Frequently Asked Questions about the article

I.L. Markov, Limits on fundamental limits to computation,

Nature 512, 147-154 (14 August 2014)

http://www.nature.com/nature/journal/v512/n7513/full/nature13570.html

Revision 141120 **Contact: imarkov@eecs.umich.edu**

Latest revision: http://web.eecs.umich.edu/~imarkov/Limits-Nature-FAQ.pdf

1. **Is there a short summary of the article?**
   Yes, please see this 4-min video summary http://vimeo.com/103772289, the Nature editorial (http://www.nature.com/news/future-computing-1.15704) and the NSF press release (http://www.nsf.gov/news/news_summ.jsp?cntn_id=132339), as well as coverage in EE Times (http://www.eetimes.com/document.asp?doc_id=1323507) and Ars Technica (http://arstechnica.com/science/2014/08/are-processors-pushing-up-against-the-limits-of-physics/).

2. **The article mostly discusses hardware limits. What about limits to software?**
   While crucially important in modern computing, software appears to be a choice, not a necessity, at least from the scientific (rather than engineering) perspective. A software program can be implemented in a specialized integrated circuit to obtain improvements in speed and energy. Of course, this is only done in rare cases for practical and business reasons, as explained in the Nature article. On the other hand, limits to software are certainly worth a separate discussion. The Nature paper touches upon several algorithmic limits such as P vs. NP and limits to parallel algorithms.

3. **How promising are current and emerging memory technologies?**
   Conventional DRAM memories are not scaling as well as before because of limits on the size of capacitors. There are several exciting new memory technologies that add nonvolatility, and also new stacked 3D memories that offer greater data volumes and faster access. Faster memory can significantly speed up computation. It is also possible to embed memory blocks into general-purpose chips and even perform limited computation directly in memory.

   See: Top 10 Candidates for Next-Gen Storage
   http://www.eetimes.com/document.asp?doc_id=1324027

4. **Where can I find a summary of numerical limits for MOSFETs in various configurations?**
   In the 2001 paper "Device Scaling Limits of Si MOSFETs and Their Application Dependencies"
   http://www.ece.ncsu.edu/asic/ece733/papers/Devices/FETscaling.pdf

5. **What about new types of transistors?**
   They may be smaller, faster, or more energy-efficient, but not much cheaper to produce. Power and performance bottlenecks are shifting toward interconnects. New transistors (such as CNT-FETs) may help by increasing drive strength, so that longer interconnect can be driven without buffering. Also, it is not always clear how to use so many transistors on one chip other than in large memories.

6. **What could be the contribution of 3-D circuit integration?**
Such techniques help combine chips manufactured in different technologies, but currently do not use many 3D interconnects for technical reasons. Full-blown 3D chips tend to be interconnect-limited. Recall that conventional routing has already been 3D for many years. Now with 3D placement of gates, we don't really have another physical dimension for wiring. Making room for interconnect decreases device density. Yet, 3D integration may be useful for circuits with special-case interconnect structures. Heat removal is an added problem for 3D circuits, so CPUs may not be the best candidates for such integration because they "run hot". New cooling technologies may change this.

7. **You outlined serious limits to parallelism, but we see applications with unbounded parallelism, such as image processing - how do you explain this?**
Image processing and other "embarrassingly parallel" applications require very little communication. You may be dealing with 100,000 separate computations (say, one per pixel or one per shaded triangle). Unrelated computations can certainly be performed in parallel, but when results are assembled in one place or shared inputs are read, this links those computations, The amount of "easy" parallelism depends on how independent the output bits are from each other. For example, if you add a vector with 1M numbers to another such vector, the output values are independent. If you need to add two 1M-digit numbers, the output digits are closely related. If you need to multiply 1M-digit numbers, the output digits are even more related. For modular exponentiation, you get fewer outputs, but they are even more tightly connected. In the meantime, slow processing can mask interconnect latencies and thus push back the limits related to communication.

See the book R. Greenlaw, H. J. Hoover, W. L. Ruzzo, Limits to Parallel Computation: P-completeness Theory: https://homes.cs.washington.edu/~ruzzo/papers/limits.pdf

8. **Shouldn't we invest in better compilers for parallel computing? Parallel algorithms are difficult to find, so parallelism should be extracted automatically from sequential programs.**
This is a very well-established area of research both in academia and in the industry. Over the last 20-30 years, it had notable successes, but its limitations are known well too. For many problems, finding asymptotically efficient parallel algorithms (by hand or through a clever compiler) would solve major open problems in theoretical computer science, and is generally considered unlikely. For some problems, such algorithms are known, but provably differ from best possible sequential algorithms so much that there is no hope for a compiler to convert a sequential algorithm into a good parallel algorithm (see the above-referenced book on limits to parallel computation).

