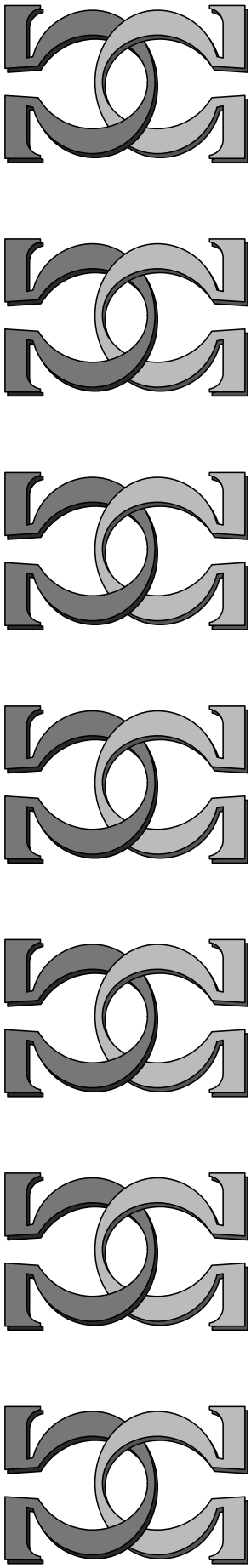

CDMTCS
Research
Report
Series

The Road to Quantum Computational Supremacy

C. S. Calude[1] and E. Calude[2]

[1]University of Auckland, New Zealand
[2]Massey University at Albany, Auckland, New Zealand

CDMTCS-514
November 2017

Centre for Discrete Mathematics and
Theoretical Computer Science

The Road to Quantum Computational Supremacy

Cristian S. Calude[1] and Elena Calude[2]

[1] Department of Computer Science, University of Auckland
Private Bag 92019, Auckland, New Zealand
c.calude@auckland.ac.nz
[2] Institute of Natural and Mathematical Sciences, Massey University at Albany
Private Bag 102-904 North Shore MSC, Auckland, New Zealand
e.calude@massey.ac.nz

Abstract. We present an idiosyncratic view of the race for quantum computational supremacy. Google's approach and IBM challenge are examined. An unexpected side-effect of the race is the significant progress in designing fast classical algorithms. Quantum supremacy, if achieved, won't make classical computing obsolete.

A hyper-fast quantum computer is the digital equivalent of a nuclear bomb; whoever possesses one will be able to shred any encryption and break any code in existence.[3] [41]

1 Fairy tales or more cautionary tales?

Following the development of Shor's quantum algorithm [57] in 1994 and Grover's quantum algorithm [38] two years later, quantum computing was seen as a bright beacon in computer science, which led to a surge of theoretical and experimental results. The field captured the interest and imagination of the large public and media, and not surprisingly, unfounded claims about the power of quantum computing and its applications proliferated.

A certain degree of pessimism began to infiltrate when experimental groups floundered while attempting to control more than a handful of qubits. Recently, a broad wave of ambitious industry-led research programmes in quantum computing—driven by D-Wave Systems,[4] the tech giants Google, IBM, Microsoft, Intel and startups like Rigetti Computing and Quantum Circuits Incorporated—has emerged[5] and bold claims about a future revolutionised by quantum computing are resurfacing.

Governments are also involved: phase 1 (2015–2019) £330 million of the UK government programme on quantum technologies [5] is rolling and the European Commission has announced a €1 billion initiative in quantum technology [7]. The European flagship quantum programme, whose explicit goal is to stimulate a "second quantum revolution", aims to "build a universal quantum computer able to demonstrate the resolution of a problem that, with current techniques on a supercomputer, would take longer than the age of the universe" by 2035, [1]; see also Figure 1.

Undoubtably, these programmes are extremely beneficial to the development of various quantum technologies, but, are the claims about the future of quantum computing realistic? "We tend to be too optimistic about the short run, too pessimistic about the long run" said recently J. Preskill [55]; see also [14, 60].

[3] A typical example of incorrect, largely-spread, sentence quoted from a recent mystery novel.
[4] The company relatively steady progress in producing and selling the first series of D-Wave quantum computers has gone from 28 qubits in 2007 to more than 2,000 in their latest 2000Q^{TM} System machine [6].
[5] Of course, the industry work is based and has continued the academic efforts, sometimes using successful experimentalists from academia, like Google.

