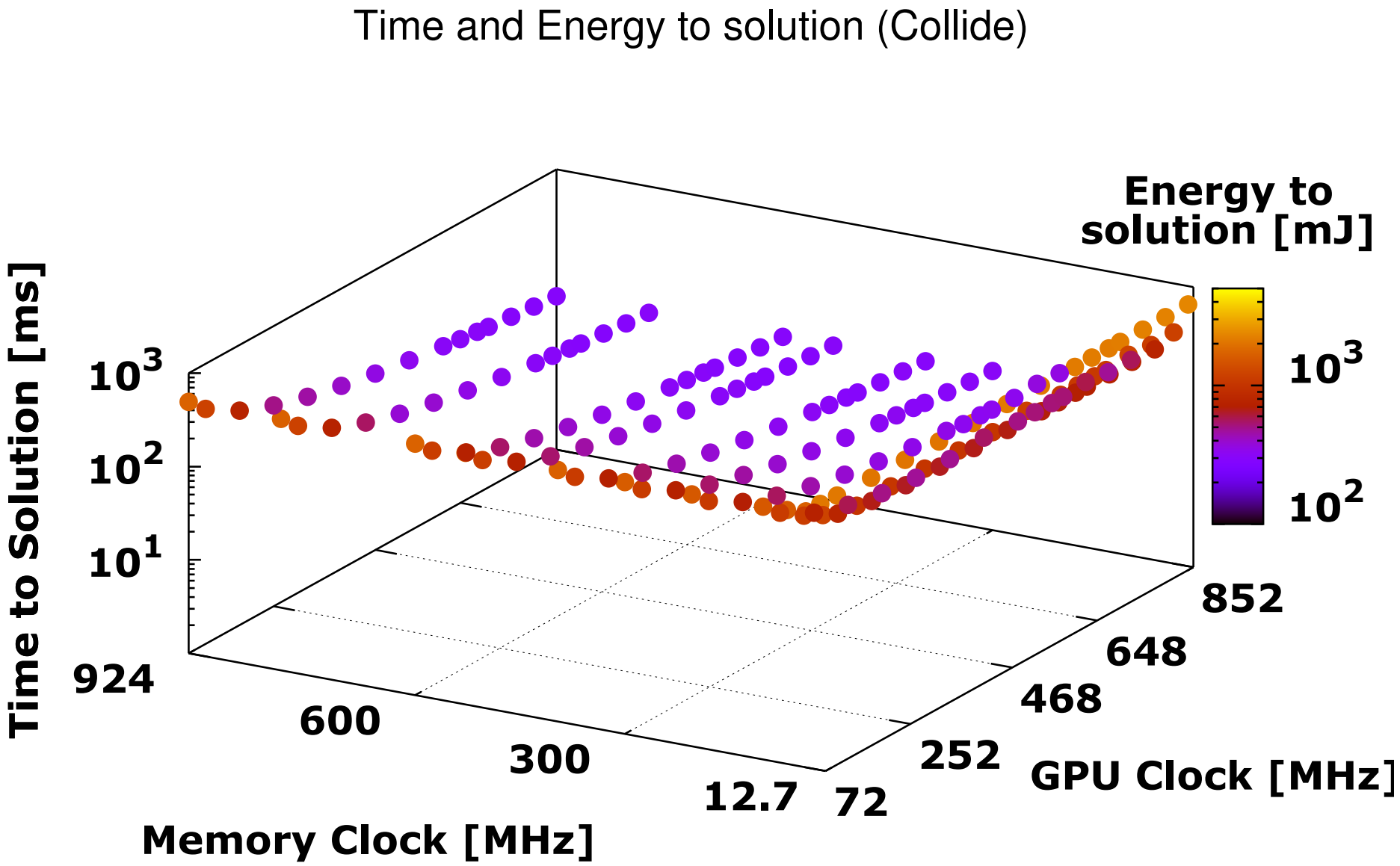
C with NEON intrinsics, on the Cortex A15



