Circuits, Parallelism, and Quantum Advantage

COMP/MATH 4803 Final Project Report

Rebecca Kempe

1 Introduction

1.1 What is Quantum Advantage?

Quantum computing is an emerging paradigm of computation that significantly rose in interest in the 1980s and 1990s. In 1982, Richard Feynmann gave a talk in which he reasoned that classical computers could not efficiently perform calculations describing quantum phenomena, and suggested that building computers that performed computations based on quantum mechanics could perhaps avoid those problems. While the idea of merging quantum mechanics and information theory had been around since the 1970s, Feynmann's talk increased interest and attention to these ideas [8, 16]. In 1994, Peter Shor discovered a quantum algorithm that could efficiently factor large integers into primes, and Lov Grover showed that search problems could be sped up on a quantum computer in 1995 [16]. Both of these innovations brought a dramatic increase in interest to the field, and the implications of Shor's algorithm on the security of online transactions and encrypted communications (or possibly, future lack thereof) are wide-reaching [8, 16].

The race towards advancements in quantum technologies is spurred forward by a hope that they will engender massive real-world increases in computational power. One driving factor is the fact that Moore's law, which (roughly) states that the power of classical computers should double about every two years [16], is slowly becoming obsolete due to physical limitations and costs—in fact, some experts consider Moore's law to already be obsolete [17, 20, 21]. The hope is that moving to a new paradigm of computing will enable the progression of advancements in the speed of computation to continue.

Two terms that are often thrown around while discussing advancements in quantum technology are "quantum advantage" and "quantum supremacy." Like many terms associated with emerging technologies, these are ill-defined buzzwords that have varying meanings in different contexts, and which are sometimes used interchangeably. A variety of definitions can be found in [6, 7, 12, 14, 18, 19]. The most common definitions used are that quantum supremacy is an experimental demonstration of a quantum computer solving a problem that is intractible for any classical computer, whereas quantum advantage would require such a demonstration to involve solving a real-world problem that is relevant for commercial purposes. Recently, the Google Quantum AI lab developed a quantum chip called Willow that they claim will be able to achieve real quantum advantage thanks to advances in error correction [9, 15].

The paper this report discusses, titled "Quantum Advantage with Shallow Circuits" [3], instead uses "quantum advantage" to mean a case of a specific subset of quantum algorithms being able to solve problems that the equivalent class of classical algorithms cannot. In contrast to the more common definitions of quantum advantage discussed above, no real-world, commercially applicable gains are being claimed, and the advantage demonstrated is highly specific and does not apply to the broader case of quantum vs classical computation in general. However, the result is important because it was one of the first major steps to showing that quantum computers may be able to solve more problems than classical computers.

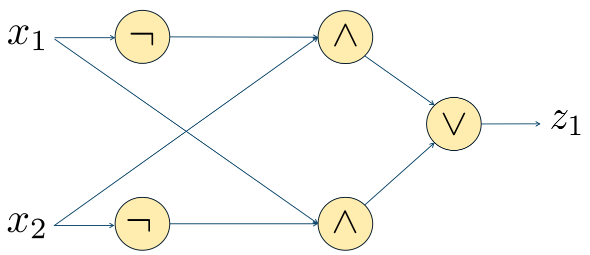

Figure 2.1: A classical circuit that computes $x_1 \oplus x_2$.

1.2 Overview

The remainder of this report will introduce the concepts necessary to understand the results of Bravyi et al. [3]. Their landmark paper proved that constant-depth quantum circuits composed of one- and two-qubit gates (also known as **shallow quantum circuits**, or **SQCs**) can solve a computational problem that constant-depth circuits with bounded fan-in gates cannot solve. Constant-depth circuits with bounded fan-in gates are the class of circuits that solve "problems that can be solved on a parallel machine in a constant time independent of the problem size using a polynomial number of processors" [3]. These problems are exactly problems in the class $\mathbf{NC}^0$ [1]. For brevity, we will abuse terminology and refer to constant-depth circuits with bounded fan-in gates as "$\mathbf{NC}^0$ **circuits**". The main result of [3] is that SQCs can solve the 2D Hidden Linear Function Problem (2D HLFP) whereas $\mathbf{NC}^0$ circuits cannot. In fact, Bravyi et al. prove the following theorem, which states that circuits that solve the 2D HLFP must have logarithmic depth.