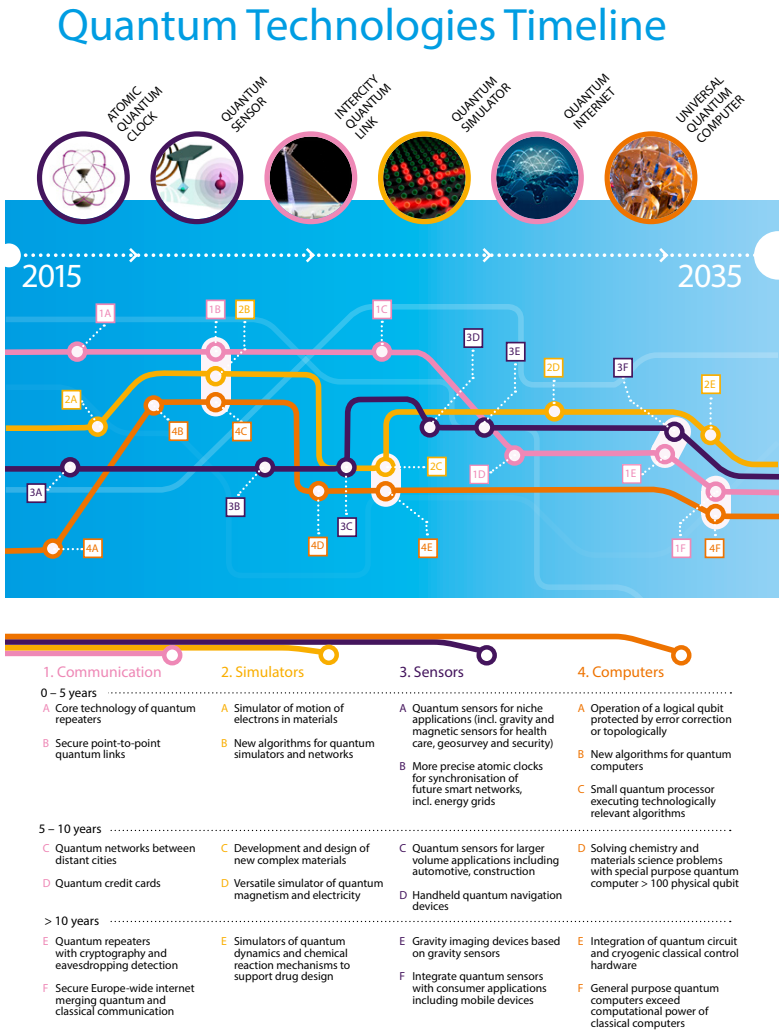

Fig. 1. Quantum timeline: 2015–2035, [1]

2 Quantum algorithmics

While Shor's algorithm, Deutsch-Jozsa algorithm and various others in the "black-box" paradigm[6] are believed to provide an exponential speedup over classical computers, this is far from the case in general. We said "believed" because the superiority of Shor's quantum algorithm over classical ones is still an open problem and various techniques allowing efficient classical simulation of quantum algorithms have been successfully developed [13, 24, 37] even for some "black-box" quantum ones [12, 26].

In fact, since the introduction of Shor's and Grover's algorithms some twenty years ago, the development within the field of quantum algorithmics has been rather slow—see [8] for a global picture—and many of them are novel uses of a handful of core quantum algorithms. So, why are there so few quantum algorithms that offer speed-up over classical algorithms? Although written more than a decade ago, Shor's article [58] is still actual:

[6] Where access to a quantum black-box or "oracle" with certain structural properties is assumed.

> The first possible reason is that quantum computers operate in a manner so different from classical computers that our techniques for designing algorithms and our intuitions for understanding the process of computation no longer work. The second reason is that there really might be relatively few problems for which quantum computers can offer a substantial speed-up over classical computers, and we may have already discovered many or all of the important techniques for constructing quantum algorithms.

Best quantum algorithms typically provide a quadratic or low-order polynomial speedup [36]. Furthermore, there are pointers [9,21] suggesting that quantum computers cannot offer more than a (perhaps small) polynomial advantage for **NP**-complete problems,[7] and such a speedup would struggle to compete with the heuristic approaches commonly used to solve them in practice. However, even a polynomial-order speedup could be of significant benefit for problems requiring exact solutions or for problems that can classically be solved in sub-exponential time, like the graph isomorphism problem (see [29]).

Grover's quantum algorithm [38] is an interesting example: access to an unsorted quantum database that can be queried with a quantum input is given, and asked if it contains a specific entry. Grover's algorithm offers a *provable* speedup. However, the speedup is not exponential and, more importantly, the problem it solves is far from being realistic: the cost of constructing the quantum database could negate any advantage of the algorithm, and in many classical scenarios one could do much better by simply creating (and maintaining) an ordered database. Using Grover's algorithm as a subroutine for solving problems in image processing is more efficient because the cost of preparing the quantum "database" can be spread out over several calls [45]; this strategy motivated a new hybrid quantum-classical paradigm for embedded quantum annealing algorithms [15]. Other applications are discussed in [48].

Quantum simulation, quantum-assisted optimisation and quantum sampling are believed to offer near-term quantum solutions to hard problems that may lead even to commercialisation [47].

3 What is quantum computational supremacy?

The quantum computational advantage for simulating quantum systems was first stated by Feynman in 1981, in one of the pioneering papers in quantum computing [35] (the other one was Manin [46]). What is the justification of Feynman's insight? According to the data processing inequality [19,31], (classical) post-processing cannot increase information. This suggests that to run an accurate classical simulation of a quantum system one must know a lot about the system before the simulation is started [17]. Feynman [35] argued that a quantum computer does not need to have so much knowledge. This line of reasoning seemingly inspired Deutsch [34] to state

> **The postulate of quantum computation**: Computational devices based on quantum mechanics will be computationally superior compared to digital computers.

A spectacular support for this postulate came from Shor's 1994 polynomial factoring quantum algorithm [57] in spite of the fact that the problem whether factoring is in **P** was, and still is, open. The belief that factoring integers is computationally hard[8] is essential for much of modern cryptography and computing security.

In 2011 the syntagm "quantum supremacy" was coined and discussed[9] by J. Preskill in his Rapporteur talk "Quantum Entanglement and Quantum Computing" [53] at the 25th *Solvay Conference on Physics* (Brussels, Belgium, 19–22 October 2011):

[7] Perhaps the most important class of "difficult computational problems" such as the well known travelling-salesman problem, which have applications in almost every area of science and beyond, from planning and logistics to microchip manufacturing.