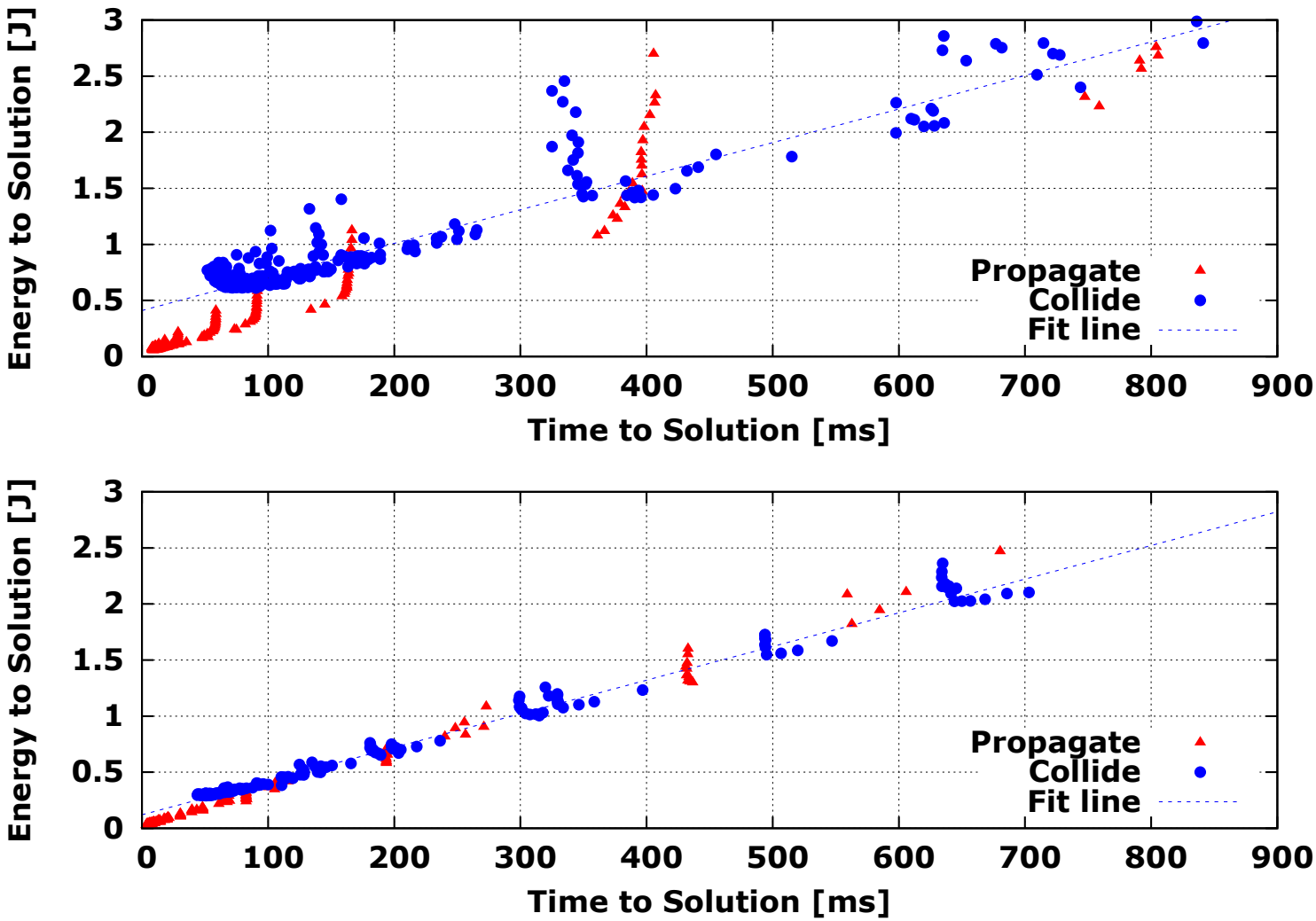
CUDA on the GK20A



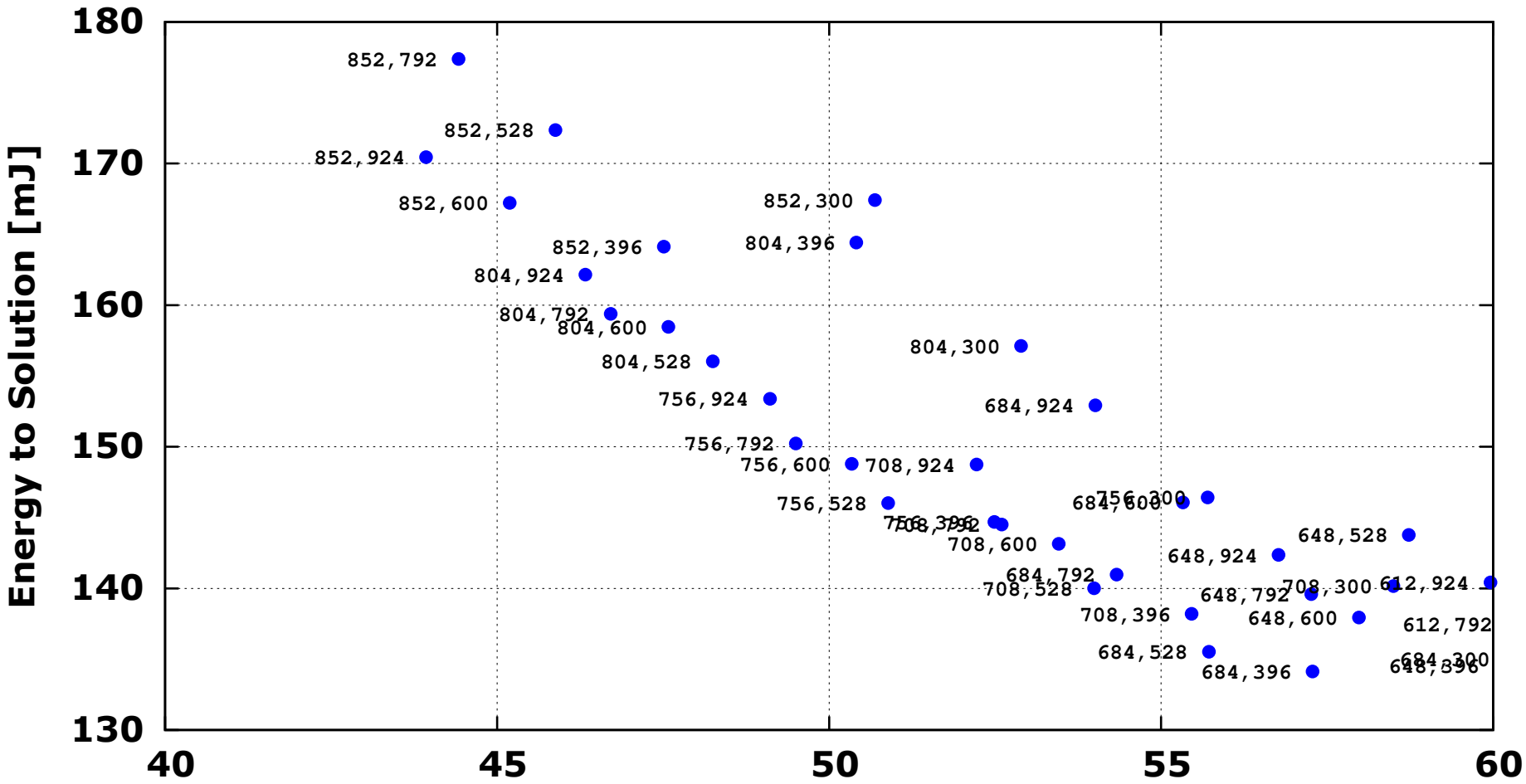
CUDA on the GK20A



Energy to Sol. vs Time to Sol. CPU(top), GPU(bottom)