Theorem 1.1. *A classical probabilistic circuit with fan-in at most k which solves all size-n instances of the 2D Hidden Linear Function problem with probability greater than $\frac{7}{8}$ must have depth at least $\frac{\log n}{8 \log k}$.*

In the first half of the report, we discuss preliminaries such as classical circuits, quantum circuits, and the relationship between parallelism, $\mathbf{NC}^0$ circuits, and SQCs. In the second half of this report, we discuss hidden linear function problems, introduce the 2D HLFP, and give some insight as to why the 2D HLFP cannot be solved by $\mathbf{NC}^0$ circuits. Finally, we summarize the insights from [3] and point to further reading.

2 Classical Circuits

A circuit (a.k.a. boolean circuit, classical circuit) is a model for **non-uniform** computation. In contrast to the Turing Machine model, which is **uniform**, meaning that the same Turing Machine can be used for every input size, circuits require a different algorithm for each size of input. Circuits compute Boolean functions $f : \{0,1\}^n \to \{0,1\}$, and can be viewed as extensions of boolean formulas. One motivation for studying circuits is the hope that they could make it easier to prove $\mathbf{P} \neq \mathbf{NP}$; however, this has yet to pan out [1]. An example of a classical circuit can be seen in Figure 2.1. We now give several definitions.

Definition 2.1 (Circuit). A classical circuit is a directed acyclic graph with n source nodes and one sink node. The source nodes are inputs, the sink node is the output, and the other (internal) nodes are gates [1]. There are also variations of circuits which compute functions $f : \{0,1\}^m \to \{0,1\}^n$; these circuits have multiple output nodes [3].

Definition 2.2 (Gate, Fan-in, Fan-out). A gate is an internal node in a circuit. Every gate computes a specified boolean function $f : \{0,1\}^k \to \{0,1\}$, where k is the in-degree or "fan-in" of the gate. The "fan-out", or out-degree of a gate corresponds to the number of times its output is copied and used by the rest of the circuit. The typical gate set for classical circuits consists of AND, OR, and NOT gates, though other gates may also be used [1, 3].

Definition 2.3 (Size, Depth). The size of a circuit is the number of gates in the circuit. The depth of a circuit is the number of gates in the longest path from an input to the output in its graph representation [1].

Definition 2.4 (Probabilistic Circuit). Let $\mathcal{C}$ be a circuit with input $x \in \{0,1\}^m$ and output $z = f(x) \in \{0,1\}^n$. A probabilistic circuit is a circuit in which some of the input bits are chosen at random. That is, $x = x'r$ where $r \in \{0,1\}^l$ is a string drawn from an arbitrary random distribution and $x' \in \{0,1\}^{m-l}$ [3].

Definition 2.5 (Correlated Variables). Let x_i be the ith variable (bit) of the input string, z_j be the jth variable of the output string, and let $f : \{0,1\}^m \to \{0,1\}^n$ be the function computed by the circuit. Two variables x_i, z_j are said to be correlated if there exists a string $y \in \{0,1\}^m$ such that flipping the ith bit of y flips the jth bit of $f(y)$. Note that this is only possible if the circuit contains a path from x_i to z_j. [3]

Definition 2.6 (Lightcone). The lightcone of an input bit x_i is the set of all output bits correlated with x_i. Similarly, the lightcone of an output bit z_j is the set of all input bits correlated with z_j.

Definition 2.7 ($\mathbf{NC}^0$ circuit). A circuit computes problems belonging to the class $\mathbf{NC}^0$ if it (1) has constant depth and (2) all gates in the circuit have fan-in at most $k = O(1)$. These are also known as constant-depth circuits with bounded fan-in. For brevity, we will refer to these as $\mathbf{NC}^0$ circuits.

3 Quantum Circuits

3.1 Preliminaries: States, Computations and Quantum Bits

In contrast to classical computers, which perform computations by flipping switches between an “off state” (the “0” bit) and an “on state” (the “1” bit), quantum computers perform computations by manipulating quantum mechanical systems. In short, the input to the computation is a prepared initial quantum state (or a series of linked initial quantum states), the operations are changes to the quantum system, and the output is the state obtained after the final outcome of the quantum computations is measured.