[8] For results pointing to the opposite assumption see [13,24,32,37,50].

[9] The use of the word "supremacy"—which denotes "the state or condition of being superior to all others in authority"—was criticised in [63] because the syntagm 'white supremacy' is associated with the racial segregation and discrimination of the apartheid regime of South Africa. Proposals like "quantum advantage" or "quantum superiority" have been discussed [4], but to date none has gained ground.

> We therefore hope to hasten the onset of the era of quantum supremacy, when we will be able to perform tasks with controlled quantum systems going beyond what can be achieved with ordinary digital computers.

Recently, quantum supremacy was described in [22] as follows:

> Quantum supremacy is achieved when a formal computational task is performed with an existing quantum device which cannot be performed using any known algorithm running on an existing classical supercomputer in a reasonable amount of time.

Note the imprecision in the above formulation: the comparison is made with "any known algorithm running on an existing classical supercomputer" and the classical computation takes "a reasonable amount of time". Can this imprecision be decreased or, even better, eliminated? Just as there is no current proof that $\mathbf{P} \neq \mathbf{NP}$—one of the important open problems in classical complexity theory—there is no mathematical proof for the Postulate of quantum computation; in fact, the Postulate is not amenable to a proof. The hypothesis $\mathbf{P} \neq \mathbf{NP}$ can be used for deriving useful results; similarly, adopting assumptions in terms of both quantum physics and classical complexity theory—which can be justified heuristically or experimentally—can lead to precise statements which can be proved or disproved. The following two assumptions

> **The postulate of noise**: Quantum systems are inherently noisy.
>
> **The Extended Church-Turing Thesis**: A probabilistic Turing machine can efficiently simulate any realistic model of computation.

have been used by Kalai [42] to challenge the Postulate of quantum computation. Here "efficiently" means "with at most polynomial overhead"; the adjective "realistic" (or "reasonable" as an alternative) refers to a "physically realisable in principle". It is worth mentioning that these assumptions are themselves challengeable; see for example [22] for the Extended Church-Turing Thesis.

A quantum computational supremacy experiment has to prove both a lower bound and an upper bound. In Google's proposed experiment—to be discussed in details in Section 6—the upper bound is given by a quantum algorithm running on a quantum computer with 49 qubits[10]—a mathematical fact and an engineering artefact (the construction of the quantum machine); the lower bound is necessary for proving that no classical computer can simulate the sampling from the output distributions of pseudo-random quantum circuits.

Upper bounds are positive results while lower bounds are negative. Upper bounds are useful when we want to show that a problem can be solved by a "good" algorithm. But if we want to argue that no algorithm solving a problem can be better than a given one, or perhaps that some problem is so hard that we can't possibly hope to find a good solution to it, we need lower bounds.

In mathematics and theoretical computer science it is well-known that negative results are more difficult to prove than positive ones. In classical computability theory it is more difficult to prove incomputability than computability, and in complexity theory lower bounds are more difficult to prove than upper bounds [59]. The superiority of Shor's quantum algorithm [57] is a prime example. A methodology for proving lower bounds in quantum computing is discussed in [39, p. 144–149]. Sometimes unproved claims about the quantum superiority of a quantum algorithm have been shown to be incorrect: an example is the superiority of Deutsch's quantum algorithm over any classical one, see [26, 34].

[10] A qubit is a 2-state quantum system. There are many ways to build qubits, hence not all qubits are equal. The magic number 49 (or 50) refers to qubits in the quantum circuit model which are more difficult to control than the qubits used by the D-Wave machine [27] (to embed a complete graph of N vertices in D-Wave hardware Chimera graph we need approximately N^2 qubits, so 2,048 D-Wave qubits correspond to about fully connected 45 qubits) or the trapped atom qubits used by specialised quantum simulators [20, 65].

Another issue is correctness: how do we know that the quantum computer solution is indeed correct—quantum computing is a probabilistic type of computation—if we can't check it with a reliably tested classical computer? Even classically correctness is a difficult problem. The Ackermann A function [16] is a singular example: computing the value of $A(x, y)$ is prohibitively difficult because the function is computable but not primitive recursive, but testing the predicate $A(x, y) = z$ is very easy [25].

Finally, the discussion about quantum supremacy suggests a misleading comparison between classical and quantum computing. If a quantum computer can outdo **any** classical computer on one problem we have quantum supremacy, even if classical computers could be at least as good as quantum ones in solving many (most) other problems.

Put it bluntly, *quantum supremacy, if achieved, won't make classical computing obsolete.* In fact, the hybrid approach combining quantum and classical computing, briefly mentioned in Section 2, could be a good strategy in solving some (many) difficult problems [15].

4 Criteria for quantum computational supremacy

Harrow and Montanaro [40] have recently proposed a reasonable list of criteria for a quantum supremacy experiment. According to them we need to have:

1. a well-defined computational problem,
2. a quantum algorithm solving the problem which can run on a near-term hardware capable of dealing with noise and imperfections,
3. an amount of computational resources (time/space) allowed to any classical competitor,
4. a small number of well-justified complexity-theoretic assumptions,
5. a verification method that can efficiently distinguish between the performances of the quantum algorithm from **any** classical competitor using the allowed resources.

