7. Kamil SA, Shalf J, Oliker L, Skinner D (2005) Understanding ultra-scale application communication requirements. In: IEEE international symposium on workload characterization (IISWC) Austin, 6–8 Oct 2005 (LBNL-58059)
8. Kamil S, Chan Cy, Oliker L, Shalf J, Williams S (2010) An auto-tuning framework for parallel multicore stencil computations. In: IPDPS 2010, Atlanta
9. Hendry G, Kamil SA, Biberman A, Chan J, Lee BG, Mohiyuddin M, Jain A., Bergman K, Carloni LP, Kubiatocics J, Oliker L, Shalf J (2009) Analysis of photonic networks for chip multiprocessor using scientific applications. In: NOCS 2009, San Diego
10. Mohiyuddin M, Murphy M, Oliker L, Shalf J, Wawrzynek J, Williams S (2009) A design methodology for domain-optimized power-efficient supercomputing, In: SC 09, Portland
11. Balfour J, Dally WJ (2006) Design tradeoffs for tiled cmp on-chip networks. In ICS '06: Proceedings of the 20th annual international conference on supercomputing

Grid Partitioning

►Domain Decomposition

Gridlock

►Deadlocks

Group Communication

►Collective Communication

Gustafson's Law

John L. Gustafson
Intel Labs, Santa Clara, CA, USA

Synonyms

Gustafson–Barsis Law; Scaled speedup; Weak scaling

Definition

Gustafson's Law says that if you apply P processors to a task that has serial fraction f, scaling the task to take the same amount of time as before, the speedup is

$$\begin{aligned}\text{Speedup} &= f + P(1-f)\\ &= P - f(P-1).\end{aligned}$$

It shows more generally that the serial fraction does not theoretically limit parallel speed enhancement, if the problem or workload scales in its parallel component. It models a different situation from that of Amdahl's Law, which predicts time reduction for a fixed problem size.

Discussion

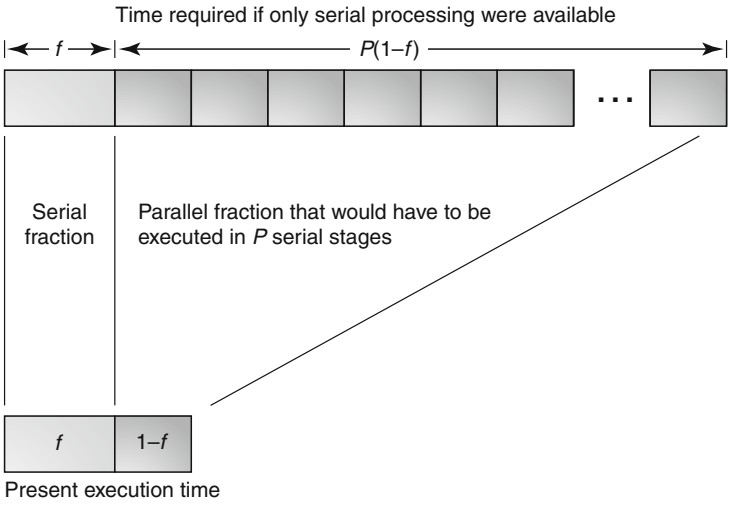
Graphical Explanation

Figure 1 explains the formula in the Definition:

The time the user is willing to wait to solve the workload is unity (lower bar). The part of the work that is observably serial, f, is unaffected by parallelization. The remaining fraction of the work, $1 - f$, parallelizes perfectly so that a serial processor would take P times longer to execute it. The ratio of the top bar to the bottom bar is thus $f + P(1 - f)$. Some prefer to rearrange this algebraically as $P - f(P - 1)$.