Quantum mechanical systems are modelled by Hilbert spaces; when it comes to quantum computing, we can usually assume that these are finite-dimensional complex vector spaces. A state in a quantum system is modelled by a unit vector. A basis for the system, known as the measurement basis, defines the set of states that can be input to the computation or obtained as output variables post-computation. While intermediate states are linear combinations of measurement basis states, measurement causes each variable to probabilistically collapse down to a single basis state.

Quantum bits, or qubits, are the quantum analogue of classical bits. They are represented by unit vectors, or states, in a 2D complex vector space. The measurement basis, denoted $\{|0\rangle, |1\rangle\}$, is roughly analogous to the pair of classical bits $\{0,1\}$. All qubits are of the form

$$\alpha|0\rangle + \beta|1\rangle,\ |\alpha|^2 + |\beta|^2 = 1,$$

where α and β are complex numbers. The probability of measuring $|0\rangle = |\alpha|^2$, and the probability of measuring $|1\rangle = |\beta|^2$.

3.2 Gates, Universal Gate Sets, and Circuits

In this section, we mainly consider one- and two-qubit gates.

Definition 3.1 (One-Qubit Gate). A single qubit quantum gate is a unitary operator U on a 2D complex vector space. Let $|\phi\rangle$ be an arbitrary qubit. Then the gate U performs the action $|\phi\rangle \to U|\phi\rangle$.

Definition 3.2 (Controlled Qubit Gate). A controlled quantum gate is a two-qubit gate that has a target qubit as a well as a control qubit. The gate performs a unitary action on the target qubit based on the value of the control qubit. These gates are typically denoted CU. Let $|\phi\rangle|\psi\rangle$ be an arbitrary two-qubit state, where $|\phi\rangle$ is the control qubit. Then the gate CU performs the action $|\phi\rangle|\psi\rangle \to |\phi\rangle U^{\phi}|\psi\rangle$.

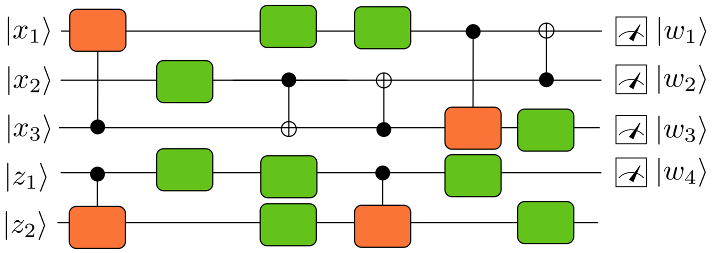

Figure 3.1: A shallow quantum circuit. Green gates are one-qubit gates, orange gates are arbitrary controlled gates, and all other gates are CNOT gates.

In classical computing, all possible operations can be carried out using the set of AND, OR, and NOT gates. This is called a universal gate set. In quantum computing, we have a theorem that states that any unitary gate can be arbitrarily approximated by a finite universal gate set, where all gates are either one-qubit gates or controlled qubit gates [16]. The authors of [3] use a modified Clifford+T gate set, namely the gates

$$H = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix}, \ S = \begin{pmatrix} 1 & 0 \\ 0 & -i \end{pmatrix}, \ T = \begin{pmatrix} 1 & 0 \\ 0 & e^{i\pi/4} \end{pmatrix},$$

and the controlled gate CZ where X, Y, and Z are Pauli matrices.

Definition 3.3 (Quantum Circuit, Depth, Shallow Quantum Circuit). A quantum circuit of depth d consists of a sequence of d layers of one- and two-qubit gates, where each layer of gates consist only of gates which do not share qubits [3]. The depth can also be viewed as the number of timesteps of gates, where two gates can be simultaneously evaluated if their inputs are disjoint and neither gate relies on the other already having been evaluated. If the depth d is constant w.r.t. the input size ($d = k = O(1)$), then the circuit is called a shallow quantum circuit (SQC). An example of an SQC can be seen in Figure 3.1.

