Article Presentation:

Huang, T., Zhu, Y., Qiu, M. et al., “Extending Amdahl’s Law and Gustafson’s Law by Evaluating Interconnections on Multi-Core Processors,” *Journal of Supercomputing* (2013) 66: 305. https://doi.org/10.1007/s11227-013-0908-9

CPSC 5260 Parallel Algorithms
Jonathan Land

Structure of Presentation

I. Previous studies that influenced this article’s content

II. Main problem(s) that the authors are answering

III. Proposed solutions to the problem(s)

IV. Research suggestions

I. Previous studies that influenced this article’s content

I. Previous studies that influenced this article's content

A. Hill MD, Marty MR (2008), "Amdahl's Law in the Multicore Era," Computer 41(7): 33–38.

- This study considered performance issues within multi-core architecture. Hill and Marty presented a cost model for the chip area ($A(m,r) = m \times r$) to show the limitations of speedup when parallelizing.
  - **Main issue: Overlooked inter-core communication**

B. Woo DH, Lee HHS (2008), "Extending Amdahl's Law for Energy-Efficient Computing in the Multi-core Era. Computer 41(12): 24–31.

- This study attempted to extend Amdahl's law by including energy models into Amdahl's formula/model.

I. Previous studies that influenced this article’s content (contd.)

C. Sun X-H, Chen Y (2010), “Reevaluating Amdahl’s Law in the Multicore Era,” *Journal of Parallel and Distributed Computing* 70(2): 183–188

- This study sought to extend Amdahl’s and Gustafson’s laws.
- Argued that these laws could be interpreted correctly if users would increase computing demands when given more computing power.

II. Main problem(s) that the authors are answering

II. Main problem(s) that the authors are answering

A. Amdahl’s and Gustafson’s laws can be easily misinterpreted and can produce imprecise results.

   Primary reason: Their laws do not adequately account for *inter-core communication* (interconnects) and the overhead this causes in their cost models.

   When these laws were created, “all processors contained single core with no room for concern over on-chip communications between multiple cores.”

   Amdahl: overly-pessimistic

   Gustafson: overly-optimistic

   Hill and Marty: overlook interconnects and inter-core communication

B. Both laws tend to oversimplify the computational complexities within parallelization.

C. So, the article qualifies Amdahl’s and Gustafson’s laws a bit more to consider the importance of interconnection when discussing/applying speedup formulas.

II. Main problem(s) that the authors are answering(contd.)

- What is the purpose of interconnects?
  - Interconnects are used to connect computer components
  - Can connect…
    - Processors and processors
    - Processors and memories (banks)
    - Processors and caches (banks)
    - Caches and caches
    - I/O devices

II. Main problem(s) that the authors are answering(contd.)

- Effects of interconnects
  - Effects how large a system one can build
  - Effects how easily more processors can be added
  - Effects performance and efficiency
    - How fast processors, caches, and memory communicate
    - How long the latencies are
    - How much energy is spent in communicating

III. Proposed solutions to the problem(s)

III. Proposed solutions to the problem(s)

- Extending Amdahl's Law: Amdahl considered the parallelism on a system speedup given a *fixed-size problem*. Speedup is defined as the sequential execution time over parallel execution, which is shown in the following equation:

$$S_A(f,m) = \frac{1}{(1-f)+\frac{f}{m}}$$

III. Proposed solutions to the problem(s) (contd.)

- The authors extend Amdahl’s law by introducing the parameter *i*, or the number of interconnects, in order to represent the number of links of a single node or core in Network on Chip (NOCs) on multi-core processors.

$$S_A(f,m,i) = \frac{1}{f_t^s + f_c^s + f_{\frac{c}{m}}^p + f_{\frac{t}{i}}^p}$$

III. Proposed solutions to the problem(s) (contd.)

- Extending Gustafson’s Law: Gustafson wanted to show that parallelization allows us to deal with larger computational problem sizes in the same amount of time.

- The scale of the problem size is bounded by the execution time.

SG = scaled workload within a fixed period of time / original workload within a fixed period of time

III. Proposed solutions to the problem(s) (contd.)

- Here, *w* and *w’* represent the original workload and the scaled-up workload.
- Meaning: For Gustafson, a computer with *m* cores can deal with a larger workload than a single-core computer in the same amount of time.

$$S_G(f,m) = \frac{w'}{w}$$

$$= (1-f)xw + \frac{fmw}{w}$$

$$= (1-f) + fm$$

III. Proposed solutions to the problem(s) (contd.)

- **Disagreement(s): Gustafson accounts for the *computational workload*, but does *not* account for how parallel computation also increases *communication overhead*.**

- The authors suggest that the following revision or extension of Gustafson’s law is necessary to show that if performance of *interconnection* stays the same, or does not grow with the number of cores, then the execution time will increase (i.e., there will be no speedup).