The diagram resembles the one used in the explanation of Amdahl's Law (see ►Amdahl's Law) except that Amdahl's Law fixes the *problem size* and answers the question of how parallel processing can reduce the execution time. Gustafson's Law fixes the *run time* and answers the question of how much longer time the present workload would take in the absence of parallelism [5]. In both cases, f is the experimentally observable fraction of the current workload that is serial. The similarity of the diagram to the one that explains Amdahl's Law has led some to "unify" the two laws by a change of variable. It is an easy algebraic exercise to set the upper bar to unit time and express the f of Gustafson's Law in terms of the variables of Amdahl's Law, but this misses the point that the two laws proceed from *different premises*. Every attempt at unification begins by applying the same premise, resulting in a circular argument that the two laws are the same.

The fundamental underlying observation of Gustafson's Law is that more powerful computer systems usually solve larger problems, not the same size problem in less time. Hence, a performance enhancement like parallel processing expands what a user can do with a computing system to match the time the user is willing to wait for the answer. While computing power has increased by many orders of magnitude over the last



Gustafson's Law. Fig. 1 Graphical derivation of Gustafson's Law

half-century (see ►Moore's Law), the execution time for problems of interest has been constant, since that time is tied to human timescales.

History

In a 1967 conference debate over the merits of parallel computing, IBM's Gene Amdahl argued that a considerable fraction of the work of computers was inherently serial, from both algorithmic and architectural sources. He estimated the serial fraction f at about 0.25–0.45. He asserted that this would sharply limit the approach of parallel processing for reducing execution time [1]. Amdahl argued that even the use of two processors was less cost-effective than a serial processor. Furthermore, the use of a large number of processors would never reduce execution time by more than $1/f$, which by his estimate was a factor of about 2–4.

Despite many efforts to find a flaw in Amdahl's argument, "Amdahl's Law" held for over 20 years as justification for the continued use of serial computing hardware and serial programming models.

The Rise of Microprocessor-Based Systems

By the late 1970s, microprocessors and dynamic random-access memory (DRAM) had dropped in price to the point where academic researchers could afford them as components in experimental parallel designs. Work in 1983 by Charles Seitz at Caltech using a message-passing collection of 64 microprocessors [11] showed excellent absolute performance in terms of floating-point operations per second, and seemed to defy Amdahl's pessimistic prediction. Seitz's success led John Gustafson at FPS to drive development of a massively parallel cluster product with backing from the Defense Advanced Research Projects Agency (DARPA). Although the largest configuration actually sold of that product (the FPS T Series) had only 256 processors, the architecture permitted scaling to 16,384 processors. The large number of processors led many to question: *What about Amdahl's Law?* Gustafson formulated a counterargument in April 1986, which showed that performance is a function of both the problem size and the number of processors, and thus Amdahl's Law need not limit performance. That is, the serial fraction f is not a constant but actually decreases with increased problem size. With no experimental evidence to demonstrate the idea, the counterargument had little impact on the computing community.

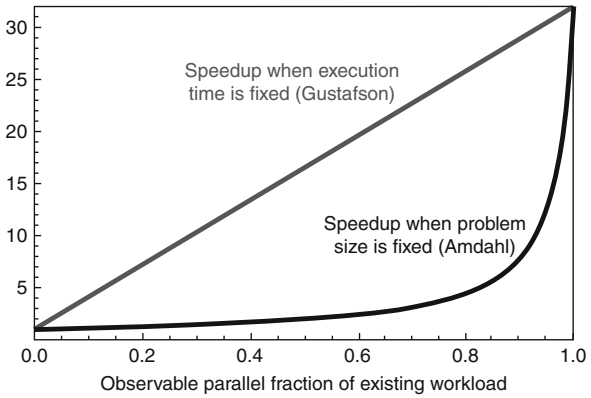
An idea for a source of experimental evidence arose in the form of a challenge that Alan Karp had publicized the year before [8]. Karp had seen announcements of the 65,536-processor CM-1 from Thinking Machines and the 1,024-processor NCUBE10 from nCUBE, and believed Amdahl's Law made it unlikely that such massively parallel computers would achieve a large fraction of their rated performance. He published a skeptical challenge and a financial reward for anyone who could demonstrate a parallel speedup of over 200 times on

three real applications. Karp suggested computational fluid dynamics, structural analysis, and econometric modeling as the three application areas and gave some ground rules to insure that entries focused on honest parallel speedup without tricks or workarounds. For example, one could not cripple the serial version to make it artificially 200 times slower than the parallel system. And the applications, like the three suggested, had to be ones that had interprocessor communication throughout their execution as opposed to "embarrassingly parallel" problems that had communication only at the beginning and end of a run.