Large integer factoring is a typical problem for a quantum supremacy experiment. Indeed, it is well-defined, it has huge practical importance, there are efficient quantum algorithms solving it (Shor's algorithm and variants [32,57]), the complexity-theoretic assumption is that no classical algorithm can factor essentially faster than the current ones and the solution is quickly verifiable. This seems an almost ideal candidate, except for a) the strong complexity-theoretic assumption [50] and b) the lack of a near-term hardware running such a quantum algorithm for sufficiently large integers (say a 2,048-bit number), see [40]. A possible solution for b) could be a hybrid (quassical) approach [15].

Harrow and Montanaro [40] state that "we do not require that the computational task[11] is of practical interest". This is a strong assumption in itself which is adequate only for a foundational study.

Table 1 in [40], p. 205, lists seven plausible approaches to quantum computational supremacy: factoring, single photons passing through a linear-optical network (boson sampling), quantum circuits on many qubits and only a few layers of quantum gates (low-depth circuits), random quantum circuits containing gates that either all commute or do not commute (instantaneous quantum polynomial-time, IQP), quantum approximate optimisation algorithms (QAOA), quantum adiabatic optimisation and quantum analogue simulation. These approaches are then evaluated according to usefulness, assumption implying no classical simulation and difficulties to solve on a quantum computer and to verify. Factoring is the only useful problem, simulation is often useful, adiabatic optimisation could be useful and the remaining three problems do not seem to be useful. Factoring is the hardest to solve on a quantum computer, boson sampling, adiabatic optimisation and analogue simulation are easy and the remaining three are moderately difficult. Only factoring is easy to verify. The complexity-theoretic assumptions are generally very strong, assessing their plausibility is a very difficult task and, generally, conclusions are rather controversial. A detailed complexity-theoretic analysis of various possible quantum supremacy experiments can be found in [11]. The papers [11,40] are exceptionally singular in offering balanced and more formal analyses.

[11] Their formulation for what we call a computational problem.

5 Is the quest for quantum computational supremacy worthwhile?

Apart publicity and marketing, is the effort of demonstrating the quantum computational supremacy justified? What are the (possible) benefits? Can the claim of quantum computational supremacy be falsified?

We will start with the second question. The main benefit could be foundational and philosophical: a better understanding of the nature of quantum mechanics through its computational capabilities.[12] Such a gain will boost the efforts of not only building larger-scale quantum computers but also, and, more importantly, developing new and powerful algorithms for these machines possibly leading to solutions to important practical problems. From this perspective the answer to the first question is affirmative.

Let us examine closer the foundational gain. A successful quantum supremacy experiment could be a complement to Bell experiment: the later refuted local hidden models of quantum mechanics, while the former *seems* to invalidate the Extended Church-Turing Thesis [64]. The paper [40] discusses the advantages of a successful quantum supremacy experiment, even one that barely surpasses any classical competitor, illustrated with hard-to-simulate classical systems like protein folding or fluid dynamics. Here we suggest a different perspective which motivated the tentative formulation above. The Extended Church-Turing Thesis—which incidentally has nothing to do with neither Church nor Turing—is a foundational principle of classical complexity theory which ensures that the polynomial time class **P** is well defined.[13] The Thesis places strong constraints, one of them being that *the model of computation is digital*. For example, analog computers are excluded because they assume infinite arithmetic precision. Furthermore, it is known that an infinite precision calculator with operations +, x, =0?, can factor integers in polynomial time (for a powerful such an algorithm see [62]).[14] But, are quantum computers a "reasonable" model of computation? Are quantum systems digital? At first glance quantum computers (and, more generally, quantum systems) appear to be analog devices, since a quantum gate is described by a unitary transformation, specified by complex numbers; a more in-dept analysis is still required.

What does it take to refute the claim of quantum computational supremacy? This amounts to prove that any computation performed by any quantum computer can be simulated by a classical machine in polynomial time, a weaker form of the Extended Church-Turing Thesis. This statement cannot be proved for the same reasons the Church-Turing Thesis cannot be proved: obviously, they may be disproved. The paper [51] presents efficient classical boson sampling algorithms and a theoretical analysis of the possibility of scaling boson sampling experiments; it concludes that "near-term quantum supremacy via boson sampling is unlikely".

6 Google quantum computational supremacy

In the landscape of various proposals for quantum computational supremacy experiments Google's approach is not only well documented, but had chances to be completed really very soon [49]. The proposed experiment is not about solving a problem: it is the computational task of sampling from the output distribution of pseudo-random quantum circuits built from a universal gate set. This computational task is difficult because as the grid size increases, the *memory needed to store everything increases classically exponentially.*[15] The required memory for a $6 \times 4 = 24$–qubit grid is just 268 megabytes, less than the average smartphone, but for a $6 \times 7 = 42$–qubit grid it jumps to 70 terabytes, roughly 10,000 times that of a high-end PC. Google has used Edison, a supercomputer housed by the US National Energy Research Scientific Computing Center and ranked 72 in the Top500 List [2], to simulate the behaviour of the grid of 42 qubits. The classical simulation stopped at this stage because going to the next size

[12] A beautiful result regarding the computational power of algorithmic random strings was proved in [30]. This was used as a test of quality for quantum randomness in [28].

[13] The Thesis equating feasible computation with polynomial-time computation has significantly less "evidence" than the Church-Turing Thesis; in fact, according to [33], it "lacks evidence".

[14] The quantum version of analogue computers, continuous-variable quantum computers, have been theoretically studied [43]; the model in [61] offers a universal gate set for both qubits and continuous variables.