9. **How promising is approximate computing? What are its limits?**
The Nature paper recalls that slightly perturbing the input of the Simplex algorithm for linear programming helps avoiding the exponential worst cases with high probability, and this explains why the Simplex algorithm does not take exponential time in practice - pathological inputs are few and far in between. But for most computations, twiddling the input or approximating the output does not affect complexity as much. Common sense tells us that if an application does not require high precision, achieving such precision is a waste of resources. For example, iterative algorithms in numerical analysis use related convergence criteria. However, if we start aggressively relaxing the accuracy of algorithmic or even hardware blocks, approximation errors may accumulate. In other words, straightforward approximate computing is not closed under composition, whereas exact computing is.

10. **What about memristors**?
A promising technology, but apparently large-scale memristor blocks must be hybridized with CMOS, which may dilute their advantage to some extent. Memristors look useful for several specific applications, but their utility in general-purpose computers remains to be seen.

11. **How promising is optical computing?**
Its density is limited by wavelengths of common light sources and absorption spectra of common materials. Features in modern ICs are much smaller than practical light wavelengths, but X-ray light is absorbed too well by relevant materials. Photonic data are almost impossible to store in large quantities and need to be converted into electronic or magnetic states, which requires large specialized devices (much larger than wires and photonic waveguides). Bending photonic interconnects leads to energy loss, and techniques for crossing two optical interconnects also incur losses. Yet, Intel found great uses for optical interconnect between chips in a data servers and for photonic on-chip waveguides in multicore chips.

12. **Can the interconnect bottleneck be solved using wireless communication?**
The bandwidth available to wireless communication appears much smaller than the bandwidth available to wired communications. Additional limitations can be found for specific frequencies, say, 2.4GHz, which corresponds to the 125mm wavelength and requires fairly large transmitters (compared to the size of individual transistors). Very few such transmitters can fit on a chip.

13. **What about graphene**?
Unlike semiconductors, graphene does not possess an energy gap, which undermines its uses in digital computing. Ongoing research aims to impart graphene with new properties, but remains far from successful engineering technologies at the moment.

14. **Why do you claim that a signal cannot cross a chip in one clock cycle anymore, whereas light can travel 0.3m in 1ns (one clock cycle of a 1GHz CPU)?**
On-chip signals do not travel in the vacuum, but even the EM propagation speed in Copper vastly overestimates how quickly a signal transition can move on a chip. Delay is computed as the RC product, which grows quadratically with wire length. Propagation delay can be made linear, but at the cost of inserting buffers (which pump energy into the wire). Each buffer includes several transistors, which adds switching delay - a huge slowdown compared to the speed of light.

15. **When you discuss the thermodynamic threshold and the reversibility of computation, how would a chip with more than 10^9 transistors change your conclusions?**
This discussion meant to suggest (through an omitted lower-bound calculation) that power consumption per transistor remains far from the thermodynamic threshold, questioning the logic behind reversible computing. Some chips today include more than 10^9 transistors, but are dominated by CPU caches, where transistors switch very infrequently. A rigorous version of this argument would need to account for activity factors - how often do transistors switch on average? The number of transistors switching within the same clock cycle is still far below 10^9 today.

16. **In the conclusions, we read that "only CPUs, graphics processing units, field-programmable gate-arrays and dense memory integrated circuits will remain viable at the end of Moore's law". Which other types of chips would you consider in this context?**
It was a mistake on my part to neglect *wireless baseband processors* used in cellular phones, which include large digital circuits, such as Fast Fourier Transforms, and the algorithms specified in wireless

communication standards, such as the 802.11 family. In addition to the huge market, these chips must be very energy-efficient because they are used in portable electronics. *Bitcoin miners* should also be considered, albeit for different reasons. Their market is relatively small and volatile, but these chips are essentially devices to convert electricity into money. Therefore, the competition in energy-efficiency of these chips is fierce, while they can tolerate higher manufacturing cost.

17. **You seem pessimistic about quantum computing, why so**?
Quantum computing remains a very interesting line of research, but its potential for general-purpose computing is small. It can make significant impact in 10-20 years, but not by competing with traditional technologies. Even in the field of molecular simulation, many challenges can be addressed without modeling quantum effects, but must use a large number of non-quantum parameters - here conventional application-specific integrated circuits are far more promising than quantum computers. Moreover, some of the most promising research directions in quantum computing currently fall under basic science, rather than engineering. For example, some rely on new physical particles that have not yet been discovered. Dense memories - a pillar of conventional computing - are not even on the horizon for quantum computing. Other challenges tracked by the semiconductor industry - device density, power density control, interconnect-limited design, fabrication cost and yield - have not been addressed at the same depth for quantum computers. Some of the recent claims of commercial quantum-computing devices were apparently made to stake an early claim and maintain a development muscle, in case someone figures out quantum computers in the future. In the meantime, several impressive results in applied physics have been inspired by the promise of quantum computing.