By 1987, no one had met Karp's challenge, so Gordon Bell adopted the same set of rules and suggested applications as the basis for the Gordon Bell Award, softening the goal from 200 times to whatever the best speedup developers could demonstrate. Bell expected the initial entries to achieve about tenfold speedup [2].

The purchase by Sandia National Laboratories of the first 1,024-processor NCUBE 10 system created the opportunity for Gustafson to demonstrate his argument on the experiment outlined by Karp and Bell, so he joined Sandia and worked with researchers Gary Montry and Robert Benner to demonstrate the practicality of high parallel speedup. Sandia had real applications in fluid dynamics and structural mechanics, but none in econometric modeling, so the three researchers substituted a wave propagation application. With a few weeks of tuning and optimization, all three applications were running at over 500-fold speedup with the fixed-size Amdahl restriction, and over 1,000-fold speedup with the scaled model proposed by Gustafson. Gustafson described his model to Sandia Director Edwin Barsis, who suggested explaining scaled speedup using a graph like that shown in Fig. 2.

Barsis also insisted that Gustafson publish this concept, and is probably the first person to refer to it as "Gustafson's Law." With the large experimental speedups combined with the alternative model, *Communications of the ACM* published the results in May 1988 [5]. Since Gustafson credited Barsis with the idea of expressing the scaled speedup model as graphed in Fig. 2, some refer to Gustafson's Law as the Gustafson–Barsis Law. The three Sandia researchers



Gustafson's Law. Fig. 2 Speedup possible with 32 processors, by Gustafson's Law and Amdahl's Law

published the detailed explanation of the application parallelizations in a *Society of Industrial and Applied Mathematics* (SIAM) journal [4].

Parallel Computing Watershed

Sandia's announcement of 1,000-fold parallel speedups created a sensation that went well beyond the computing research community. Alan Karp announced that the Sandia results had met his Challenge, and Gordon Bell gave his first award to the three Sandia researchers. The results received publicity beyond that of the usual technical journals, appearing in *TIME*, *Newsweek*, and the US Congressional Record. Cray, IBM, Intel, and Digital Equipment began work in earnest developing commercial computers with massive amounts of parallelism for the first time.

The Sandia announcement also created considerable controversy in the computing community, partly because some journalists sensationalized it as a proof that Amdahl's Law was false or had been "broken." This was never the intent of Gustafson's observation. He maintained that Amdahl's Law was the correct answer but to the wrong question: "How much can parallel processing reduce the run time of a current workload?"

Observable Fraction and Scaling Models

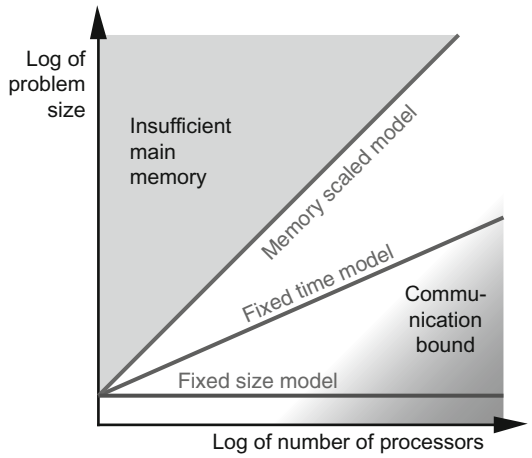
As part of the controversy, many maintained that Amdahl's Law was still the appropriate model to use in all situations, or that Gustafson's Law was simply

a corollary to Amdahl's Law. For scaled speedup, the argument went that one simply works backward to determine what the f fraction in Amdahl's Law must have been to yield such performance. This is an example of circular reasoning, since the proof that Amdahl's Law applies begins by assuming it applies.