[15] But, do we *really need* to store everything?

up *was thought to be currently impossible: a 48-qubit grid would require 2,252 petabytes of memory, almost double that of the top supercomputer in the world.* The path to quantum computational supremacy was obvious: if Google could solve the problem with a 50–qubit quantum computer, it will have beaten every other computer in existence.

The abstract of the main paper describing the theory behind the experiment [22] reads:[16]

> A critical question for the field of quantum computing in the near future is whether quantum devices without error correction can perform a well-defined computational task beyond the capabilities of state-of-the-art classical computers, achieving so-called quantum supremacy. *We study the task of sampling from the output distributions of (pseudo-)random quantum circuits, a natural task for benchmarking quantum computers. Crucially, sampling this distribution classically requires a direct numerical simulation of the circuit, with computational cost exponential in the number of qubits.* This requirement is typical of chaotic systems. *We extend previous results in computational complexity to argue more formally that this sampling task must take exponential time in a classical computer.* We study the convergence to the chaotic regime using extensive supercomputer simulations, modeling circuits with up to 42 qubits—the largest quantum circuits simulated to date for a computational task that approaches quantum supremacy. We argue that while chaotic states are extremely sensitive to errors, quantum supremacy can be achieved in the near-term with approximately fifty superconducting qubits. We introduce cross entropy as a useful benchmark of quantum circuits which approximates the circuit fidelity. We show that the cross entropy can be efficiently measured when circuit simulations are available. *Beyond the classically tractable regime, the cross entropy can be extrapolated and compared with theoretical estimates of circuit fidelity to define a practical quantum supremacy test.*

Google was on track to deliver before the end of the year. Alan Ho, an engineer in Google's quantum AI lab, revealed the company's progress at a quantum computing conference in Munich, Germany. According to [56]:

> His team is currently working with a 20–qubit system that has a "two–qubit fidelity" of 99.5 per cent—a measure of how error-prone the processor is, with a higher rating equating to fewer errors. For quantum supremacy, Google will need to build a 49–qubit system with a two–qubit fidelity of at least 99.7 per cent. Ho is confident his team will deliver this system by the end of this year.

Let us note that many, if not most, discussions about quantum computational supremacy focus on the most exciting possibilities of quantum computers, namely the upper bound. What about the lower bound? The article [22] refers cautiously to the lower bound in the abstract: "We extend previous results in computational complexity *to argue more formally* that this sampling task must take exponential time in a classical computer." Indeed, they do not claim to have a proof for the lower bound, just a "better formal argument". Their argument is reinforced later in the introduction:

> State-of-the-art supercomputers cannot simulate universal random circuits of sufficient depth in a 2D lattice of approximately 7×7 qubits with any known algorithm and significant fidelity.

Does Google's experiment satisfy the criteria discussed in Section 4? The computational problem is well-defined, albeit a simulation, not a computational problem,[17] the quantum algorithm solving the problem will run on a quantum computer—promised to be built before the end of 2017[18]—capable of dealing with noise and imperfections, the classical competitor would be allowed a reasonable amount of computational resources and there is a plausible verification. The weakest part comes from the complexity-theoretic assumption [22]:

[16] Our emphasis.

[17] One could argue that the task itself is rather uninteresting and without obvious applications. Indeed, all the time nature is doing quantum 'things' that we don't know how to solve classically. For example, the structure of atoms can in general only be determined experimentally, but nature manages it with near perfect fidelity. If Google achieved the goal—an un-disputable big technical feat—the meaning of the achieved "supremacy" could still be debatable.

[18] "When pressed for an update, a spokesperson [for Google] recently said that 'we hope to announce results as soon as we can, but we're going through all the detailed work to ensure we have a solid result before we announce'. [18], 24 January 2018. The goal was not reached as of 24 February 2018.

> **Memory assumption**. Sampling this distribution classically requires a direct numerical simulation of the circuit, with computational cost exponential in the number of qubits

The assumption was corroborated by the statement:

> Storing the state of a 46–qubit system takes nearly a petabyte of memory and is at the limit of the most powerful computers. [49].

7 IBM challenge

The Memory assumption is crucial for the proposed lower bound, and, indeed, this was confirmed very soon. The paper [52] proved that a supercomputer can simulate sampling from random circuits with low depth (layers of gates) of up to 56 qubits.

> With the current rate of progress in quantum computing technologies, 50–qubit systems will soon become a reality. To assess, refine and advance the design and control of these devices, one needs a means to test and evaluate their fidelity. This in turn requires the capability of computing ideal quantum state amplitudes for devices of such sizes and larger. In this study, we present a new approach for this task that significantly extends the boundaries of what can be classically computed. We demonstrate our method by presenting results obtained from a calculation of the complete set of output amplitudes of a universal random circuit with depth 27 in a 2D lattice of 7×7 qubits. We further present results obtained by calculating an arbitrarily selected slice of 237 amplitudes of a universal random circuit with depth 23 in a 2D lattice of 8×7 qubits. Such calculations were previously thought to be impossible due to impracticable memory requirements. *Using the methods presented in this paper, the above simulations required 4.5 and 3.0 TB of memory, respectively, to store calculations, which is well within the limits of existing classical computers.*

Better results have been quickly announced, see for example [23]. The limits of classical simulation are not only (yet) known, but hard to predict.

In spite of this, IBM has announced a prototype of a 50–qubit quantum computer, stating that it “aims to demonstrate capabilities beyond today’s classical systems" with quantum systems of this size [3].