3.3 Parallelism and Shallow Quantum Circuits

Bravyi et al. chose to explore SQCs in their work due to the importance of parallelism in both classical and quantum computing. In classical contexts, having a parallel algorithm to solve a problem is highly desirable because it allows us to take advantage of the multiple cores available in modern processors. In quantum contexts, parallelism is also of immense importance due to "coherence time": that is, the amount of time a quantum computation can run without error correction before so much noise builds up that the results are unusable. Quantum computations have constant coherence time, and so in contexts where error-correction is unavailable, it makes sense to parallelize the computation as much as possible [3]. SQCs and $\mathbf{NC}^0$ circuits are natural analogues of each other, which is why it makes sense to compare them.

4 Hidden Linear Function Problems

4.1 Hiding a Linear Function Inside an Oracle

The Hidden Linear Function problem described by Bravyi et al. is a variation of the Bernstein-Vazirani problem, which we will introduce first. In this problem, we "hide a linear function inside an oracle" (we will see what this means shortly), and the goal is to retrieve the value of a secret string z which will help us fully define the function [3].

Setting this up, we have a boolean function $l : \{0,1\}^n \to \{0,1\}$ parameterized by a secret bit string $z \in \{0,1\}^n$, such that for $x \in \{0,1\}^n$, we have $l(x) = z^T x \mod 2$. The goal is to recover the value of z so that we can explicitly specify the function $l(x)$. We have access to an oracle (or black box) which, given a

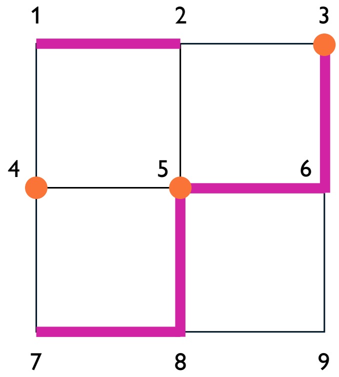

Figure 4.1: A labelled subgraph of a 3×3 grid graph.

value x, will return the value $l(x)$. Bernstein and Vazirani showed that by using a unitary operator which performs the action $|x\rangle \to U_l|x\rangle$, where $U_l|x\rangle = (-1)^{l(x)}|x\rangle$, only one query to the oracle is required, whereas a classical algorithm would require n queries to obtain z.

Bravyi et al. were inspired by this problem, but wanted to move away from the oracular model. This is because in this problem, the quantum improvement is with respect to the oracle, but such improvements do not always correspond to real-world improvements. Instead, Bravyi et al. hide a linear boolean function inside a $\mathbb{Z}_4$-valued quadratic form.

4.2 Hiding a Linear Function Inside a Quadratic Form

In this section, we use the alternative formulation of the Hidden Linear Function Problem (HLFP) given in the seminar by Bravyi [2].

We suppose that A is a binary symmetric matrix of size n, and we consider the nullspace $\mathrm{Ker}(A)$ of A, that is, the set of binary vectors x such that $Ax = 0 \pmod 2$. We now define the quadratic form

$$q(x) = x^T A x \pmod 4.$$

Since x is binary, the restriction of q onto $\mathrm{Ker}(A)$, where we restrict ourselves to inputs from $\mathrm{Ker}(A)$, can be non-zero. (This is in contrast to the real case, where the restriction is a zero function.) What's more, this restriction is a linear function, up to a factor of two: that is, $q(x) = 2 \cdot l(x)$, for some $l : \{0,1\}^n \to \{0,1\}$. Then this can be rewritten as

$$q(x) = 2z^T x \pmod 4,$$

where z is a secret bit string (in this case, it is non-unique) that parameterizes the function.

Thus, the input to the HLFP is a binary symmetric matrix A, and the output is a bit string z such that

$$q(x) = 2z^T x \pmod 4 \text{ for all } x \in \mathrm{Ker}(A).$$

4.3 2D Hidden Linear Function Problem

The 2D HFLP restricts A to be the adjacency matrix of a subgraph of an $n \times n$ grid graph, with the caveat that an edge can be included in the subgraph despite its endpoint vertices not being included. This means that for an $n \times n$ grid graph, we obtain an $n^2 \times n^2$ matrix.