For many programs, it is possible to instrument and measure the fraction of time f spent in serial execution. One can place timers in the program around serial regions and obtain an estimate of f. This fraction then allows Amdahl's Law estimates of time reduction, or Gustafson's Law estimates of scaled speedup. Neither law takes into account communication costs or intermediate degrees of parallelism. (When communication costs are included in Gustafson's fixed-time model, the speedup is again limited as the number of processors grows, because communication costs rise to the point where there is no way to increase the size of the amount of work without increasing the execution time.)

A more common practice is to measure the parallel speedup as the number of processors is varied, and fit the resulting curve to derive f. This approach confuses serial fraction with communication overhead, load imbalance, changes in the relative use of the memory hierarchy, and so on. Some refer to the requirement to keep the problem size the same yet use more processors as "strong scaling." Still, a common phenomenon that results from "strong scaling" is that it is *easier*, not *harder*, to obtain high amounts of speedup. When spreading a problem across more and more processors, the *memory per processor* goes down to the point where the data fits entirely in cache, resulting in *superlinear speedup* [3]. Sometimes, the superlinear speedup effects and the communication overheads partially cancel out, so what appears to be a low value of f is actually the result of the combination of the two effects. In modern parallel systems, performance analysis with either Amdahl's Law or Gustafson's Law will usually be inaccurate since communication costs and other parallel processing phenomena have large effects on the speedup.

In Fig. 3, Amdahl's Law governs the Fixed-Sized Model line, Gustafson's Law governs the Fixed-Time Model line, and what some call the Sun–Ni Law governs the Memory Scaled Model [12]. The fixed-time model line is an irregular curve in general, because of the communication cost effects and because the percentage of the problem that is in each memory tier (mass storage, main RAM, levels of cache) changes with the use of more processors.



Gustafson's Law. Fig. 3 Different scaling types and communication costs

Analogies

There are many aspects of technology where an enhancement for time reduction actually turns out to be an enhancement for what one can accomplish in the same time as before. Just as Amdahl's Law is an expression of the more general Law of Diminishing Returns, Gustafson's Law is an expression of the more general observation that technological advances are used to improve what humans accomplish in the length of time they are accustomed to waiting, not to shorten the waiting time.

Commuting Time

As civilization has moved from walking to horses to mechanical transportation, the average speed of getting to and from work every day has gone up dramatically. Yet, people take about half an hour to get to or from work as a tolerable fraction of the day, and this amount of time is probably similar to what it has been for centuries. Cities that have been around for hundreds or thousands of years show a concentric pattern that reflect the increasing distance people could commute for the amount of time they were able to tolerate.

Transportation provides many analogies for Gustafson's Law that expose the fallacy of fixing the size of a problem as the control variable in discussing large performance gains. A commercial jet might be able to travel 500 miles per hour, yet if one asks "How much will it reduce the time it takes me presently to walk to work and back?" the answer would be that it does not help at all. It would be easy to apply an Amdahl-type argument to the time to travel to an airport as the serial fraction, such that the speedup of using a jet only applies to the remaining fraction of the time and thus is not worth doing. However, this does not mean that commercial jets are useless for transportation. It means that faster devices are for larger jobs, which in this case means longer trips.

Here is another transportation example: If one takes a trip at 30 miles per hour and immediately turns around, how fast does one have to go to average 60 miles per hour? This is a trick question that many people incorrectly answer, "90 miles per hour." To average 60 miles per hour, one would have to travel back at infinite speed, that is, return *instantly*. Amdahl's Law applies to this fixed-distance trip. However, suppose the question were posed differently: "If one travels for an hour at 30 miles per hour, how fast does one have to travel in the next hour to average 60 miles per hour?" In that case, the intuitive answer of "90 miles per hour" *is the correct one*. Gustafson's Law applies to this fixed-time trip.