III. Proposed solutions to the problem(s) (contd.)

$$S_G(f,m,i) = \frac{workload(new)}{workload(original)} x\Delta$$

$$= \frac{f_c^s + f_c^p m}{f_c^s + f_c^p} x \frac{1}{f_c^s + f_c^p + {}_t^s + \frac{K(m) f_t^p}{i}}$$

- The inclusion of *Δ* is to correct the execution time of the original workload required by interconnection and the original processor.
- In the denominator we have the new workload execution time with *m* processors and *i* interconnects.
- By adding this factor, the authors have provided safeguards against the assumption that one can speed up a task simply by adding more cores.

Case Study

- The authors used an image processing application (CT – Computer Tomography) to evaluate their speedup model when compared the speedup models of Hill, Sun, and friends.

- Language used: CUDA

- Test data structure: task graph (DAG)

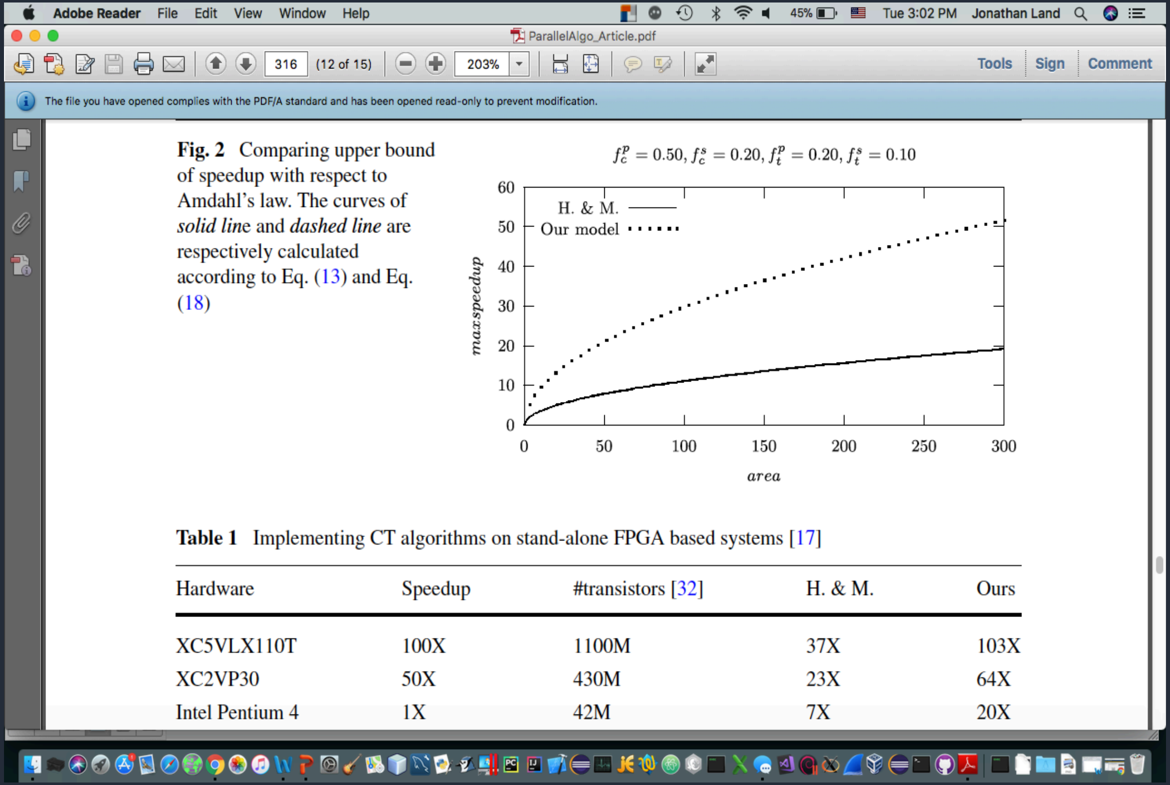
Case Study (contd.)

- Hardware used:

   1. XC5VLX110T: Field-Programmable Gate Array(s) or FPGA
   2. XC2VP130: FPGA
   3. Intel Pentium 4: CPU
   4. Quadro FX5600: GPU
   5. Tesla C1060: GPU

Case Study (contd.)

- Compares upper bound of speedup with Amdahl's law
- The solid line represents Hill & Marty's model
- The dotted line represents the viewpoint of the authors
- Authors argue that their model promises greater speedup
- They are more optimistic than Hill and Marty's study (Hill and Marty focused on *parallelism of computation*, not *workload of transmission*)
- Table 1 also shows that Hill and Marty underestimate the speedup since they ignored interconnections in their research.



The file you have opened complies with the PDF/A standard and has been opened read-only to prevent modification.