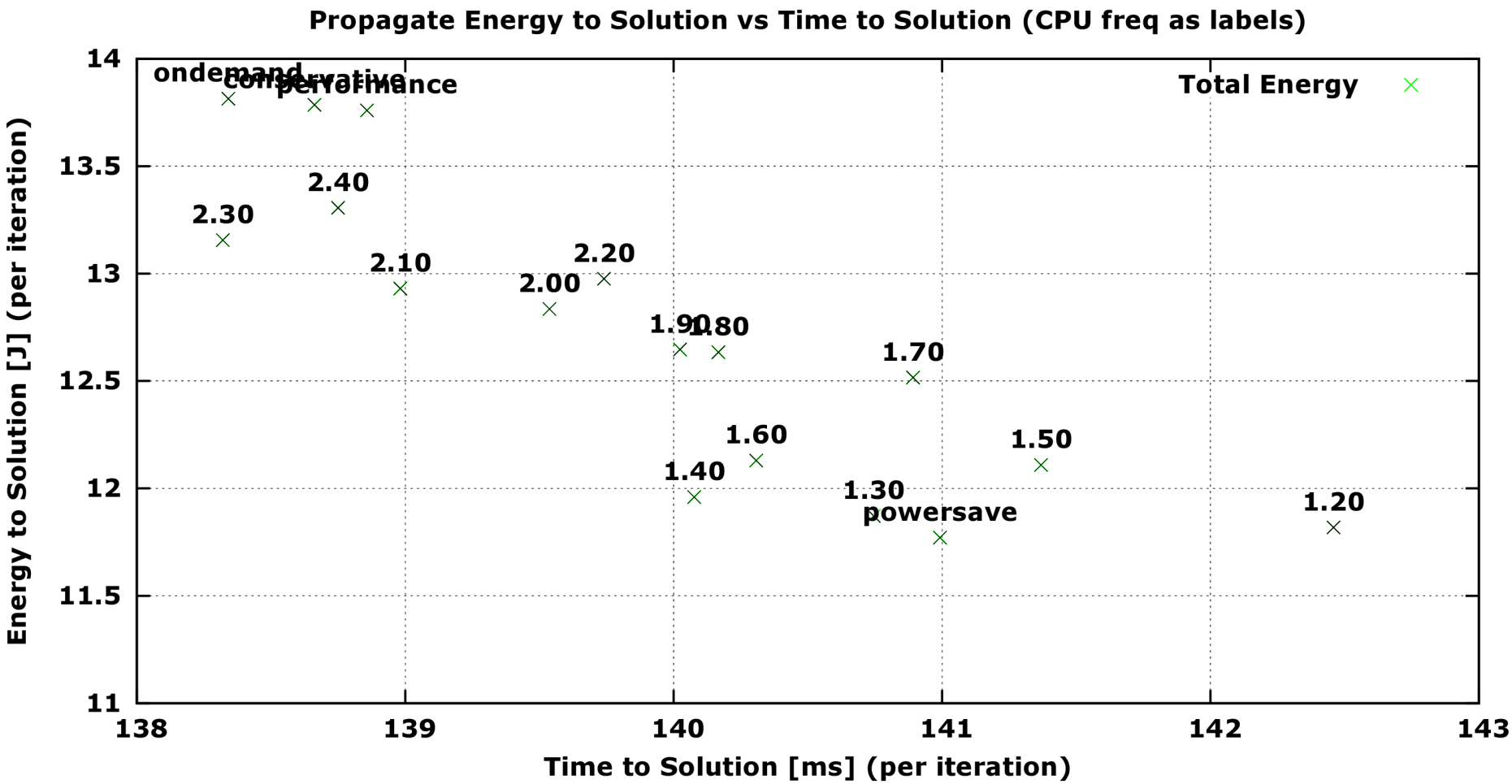
Energy to Solution vs Time to Solution (GPU GK20A) zoom