The US Census

In the early debates about scaled speedup, Heath and Worley [6] provided an example of a fixed-sized problem that they said was not appropriate for Gustafson's Law and for which Amdahl's Law should be applied: the US Census. While counting the number of people in the USA would appear to be a fixed-sized problem, it is actually a perfect example of a fixed-time problem since the Constitution mandates a complete headcount every 10 years. It was in the late nineteenth century, when Hollerith estimated that the population had grown to the point where existing approaches would take longer than 10 years that he developed the card punch tabulation methods that made the process fast enough to fit the fixed-time budget.

With much faster computing methods now available, the Census process has grown to take into account many more details about people than the simple head count that the Constitution mandates. This illustrates a connection between Gustafson's Law and the jocular Parkinson's Law: "Work expands to fill the available time."

Printer Speed

In the 1960s, when IBM and Xerox were developing the first laser printers that could print an entire page at a time, the goal was to create printers that could print several pages per second so that printer speed could match the performance improvements of computing speed. The computer printouts of that era were all of monospaced font with a small character set of uppercase letters and a few symbols. Although many laser printer designers struggled to produce such simple output with reduced time per page, the product category evolved to produce high quality output for desktop publishing instead of using the improved technology for time reduction. People now wait about as long for a page of printout from a laser printer as they did for a page of printout from the line printers of the 1960s, but the task has been scaled up to full color, high resolution printing encompassing graphics output, and a huge collection of typeset fonts from alphabets in all the world's languages. This is an example of Gustafson's Law applied to printing technology.

Biological Brains

Kevin Howard, of Massively Parallel Technologies Inc., once observed that if Amdahl's Law governed the behavior of biological brains, then a human would have about the same intelligence as a starfish. The human brain has about 100 billion neurons operating in parallel, so for us to avoid passing the point of diminishing returns for all that parallelism, the Amdahl serial fraction f would have to be about 10^{-14}. The fallacy of this seeming paradox is in the underlying assumption that a human brain must do the same task a starfish brain does, but must reduce the execution time to nanoseconds. There is no such requirement, and a human brain accomplishes very little in a few nanoseconds no matter how many neurons it uses at once. Gustafson's Law says that on a time-averaged basis, the human brain will accomplish *vastly more complex tasks* than what a starfish can attempt, and thus avoids the absurd conclusion of the fixed-task model.

Perspective

The concept of scaled speedup had a profound enabling effect on parallel computing, since it showed that simply asking a different question (and perhaps a more realistic one) renders the pessimistic predictions of Amdahl's Law moot. Gustafson's 1988 announcement of 1,000-fold parallel speedup created a turning point in the attitude of computer manufacturers towards massively parallel computing, and now all major vendors provide platforms based on the approach. Most (if not all) of the computer systems in the TOP500 list of the worlds' fastest supercomputers are comprised of many thousands of processors, a degree of parallelism that computer builders regarded as sheer folly prior to the introduction of scaled speedup in 1988.

A common assertion countering Gustafson's Law is that "Amdahl's Law still holds for scaled speedup; it's just that the serial fraction is a lot smaller than had been previously thought." However, this requires inferring the small serial fraction from the measured speedup. This is an example of circular reasoning since it involves choosing a conclusion, then working backward to determine the data that make the conclusion valid. Gustafson's Law is a simple formula that predicts scaled performance from experimentally measurable properties of a workload.

Some have misinterpreted "scaled speedup" as simply increasing the amount of memory for variables, or increasing the fineness of a grid. It is more general than this. It applies to every way in which a calculation can be improved somehow (accuracy, reliability, robustness, etc.) with the addition of more processing power, and then asks *how much longer the enhanced problem would have taken to run without the extra processing power*.

Horst Simon, in his 2005 keynote talk at the International Conference on Supercomputing, "*Progress in Supercomputing: The Top Three Breakthroughs of the Last 20 Years and the Top Three Challenges for the Next 20 Years*," declared the invention of the Gustafson's scaled speedup model as the number one achievement in high-performance computing since 1985.