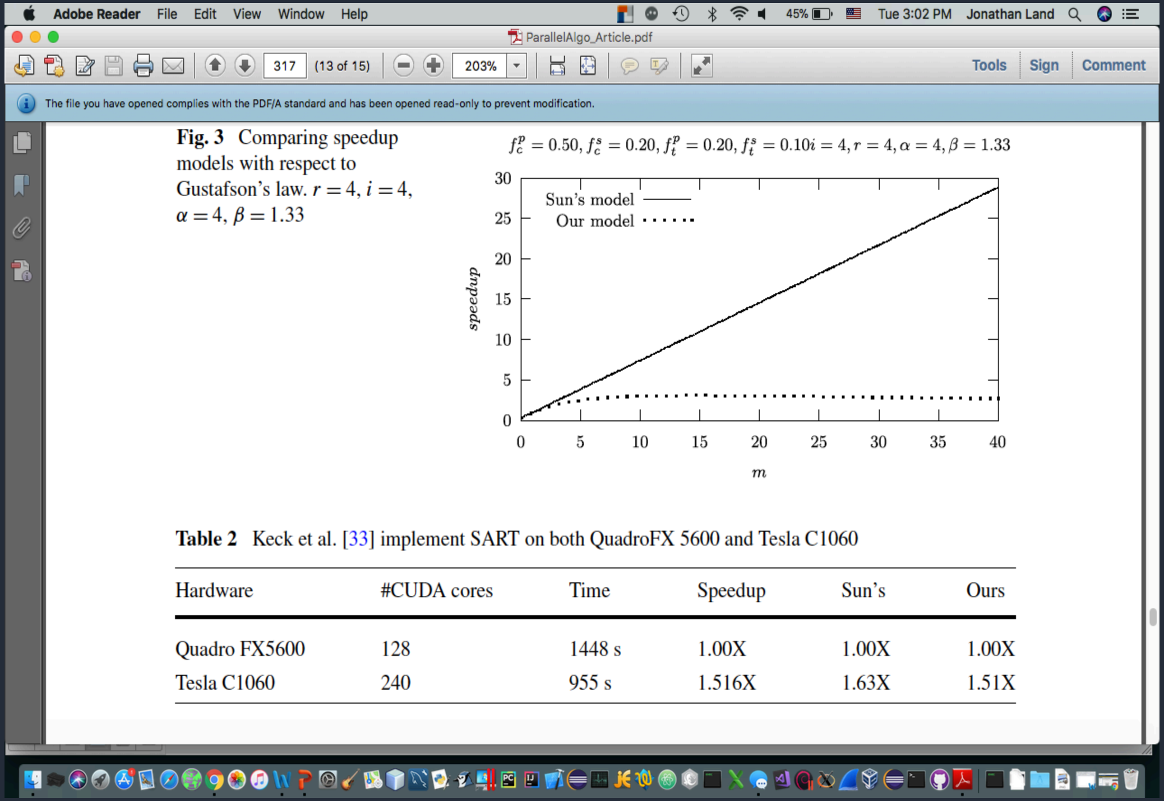
Fig. 2 Comparing upper bound of speedup with respect to Amdahl's law. The curves of *solid line* and *dashed line* are respectively calculated according to Eq. (13) and Eq. (18)

Table 1 Implementing CT algorithms on stand-alone FPGA based systems [17]

Hardware	Speedup	#transistors [32]	H. & M.	Ours
XC5VLX110T	100X	1100M	37X	103X
XC2VP30	50X	430M	23X	64X
Intel Pentium 4	1X	42M	7X	20X

Case Study (contd.)

- The dashed line represents the speedup of the author’s extensions
- Their model quickly reaches an upper bound, then decreases at a slow rate
- Compared to some studies (i.e., Sun’s), this is more pessimistic regarding speedup.
- So, a mediating cost model



The file you have opened complies with the PDF/A standard and has been opened read-only to prevent modification.

Fig. 3 Comparing speedup models with respect to Gustafson’s law. $r = 4$, $i = 4$, $\alpha = 4$, $\beta = 1.33$

Table 2 Keck et al. [33] implement SART on both QuadroFX 5600 and Tesla C1060

Hardware	#CUDA cores	Time	Speedup	Sun’s	Ours
Quadro FX5600	128	1448 s	1.00X	1.00X	1.00X
Tesla C1060	240	955 s	1.516X	1.63X	1.51X

IV. Research suggestions

IV. Research suggestions

- The author’s suggest that there should be continued research on hardware advances (multi-core chips) that reduce the interconnect bottleneck issue.

- They also suggest following these research guidelines when considering speedup:

  - Deal with interconnects at the *initial phase* of architectural designs
  - For a fixed-size problem with a given silicon area, the optimized number of cores and interconnects are respectively related to the parallelism of computation and transmission workload of a task.
  - The upper bound speedup of architecture is closely related to the task running on it and the silicon area it costs.
  - The area percentage spent on cores and interconnects is basically determined by the computation-to-transmission workload ratio of a task.

Summarizing this Summary

- Just because there is speedup theoretically, does not necessarily mean that there is speedup actually.

- Amdahl’s and Gustafson's laws should be interpreted in light of present-day hardware advances, as well as the limitations and tradeoffs that these advances bring with them (as of yet, there is no “perfect” system).

- Interconnects have a major influence on the performance of multi-core systems.