8 Latest developments

At 2018 Consumer Electronics Show in Las Vegas, Intel CEO Brian Krzanich reported “the successful design, fabrication and delivery of a 49-qubit superconducting quantum test chip” [44]. The 49-qubit superconducting quantum test chip is called “Tangle Lake” after a chain of lakes in Alaska known for extreme cold temperatures. At the event, Mike Mayberry, managing director of Intel Labs said: “We expect it will be five to seven years before the industry gets to tackling engineering-scale problems, and it will likely require 1 million or more qubits to achieve commercial relevance.” In [54] John Preskill aptly said: “Quantum computers with 50-100 qubits may be able to perform tasks which surpass the capabilities of today’s classical digital computers, but noise in quantum gates will limit the size of quantum circuits that can be executed reliably. . . . Quantum technologists should continue to strive for more accurate quantum gates and, eventually, fully fault-tolerant quantum computing.” Jay Gambetta, from IBM Thomas J. Watson Research Center believes that “a universal fault-tolerant quantum computer, which has to use logical qubits, is still a long way off”, [18].

9 Instead of conclusions

Does the paper [52] destroy the quest for quantum computational supremacy? Is there any incompatibility between the classical simulation reported in [52] and the IBM statement cited at the end of Section 7? Tentatively we answer with no both questions. The following paragraph [10] is relevant:

> This paper[19] does not undercut the rationale for quantum supremacy experiments. The truth, ironically, is almost the opposite: it being possible to simulate 49–qubit circuits using a classical computer is a precondition for Google's planned quantum supremacy experiment, because it's the only way we know to check such an experiment's results! The goal, with sampling-based quantum supremacy, was always to target the "sweet spot," which we estimated at around 50 qubits, where classical simulation is still possible, but it's clearly orders of magnitude more expensive than doing the experiment itself. If you like, the goal is to get as far as you can up the mountain of exponentially, conditioned on people still being able to see you from the base. Why? Because you can. Because it's there.[20] Because it challenges those who think quantum computing will never scale: explain this, punks! But there's no point unless you can verify the result.

Here are two more lessons. The first is not to underestimate the importance of mathematical modelling and proving (lower bounds, in particular). As the title of the blog [10] says, "2^n is exponential, but 2^{50} is finite", the difference between exponential and polynomial running times is asymptotic and in some concrete cases it is a challenge to find finite evidence for the difference. Furthermore, proving that a problem is in **P** itself is not a guarantee that there is an algorithm in **P** that is practically useful: primality has been known to be in **P** since 2002, but all known deterministic algorithms are too slow in practice, so probabilistic tests of primality continue to be used.

Secondly, the conversation on quantum computing, quantum cryptography and their applications needs an infusion of modesty (if not humility), more technical understanding and clarity as well as less hype. Raising false expectations could be harmful for the field.

Finally, the race quantum vs. classical is running so fast—a sample is given by the references posted/published since October 2017, the month when the paper [52] was posted, cited in the bibliography—that by the time this paper will be printed some of the results discussed could be obsolete.

Dedication

This paper is dedicated to the memory of Jon Borwein (1951–2016) whose broad mathematical interests included also quantum computing.

Acknowledgment

We thank N. Allen for fruitful discussions, specifically for insight on Feynman's paper [35], and R. Brent, R. Goyal, R. Hua, K. Svozil and an anonymous referee for critical comments and suggestions. This work has been supported in part by the Quantum Computing Research Initiatives at Lockheed Martin.

[19] That is, [52].

[20] "It is not the mountain we conquer but ourselves", as Edmund Hillary aptly said.