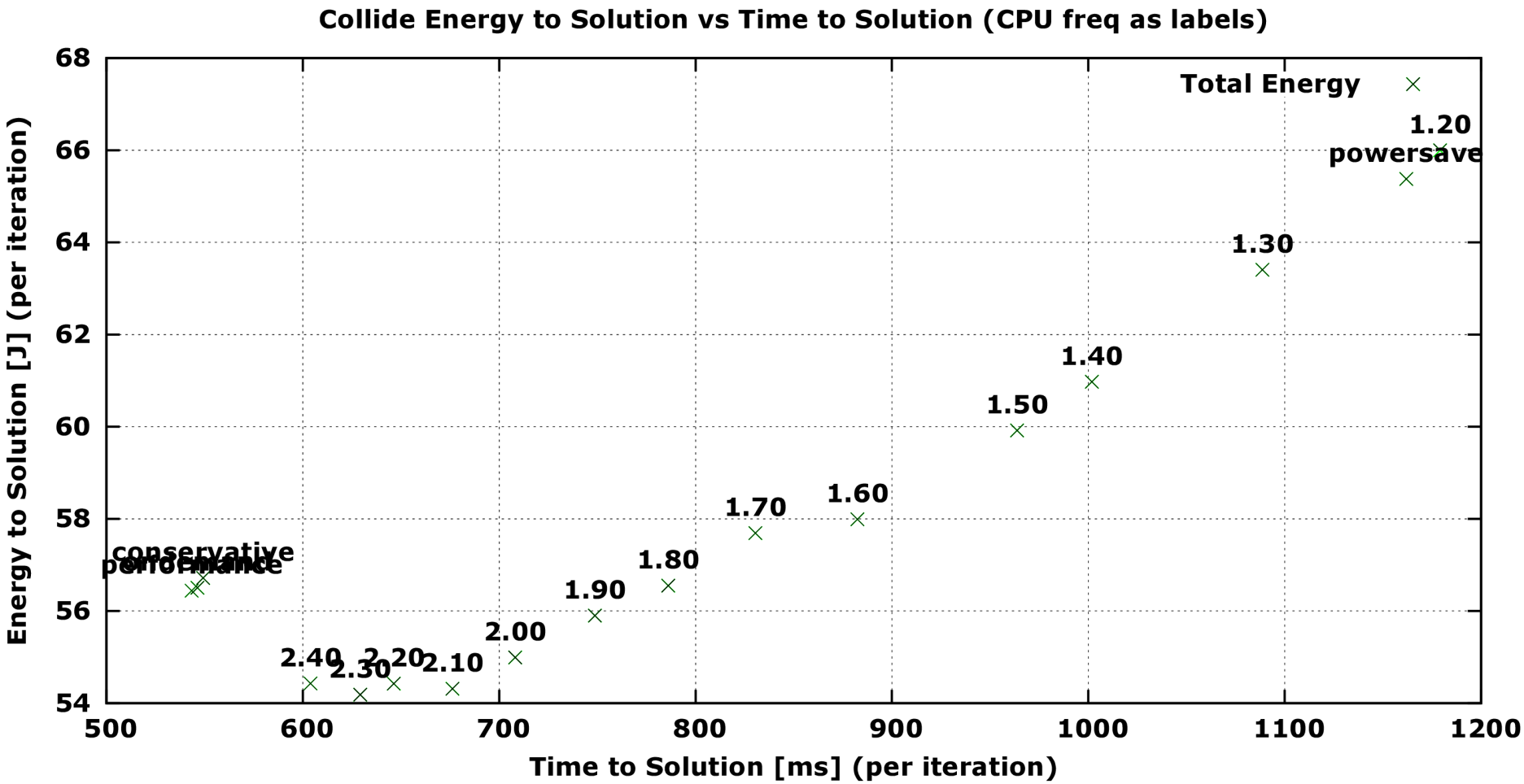
Outline

Outline

Energy/Time to Solution Propagate DP

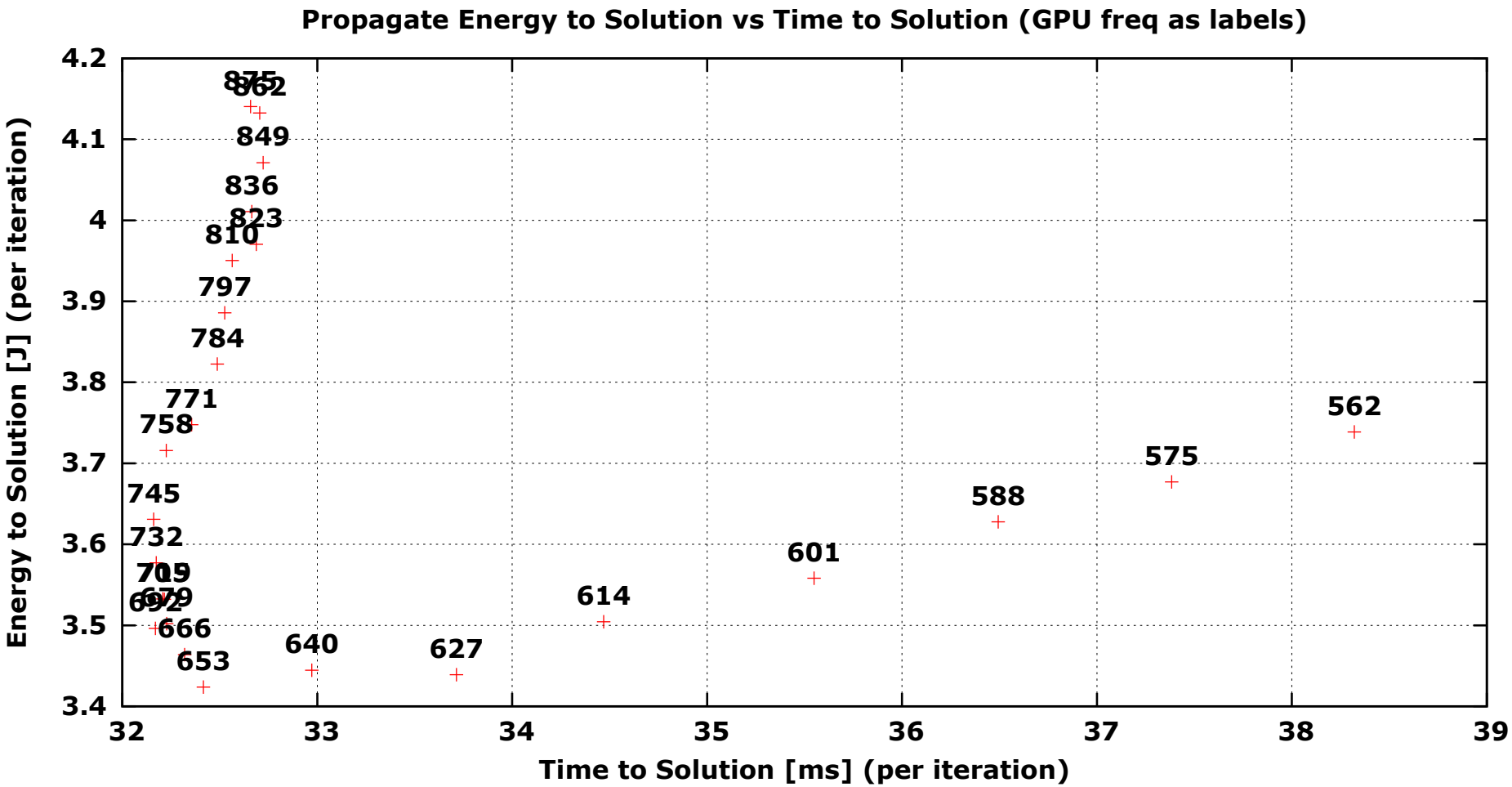


Energy/Time to Solution Collide DP

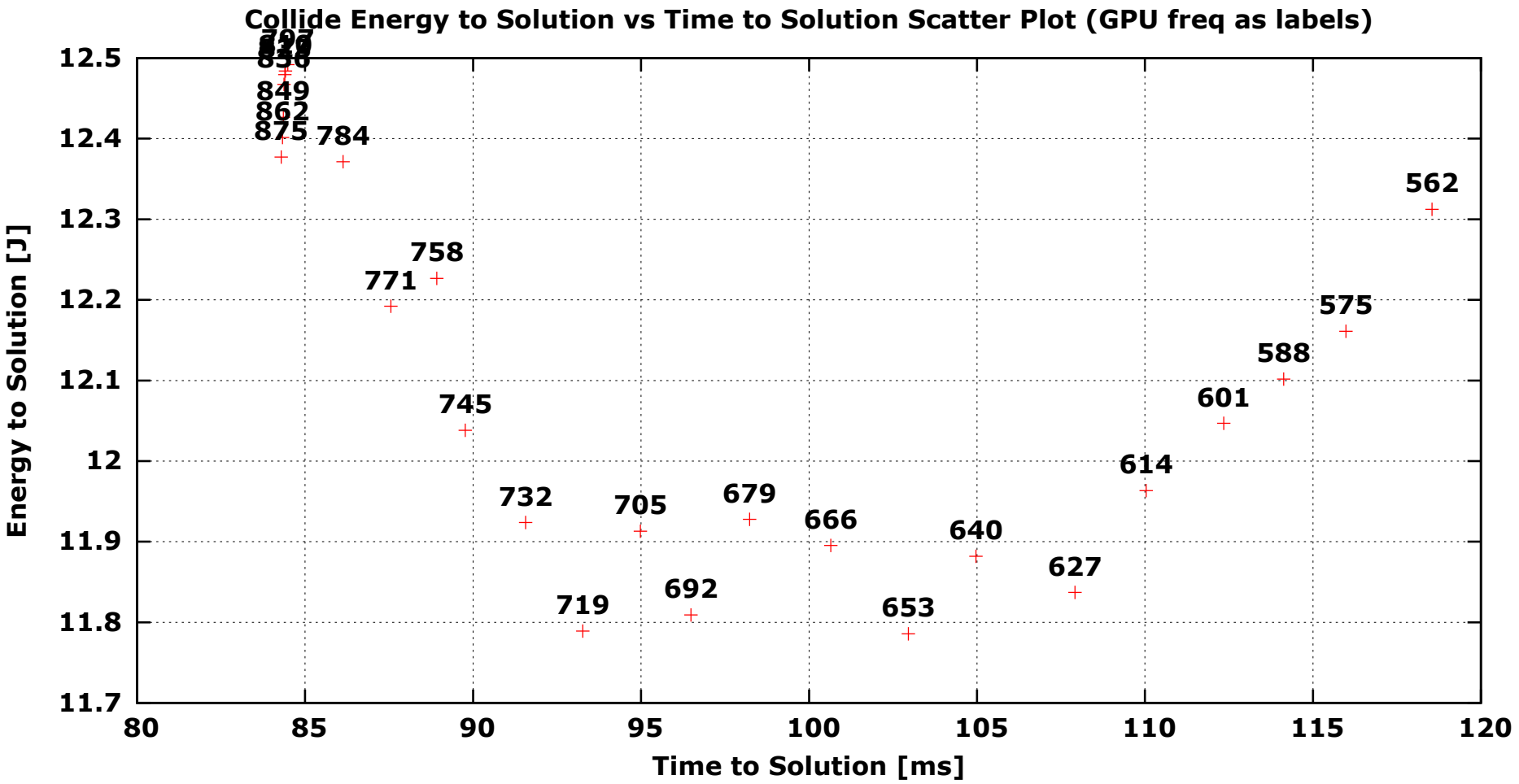


Outline

Energy/Time to Solution Propagate DP



Energy/Time to Solution Collide DP



Outline

Conclusions

- limited but not negligible power optimization is possible by adjusting clocks on a function-by-function basis (between $\approx 5 \cdots 20\%$).
- baseline power consumption is relevant ($\approx 30\%$)
- any performance optimization from the software point of view lead to an energy saving
- options to run the processor at very low frequencies seem almost useless (for HPC applications)

Computing facilities may start to let you pay the consumed energy...
...so its better to start to optimize your code