18. What about DNA computing?
It is many orders of magnitude slower than conventional computing and does not offer compelling engineering advantages. However, biological processes have to perform computation on their own timescale using available hardware (or *wetware*), and biologists legitimately need to understand how those computations work, perhaps even alter or extend those computations.

19. **How will biological models like neural networks figure into extending Moore's Law? What are the smartest obstacles to try surmounting today?**
Biological systems are also subject to fundamental limits. The human brain connectivity is 3D, but individual "devices" are quite big and slow. Just like modern integrated circuits, the brain is interconnect-limited. It is much more energy-efficient, uses low supply voltages, liquid cooling, and a very different power network. It also needs to rest and chemically clean itself — we don't quite know why. We also know that the brain is disappointing as a general-purpose computer. It can't multiply many 64-bit numbers per second, copy stored information in bulk, or think a hundred thoughts at once (texting while driving is illegal for that reason). However, the brain is a great multimedia processor, handles uncertainty well, and is capable of intuition, creativity, and other types of high-level reasoning. Figuring how this is done leaves researchers more than enough work.
If you are dealing with conventional hardware, there is significant room in large-scale algorithmic optimization of available resources so that they are used more efficiently. Conventional design techniques focus on one aspect of the system, so co-optimizing physical layout and logic circuits, number-crunching and memory access latencies, hardware and software, etc remains promising.

20. **What about brain simulators, such as recent work from IBM and HRL**?
Brain simulators have not produced faithful results so far, and their main goal has been to outperform existing computers using neuromorphic algorithms and hardware. Such comparisons have been hotly debated, and my understanding is that conventional computers and algorithms currently remain far

superior in full applications. On the positive side, reverse engineering the brain is a grand challenge for science and engineering, and simulating the brain is one of the best strategies toward the goal.

21. **What do you think are the most promising technology developments that will improve computer performance in the near future**?
The increase in available memory capacity, as well as optical communication between chips and chip modules. Positioning memory blocks closer to where they are used is another promising optimization - while a well-known concept, its application currently requires specialty work in individual use cases. Fast and dense nonvolatile memories (known as *universal memories*) promise to reduce power consumption and improve performance portable electronics, while greatly reducing the time needed to turn a device on. Among non-memory research-stage technologies, carbon-nanotube transistors look promising in both power and performance, but have not yet achieved sufficient densities to compete with leading-edge silicon CMOS integrated circuits. Also, current manufacturing techniques for carbon-based transistors suffer very significant process variations. Many special-purpose computations can be performed much faster by developing specialized chips, even with existing computing technologies, but designing such chips requires significant effort. When researchers use emerging technologies in such cases, it is not always clear how much gain is due to emerging technologies and how much is due to specialization, especially when the comparisons are made against general-purpose desktop hardware. Yet another line of study is cryogenic (very low temperature) electronics, where the main promise is to improve energy efficiency, while amortizing the costs of cooling within a stationary data center. Among many useful effects here are the reduction of thermal noise and leakage (which promise to reduce threshold and supply voltage), better electrical isolation of circuit components, fine control over individual atoms and electrons (which may support very small switching devices), as well as superconductivity.

22. **What are the pitfalls in comparing new technologies to conventional computing**?
Not using the best that conventional computing can offer, such as comparing a specialized chip to a sequential program running on a desktop. Or parallelizing a weak algorithm and not comparing it to best sequential algorithms. A good comparison takes a lot of work, may require significant expertise, and is often best outsourced to a third party or several qualified groups, to ensure adequate competence and average out possible biases.

23. **What about my favorite limit to computation that was not covered in the Nature article?**
Not all limits and technologies fit in limited page space. Many known limits are far from being tight (with respect to technologies), and some are too complicated to describe in a survey article. Among the most important such limits is the Margolus-Levitin theorem which concludes that "The processing rate cannot be higher than 6×10^{33} operations per second per joule of energy". Among other things, this result accounts for the possibility of quantum computation. For details, see http://en.wikipedia.org/wiki/Margolus%E2%80%93Levitin_theorem

24. **Why did you not cite Bekenstein and other authors who proposed limits you described?**
We limited references to 99 and did not cite several results which can be found by a simple Web search, especially when Wikipedia pages exist with detailed descriptions and further references.

25. **Do you think there are significant limits to computation left for us to discover?**
Absolutely. And studying the limits of existing limits reveals hints as to additional limits.