References

1. Quantum Manifesto. http://qurope.eu/system/files/u7/93056_Quantum%20Manifesto_WEB.pdf, May 2016.
2. Edison supercomputer in TOP 500 ranking. https://www.top500.org/list/2017/06/?page=1, June 2017.
3. IBM builds 50-qubit quantum computer. http://techvibesnow.com/ibm-builds-50-qubit-quantum-computer/, November 2017.
4. Quantum advantage. The Quantum Pontiff, http://dabacon.org/pontiff/?p=11863, April 2017.
5. UK programme on quantum technologies, http://uknqt.epsrc.ac.uk. 2017.
6. D-Wave Systems, https://www.dwavesys.com/d-wave-two-syste, 2017.
7. European flagship quantum programme, https://ec.europa.eu/digital-single-market/en/news/quantum-europe-2017-towards-quantum-technology-flagship, 2017.
8. Quantum Algorithm Zoo, http://math.nist.gov/quantum/zoo/, 2017.
9. S. Aaronson. The limits of quantum. *Scientific American*, March:62–69, 2008.
10. S. Aaronson. Shtetl-Optimized – 2^n is exponential, but 26^{50} is finite. https://www.scottaaronson.com/blog/?p=3512, November, 12 2017.
11. S. Aaronson and L. Chen. Complexity-theoretic foundations of quantum supremacy experiments. Technical Report Report No. 200, Electronic Colloquium on Computational Complexity, 2016.
12. A. A. Abbott. The Deutsch-Jozsa problem: De-quantisation and entanglement. *Natural Computing*, 11(1):3–11, 2011.
13. A. A. Abbott. De-quantisation of the quantum Fourier transform. *Applied Mathematics and Computation*, 291(1):3–13, 2012.
14. A. A. Abbott and C. S. Calude. Limits of Quantum Computing: A Sceptic's View. Quantum for Quants, http://www.quantumforquants.org/quantum-computing/limits-of-quantum-computing/, June 2016 (presented by Jon Borwein).
15. A. A. Abbott, C. S. Calude, M. J. Dinneen, and R. Hua. A hybrid quantum-classical paradigm for embedded quantum annealing algorithms. CDMTCS Report Series 520, 35 pp., February 2018.
16. W. Ackermann. On hilbert's construction of the real numbers. *Mathematische Annalen*, 99:118, 1928.
17. N. Allen. Email to C. S. Calude. 19 November 2017.
18. P. Ball. The era of quantum computing is here. Outlook: Cloudy, Quanta Magazine. https://www.quantamagazine.org/the-era-of-quantum-computing-is-here-outlook-cloudy-20180124, January 2018.
19. N. J. Beaudry and R. Renner. An intuitive proof of the data processing inequality. *Quantum Info. Comput.*, 12(5-6):432–441, May 2012.
20. H. Bernien, S. Schwartz, A. Keesling, H. Levine, A. Omran, H. Pichler, S. Choi, A. S. Zibrov, M. Endres, M. Greiner, V. Vuletić, and M. D. Lukin. Probing many-body dynamics on a 51-atom quantum simulator. *Nature*, 551:579 EP –, 11 2017.
21. E. Bernstein and U. Vazirani. Quantum complexity theory. In *Proceedings of the 25th Annual ACM Symposium on Theory of Computing, San Diego, California, May 16-18, 1993*, pages 11–20. ACM Press, 1993.
22. S. Boixo, S. V. Isakov, V. N. Smelyanskiy, R. Babbush, N. Ding, Z. Jiang, M. J. Bremner, J. M. Martinis, H. N. S. Boixo, S. V. Isakov, V. N. Smelyanskiy, R. Babbush, N. Ding, Z. Jiang, M. J. Bremner, J. M. Martinis, and H. Neven. Characterizing quantum supremacy in near-term devices. arXiv:1608.00263 [quant-ph], April 2017.
23. S. Boixo, S. V. Isakov, V. N. Smelyanskiy, and H. Neven. Simulation of low-depth quantum circuits as complex undirected graphical models. https://arxiv.org/pdf/1712.05384.pdf, January 2018.
24. D. E. Browne. Efficient classical simulation of the quantum Fourier transform. *New Journal of Physics*, 9(5):146, May 2007.
25. C. Calude. Super-exponentials nonprimitive recursive, but rudimentary. *Inf. Process. Lett.*, 25(5):311–316, 1987.
26. C. S. Calude. De-quantizing the solution of Deutsch's problem. *International Journal of Quantum Information*, 5(3):409–415, 2007.
27. C. S. Calude, E. Calude, and M. J. Dinneen. Adiabatic quantum computing challenges. *ACM SIGACT News*, 46(1):40–61, March 2015.
28. C. S. Calude, M. J. Dinneen, M. Dumitrescu, and K. Svozil. Experimental evidence of quantum randomness incomputability. *Phys. Rev. A*, 82(2):022102, Aug 2010.
29. C. S. Calude, M. J. Dinneen, and R. Hua. Qubo formulations for the graph isomorphism problem and related problems. *Theoretical Computer Science*, 2017, https://doi.org/10.1016/j.tcs.2017.04.016.
30. G. J. Chaitin and J. T. Schwartz. A note on Monte Carlo primality tests and algorithmic information theory. *Communications on Pure and Applied Mathematics*, 31(4):521–527, 1978.
31. T. M. Cover and J. A. Thomas. *Elements of Information Theory*. John Wiley & Sons, New York, 1991.
32. P. L. Daniel J. Bernstein, Nadia Heninger and L. Valenta. Post-quantum RSA. https://cr.yp.to/papers/pqrsa-20170419.pdf, 2017.
33. M. Davis. Interview with Martin Davis. *Notices Amer. Math. Soc.*, 55(560–571), May 2008.