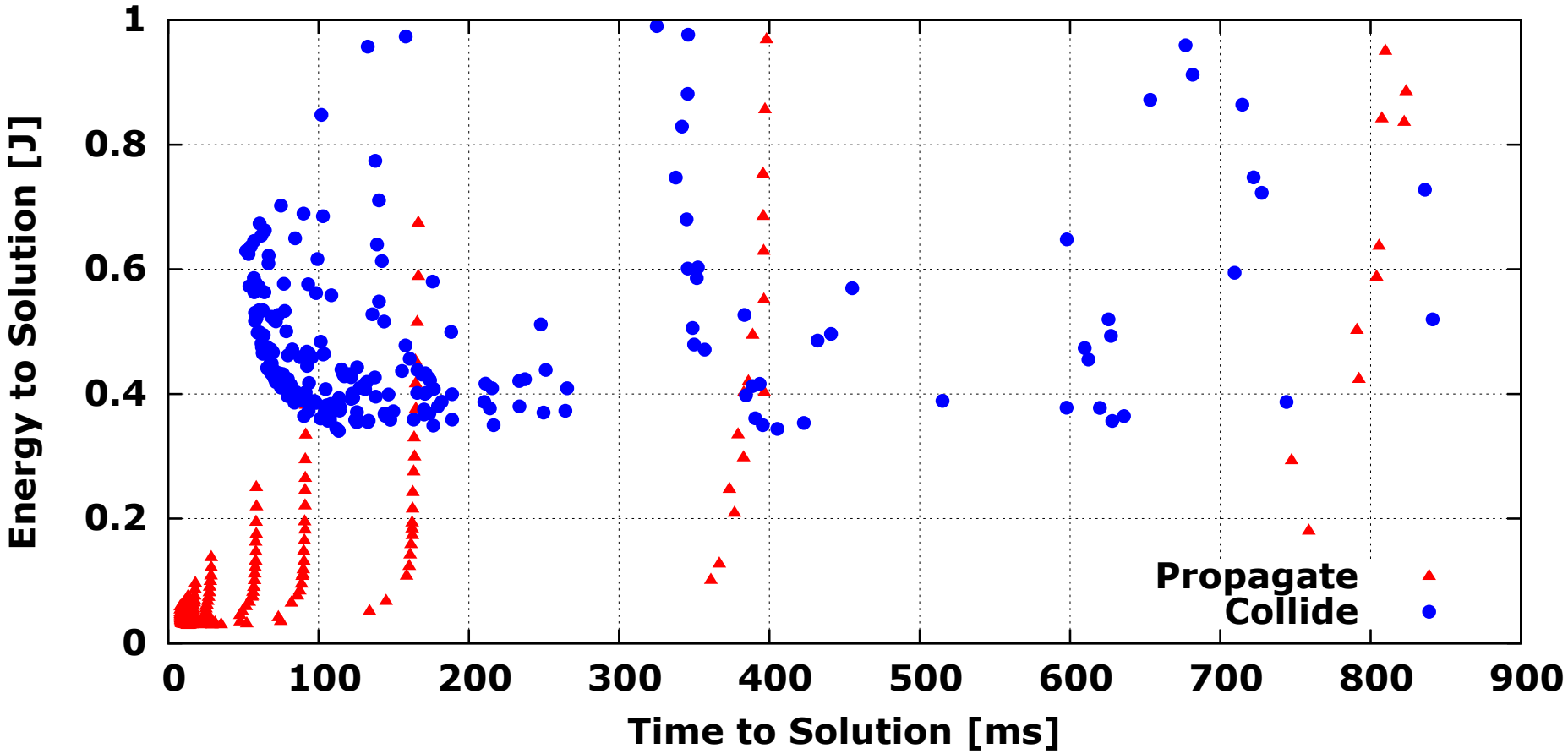
Conclusions

- limited but not negligible power optimization is possible by adjusting clocks on a function-by-function basis (between $\approx 5 \cdots 20\%$).
- baseline power consumption is relevant ($\approx 30\%$)
- any performance optimization from the software point of view lead to an energy saving
- options to run the processor at very low frequencies seem almost useless (for HPC applications)

Computing facilities may start to let you pay the consumed energy...
...so its better to start to optimize your code

Thanks for Your attention

Energy to Solution vs Time to Solution (CPU A15)



Energy to Solution vs Time to Solution (GPU GK20A)