Related Entries

►Amdahl's Law
►Distributed-Memory Multiprocessor
►Metrics

Bibliographic Entries and Further Reading

Gustafson's 1988 two-page paper in the *Communications of the ACM* [5] outlines his basic idea of fixed-time performance measurement as an alternative to Amdahl's assumptions. It contains the rhetorical question, "How can this be, in light of Amdahl's Law?" that some misinterpreted as a serious plea for the resolution of a paradox. Readers may find a flurry of responses in *Communications* and elsewhere, as well as attempts to "unify" the two laws.

An objective analysis of Gustafson's Law and its relation to Amdahl's Law can be found in many modern textbooks on parallel computing such as [7], [9], or [10]. In much the way some physicists in the early twentieth century refused to accept the concepts of relativity and quantum mechanics, for reasons more intuition-based than scientific, there are computer scientists who refuse to accept the idea of scaled speedup and Gustafson's Law, and who insist that Amdahl's Law suffices for all situations.

Pat Worley analyzed the extent to which one can usefully scale up scientific simulations by increasing their resolution [13]. In related work, Xian-He Sun and Lionel Ni built a more complete mathematical framework for scaled speedup [12] in which they promote the idea of memory-bounded scaling, even though execution time generally increases beyond human patience when the memory used by a problem scales as much as linearly with the number of processors. In a related vein, Vipin Kumar proposed "Isoefficiency" for which the memory increases as much as necessary to keep the efficiency of the processors at a constant level even when communication and other impediments to parallelism are taken into account.

Bibliography

1. Amdahl GM (1967) Validity of the single-processor approach to achieve large scale computing capabilities. AFIPS Joint Spring Conference Proceedings 30 (Atlantic City, NJ, Apr. 18–20), pp 483–485. AFIPS Press, Reston VA. At http://www-inst.eecs.berkeley.edu/~n 252/paper/Amdahl.pdf
2. Bell G (interviewed) (1987) An interview with Gordon Bell. IEEE Software, vol 4, No. 4 (July 1987), pp 102–104
3. Gustafson JL (1990) Fixed time, tiered memory, and superlinear speedup. Distributed Memory Computing Conference, 1990, Proceedings of the Fifth, vol 2 (April 1990), pp 1255–1260. ISBN: 0-8186-2113-3

4. Gustafson JL, Montry GR, Benner RE (1988) Development of parallel methods for a 1024-processor hypercube. SIAM Journal on Scientific and Statistical Computing, vol 9, No. 4, (July 1988), pp 609–638
5. Gustafson (1988) Reevaluating Amdahl's Law. Communications of the ACM, vol 31, No. 5 (May 1988), pp 532–533. DOI=10.1145/42411.42415
6. Heath M, Worley P (1989) Once again, Amdahl's Law. Communications of the ACM, vol 32, No. 2 (February 1989), pp 258–264
7. Hwang K, Briggs F, Computer Architecture and Parallel Processing, 1990. McGraw-Hill Inc., 1990. ISBN: 0070315566
8. Karp A (1985) http://www.netlib.org/benchmark/karp-challenge
9. Lewis TG, El-Rewini H (1992) Introduction to Parallel Computing, Prentice Hall. ISBN: 0-13-498924-4. 32–33
10. Quinn M (1994) Parallel Computing: Theory and Practice. Second edition. McGraw-Hill, Inc
11. Seitz CL (1986) Experiments with VLSI ensemble machines. Journal of VLSI and Computer Systems, vol 1, No. 3, pp 311–334
12. Sun X-H, Ni L (1993) Scalable problems and memory-bounded speedup." Journal of Parallel and Distributed Computing, vol 19, No. 1, pp 22–37
13. Worley PH (1989) The effect of time constraints on scaled speedup. Report ORNL/TM 11031, Oak Ridge National Laboratory

Gustafson–Barsis Law

►Gustafson's Law