34. D. Deutsch. Quantum theory, the Church-Turing principle and the universal quantum computer. *Proceedings of the Royal Society of London. Series A, Mathematical and Physical Sciences (1934-1990)*, 400(1818):97–117, 1985.
35. R. P. Feynman. Simulating physics with computers. *International Journal of Theoretical Physics*, 21:467–488, 1982.
36. M. Fürer. Solving NP-Complete problems with quantum search. In E. S. Laber, C. Bornstein, L. T. Nogueira, and L. Faria, editors, *LATIN 2008: Theoretical Informatics*, volume 4957 of *LNCS*, pages 784–792. Springer Berlin Heidelberg, 2008.
37. R. Griffiths and C. Niu. Semiclassical Fourier transform for quantum computation. *Physical Review Letters*, 76(17):3228–3231, January 1996.
38. L. K. Grover. A fast quantum mechanical algorithm for database search. In *Proceedings of the Twenty-Eighth Annual ACM Symposium on the Theory of Computing*, pages 212–219. ACM Press, 1996.
39. J. Gruska. *Quantum Computing*. McGraw-Hill, London, 1999.
40. A. W. Harrow and A. Montanaro. Quantum computational supremacy. *Nature*, 549(7671):203–209, 09 2017.
41. D. Ignatius. *The Quantum Spy*. W. W. Norton, New York, 2018.
42. G. Kalai. How quantum computers fail: Quantum codes, correlations in physical systems, and noise accumulation. arXiv:1106.0485 [quant-ph], June 2011.
43. V. M. Kendon, K. Nemoto, and W. J. Munro. Quantum analogue computing. *Philosophical Transactions of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, 368(1924):3609–3620, 2010.
44. B. Krzanich. 2018 CES: Intel Advances Quantum and Neuromorphic Computing Research. https://newsroom.intel.com/news/intel-advances-quantum-neuromorphic-computing-research/, January 2018.
45. M. Lanzagorta and J. K. Uhlmann. Hybrid quantum-classical computing with applications to computer graphics. In *ACM SIGGRAPH 2005 Courses*, SIGGRAPH '05, New York, NY, USA, 2005. ACM.
46. Y. I. Manin. Vychislimoe i nevychislimoe [Computable and Noncomputable] (in Russian). Sov. Radio. pp. 13–15. (Checked 30 November 2017). http://www.worldcat.org/title/vychislimoe-i-nevychislimoe/oclc/11674220, 1980.
47. M. Mohseni, P. Read, H. Neven, S. Boixo, V. Denchevand, R. Babbushand, A. Fowler, V. Smelyanskiy, and J. Martinis. Commercialize early quantum technologies. *Nature*, 543:171–174, March 2017.
48. A. Montanaro. Quantum algorithms: an overview. *Npj Quantum Information*, 2:15023 EP –, 01 2016.
49. C. Neill, P. Roushan, K. Kechedzhi, S. Boixo, S. V. Isakov, V. Smelyanskiy, R. Barends, B. Burkett, Y. Chen, Z. Chen, B. Chiaro, A. Dunsworth, A. Fowler, B. Foxen, R. Graff, E. Jeffrey, J. Kelly, E. Lucero, A. Megrant, J. Mutus, M. Neeley, C. Quintana, D. Sank, A. Vainsencher, J. Wenner, T. C. White, H. Neven, and J. M. Martinis. A blueprint for demonstrating quantum supremacy with superconducting qubits. arXiv:1709.06678 [quant-ph].
50. M. J. Nene and G. Upadhyay. Shor's algorithm for quantum factoring. In R. K. Choudhary, J. K. Mandal, N. Auluck, and H. A. Nagarajaram, editors, *Advanced Computing and Communication Technologies: Proceedings of the 9th ICACCT, 2015*, pages 325–331, Singapore, 2016. Springer Singapore.
51. A. Neville, C. Sparrow, R. Clifford, E. Johnston, P. M. Birchall, A. Montanaro, A. L. A. Neville, C. Sparrow, R. Clifford, E. Johnston, P. M. Birchall1, A. Montanaro4, and A. Laing. No imminent quantum supremacy by boson sampling. https://arxiv.org/pdf/1705.00686.pdf, May 2017.
52. E. Pednault, J. A. Gunnels, G. Nannicini, L. Horesh, T. Magerlein, E. Solomonik, and R. Wisnieff. Breaking the 49-qubit barrier in the simulation of quantum circuits. https://arxiv.org/abs/1710.05867, October 2017.
53. J. Preskill. Quantum computing and the entanglement frontier. In H. M. Gross, D. and A. Sevrin, editors, *The Theory of the Quantum World*, pages 63–80, Singapore, November 10 2012. World Scientific Publishing. arXiv:1203.5813 [quant-ph].
54. J. Preskill. Quantum computing in the NISQ era and beyond. https://arxiv.org/abs/1801.00862, January 2018.
55. J. Prreskill. BES Roundtable on Quantum Comoputing Opportunities in Chemical and Materials Sciences. http://www.theory.caltech.edu/~preskill/talks/DOE_BES_2017_Preskill.pdf, 31 October 2017.
56. M. Reynolds. Google on track for quantum computer breakthrough by end of 2017. https://www.newscientist.com/article/2138373-google-on-track-for-quantum-computer-breakthrough-by-end-of-2017/, June 2017.
57. P. W. Shor. Algorithms for quantum computation: discrete logarithms and factoring. In *Proceedings of the 35th Annual Symposium of on Foundations of Computer Science, Santa Fe, NM, Nov. 20-22, 1994*. IEEE Computer Society Press, November 1994. arXiv:quant-ph/9508027.
58. P. W. Shor. Why haven't more quantum algorithms been found? *J. ACM*, 50(1):87–90, Jan. 2003.
59. M. Sipser. *Introduction to the Theory of Computation*. International Thomson Publishing, 1st edition, 1996, 2013 (3rd ed.).
60. K. Svozil. Quantum hocus-pocus. *Ethics in Science and Environmental Politics (ESEP)*, 16(1):25–30, 2016.
61. S. Takeda and A. Furusawa. Universal quantum computing with measurement-induced continuous-variable gate sequence in a loop-based architecture. *Phys. Rev. Lett.*, 119:120504, Sep 2017.
62. V. Tamma. Analogue algorithm for parallel factorization of an exponential number of large integers: Ii—optical implementation. *Quantum Information Processing*, 15(12):5243–5257, Dec 2016.
63. K. Wiesner. The careless use of language in quantum information. https://arxiv.org/abs/1705.06768, May 2017.

64. A. C.-C. Yao. Classical physics and the Church-Turing Thesis. *Journal of the ACM (JACM)*, 50(1):100–105, January 2003.
65. J. Zhang, G. Pagano, P. W. Hess, A. Kyprianidis, P. Becker, H. Kaplan, A. V. Gorshkov, Z. X. Gong, and C. Monroe. Observation of a many-body dynamical phase transition with a 53-qubit quantum simulator. *Nature*, 551:601 EP –, 11 2017.