Consider the 3×3 grid graph shown in Figure 4.1. The pink lines are edges included in the subgraph, and the orange dots are vertices included in the subgraph. Then the matrix we obtain as input is the 9×9 matrix

$$A = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

which is binary and symmetric, as required. Bravyi et al. showed that there is an SQC which can solve all instances of the 2D HLFP; however, there is no $\mathbf{NC}^0$ circuit which can solve all instances of the 2D HLFP.

5 Quantum Nonlocality and Quantum Advantage

The reason why 2D HLFP cannot be solved by any $\mathbf{NC}^0$ circuit is due to the strong locality of $\mathbf{NC}^0$ circuits. $\mathbf{NC}^0$ circuits compute local functions: that is, each output bit in an $\mathbf{NC}^0$ circuit can only be correlated with a constant number of input bits. More formally, let $\mathcal{C}$ be a classical circuit and let $\mathcal{L}_{\mathcal{C}}(z_i)$ be the lightcone of an output variable z_i. If $\mathcal{C}$ has depth d and fan-in bounded by $k \in O(1)$, then for all i,

$$|\mathcal{L}_{\mathcal{C}}(z_i)| \leq k^d.$$

However, the input-output correlations of the 2D HLFP exhibit what Bravyi et al. refer to as cycle relations: properties very similar to the GHZ relation, which is a relationship between three highly entangled quantum states. SQCs are able to solve this problem due to quantum nonlocality, which states that the measurement statistics of entangled quantum states cannot be reproduced by local hidden variable models [2, 3]. However, this means that local functions cannot possibly be used to solve instances of these problems.

6 Conclusion

In summary, Bravyi et al. showed that there exists a problem solvable by shallow quantum circuits (SQCs) that is unsolvable by constant depth circuits with unbounded fan-in ($\mathbf{NC}^0$ circuits). This problem is known as the 2D Hidden Linear Function problem, and a circuit which solves this problem must have depth at least logarithmic in n (See Theorem 1.1).

There has since been other work investigating the advantage given by SQCs. We invite the reader to refer to [4, 5, 10, 11, 13] for more recent results surrounding problems solvable by SQCs. There is evidence that suggests that SQCs are more powerful than other classes of circuits in $\mathbf{NC}$, for example, circuits in $\mathbf{NC}^1$ and beyond [10]. Such results are more likely to translate to a true real-world quantum advantage when it comes to parallel algorithms.

References

[1] Sanjeev Arora and Boaz Barak. *Computational Complexity: A Modern Approach.* 1st. USA: Cambridge University Press, 2009. ISBN: 0521424267.

[2] Sergey Bravyi. *Quantum advantage with shallow circuits (Talk).* 2017. URL: https://www.youtube.com/watch?v=xogOLp36GlA.

[3] Sergey Bravyi, David Gosset, and Robert König. "Quantum advantage with shallow circuits". In: *Science* 362.6412 (2018), pp. 308–311. DOI: 10.1126/science.aar3106. eprint: https://www.science.org/doi/pdf/10.1126/science.aar3106. URL: https://www.science.org/doi/abs/10.1126/science.aar3106.

[4] Sergey Bravyi et al. "Quantum advantage with noisy shallow circuits". In: *Nature Physics* 16 (2020), pp. 1040–1045. DOI: https://doi.org/10.1038/s41567-020-0948-z.

[5] Libor Caha, Xavier Coiteux-Roy, and Robert Koenig. *A colossal advantage: 3D-local noisy shallow quantum circuits defeat unbounded fan-in classical circuits.* 2023. arXiv: 2312.09209 [quant-ph]. URL: https://arxiv.org/abs/2312.09209.

[6] Emerging Information & Technology Conference. *Quantum Utility, Advantage and Supremacy.* URL: http://www.eitc.org/research-opportunities/high-performance-and-quantum-computing/quantum-computing-technology-and-networking/quantum-computing-technology/quantum-utility-advantage-and-supremacy.

[7] Chris Ferrie. "Quantum supremacy vs. quantum advantage: which is the best target?" In: *First Principle* (June 27, 2024). URL: https://www.firstprinciples.org/article/quantum-supremacy-vs-quantum-advantage-which-is-the-best-target.

[8] Sara Gamble. "Quantum Computing: What It Is, Why We Want It, and How We're Trying to Get It." In: *Frontiers of Engineering: Reports on Leading-Edge Engineering from the 2018 Symposium.* Ed. by National Academy of Engineering. Washington (DC): National Academies Press (US), 2019, pp. 5–8. URL: https://www.ncbi.nlm.nih.gov/books/NBK538701/.

[9] Google Quantum AI and Collaborators. "Quantum error correction below the surface code threshold". In: *Nature* (2024). DOI: https://doi.org/10.1038/s41586-024-08449-y.

[10] Daniel Grier and Luke Schaeffer. "Interactive shallow Clifford circuits: Quantum advantage against NC^1 and beyond". In: *Proceedings of the 52nd Annual ACM SIGACT Symposium on Theory of Computing.* STOC 2020. Chicago, IL, USA: Association for Computing Machinery, 2020, pp. 875–888. ISBN: 9781450369794. DOI: 10.1145/3357713.3384332. URL: https://doi.org/10.1145/3357713.3384332.

[11] Atsuya Hasegawa and François Le Gall. "Quantum Advantage with Shallow Circuits Under Arbitrary Corruption". In: *32nd International Symposium on Algorithms and Computation (ISAAC 2021).* Ed. by Hee-Kap Ahn and Kunihiko Sadakane. Vol. 212. Leibniz International Proceedings in Informatics (LIPIcs). Dagstuhl, Germany: Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2021, 74:1–74:16. ISBN: 978-3-95977-214-3. DOI: 10.4230/LIPIcs.ISAAC.2021.74. URL: https://drops.dagstuhl.de/entities/document/10.4230/LIPIcs.ISAAC.2021.74.

[12] Nils Herrmann et al. "Quantum utility – definition and assessment of a practical quantum advantage". In: *2023 IEEE International Conference on Quantum Software (QSW).* 2023, pp. 162–174. DOI: 10.1109/QSW59989.2023.00028.

[13] Hsin-Yuan Huang et al. "Learning Shallow Quantum Circuits". In: *Proceedings of the 56th Annual ACM Symposium on Theory of Computing.* STOC 2024. Vancouver, BC, Canada: Association for Computing Machinery, 2024, pp. 1343–1351. ISBN: 9798400703836. DOI: 10.1145/3618260.3649722. URL: https://doi.org/10.1145/3618260.3649722.

[14] Daniel Lidar. "What is quantum advantage? A quantum computing scientist explains an approaching milestone marking the arrival of extremely powerful computers". In: *The Conversation* (Nov. 17, 2023). URL: https://theconversation.com/what-is-quantum-advantage-a-quantum-computing-scientist-explains-an-approaching-milestone-marking-the-arrival-of-extremely-powerful-computers-213306.

[15] Hartmut Neven. "Meet Willow, our state-of-the-art quantum chip". In: *The Keyword* (Dec. 9, 2024). URL: https://blog.google/technology/research/google-willow-quantum-chip/.

[16] Michael A. Nielsen and Isaac L. Chuang. *Quantum Computation and Quantum Information: 10th Anniversary Edition.* 10th. USA: Cambridge University Press, 2011. ISBN: 1107002176.

[17] James R. Powell. "The Quantum Limit to Moore's Law". In: *Proceedings of the IEEE* 96.8 (2008), pp. 1247–1248. DOI: 10.1109/JPROC.2008.925411.

[18] Quantum Zeitgeist. *What is Quantum Advantage?* Apr. 28, 2024. URL: https://quantumzeitgeist.com/what-is-quantum-advantage/.

[19] Quera. *Quantum Advantage*. URL: https://www.quera.com/glossary/advantage.

[20] Wallace Witkowski. "'Moore's Law's dead,' Nvidia CEO Jensen Huang says in justifying gaming-card price hike". In: *MarketWatch* (Sept. 22, 2022). URL: https://www.marketwatch.com/story/moores-laws-dead-nvidia-ceo-jensen-says-in-justifying-gaming-card-price-hike-11663798618.

[21] Audrey Woods. "The Death of Moore's Law: What it means and what might fill the gap going forward". In: *MIT CSAIL Alliances* (). URL: https://cap.csail.mit.edu/death-moores-law-what-it-means-and-what-might-fill-gap-going-forward.