QUANTUM ADVANTAGE ON BLOCKCHAIN TECHNOLOGIES

A THESIS SUBMITTED TO

THE UNIVERSITY OF KENT

IN THE SUBJECT OF COMPUTER SCIENCE

FOR THE DEGREE

OF PHD.

By

Joseph J. Kearney

July 2024

Acknowledgements

I would like to thank both the University of Kent and the Casper Association Academic Grants Program for funding and resources throughout my PhD.

Thank you to my supervisory panel members, Dr Rogério de Lemos, Dr Dominique Chu and Dr Sanjay Bhattacherjee.

I am grateful to Dr Pauline Bernat for encouragement to take on this PhD as well as invaluable advice.

Thank you to my family, Mr John Kearney, Mrs Lorraine Kearney, Mr Alex Kearney, Miss Claire Kearney, and Mrs Eileen Pendlebury for their constant support throughout my academic career.

Miss Dani Luff who jumped into this journey unquestionably and with unwavering support, thank-you.

I would like to also thank my examiners Professor Salvador E. Venegas-Andraca, Dr. Ciara Rafferty and Dr. Vineet Rajani.

I would also like to thank Dr. Jamie Pont for reviewing my thesis, providing helpful feedback and interview preperations.

Thank you to my supervisor Dr Carlos A. Perez-Delgado for constant support and encouragement.

This thesis is dedicated in memory of my grandfather,

John Pendlebury.

Contents

List of Tables

List of Figures

Published Papers

Details of the papers that were published as part of the research presented in this thesis are provided below, in chronological order of publication. First authorship is noted by an asterisk (*). Citation count was obtained from Google Scholar and was accurate as of 05/06/2024.

- * Vulnerability of blockchain technologies to quantum attacks (1)
  - Authors: **Kearney, J.**, Perez-Delgado, C.
  - Array
  - Year: 2021
  - Number of citations: 55
- Quantum advantage on proof of work (2)
  - Authors: Bard, D., **Kearney, J.**, Perez-Delgado, C.
  - Array
  - Year: 2022
  - Number of citations: 15
- * Quantum Blockchain Miners Provide Massive Energy Savings (3)
  - Authors: **Kearney, J.**, Perez-Delgado, C.
  - Under Peer Review

- Year: 2023

- Number of citations: 0

Abstract

This thesis will be a discussion of the intersection of two emerging technologies that have the potential to have a major impact on the world over the coming decades, blockchain technologies and quantum computation.

Blockchain technologies have exploded in popularity since their inception in the early 2010s. There are thousands of different projects and millions of users globally. Cryptocurrencies as a product of blockchains are owned and used by hundreds of millions of people globally.

Quantum devices pose a significant threat to the future of many of our most relied upon cryptographic schemes. This has the potential to impact all aspects of our lives on the internet. While it is a technology still in its infancy, with only small-scale circuits capable of very little real-world application, major investment and research over recent years has accelerated the industries' growth.

Within this thesis, it is shown how quantum computers could dramatically affect the entire blockchain industry. Some blockchains will be left vulnerable in such a way that users' cryptocurrency and ownership of their cryptographic keys could be stolen, while others could be rendered almost completely useless. It will be demonstrated that this has an impact on all blockchains save the few that have

been designed with quantum resistance in mind. While attacks to digital signature schemes such as RSA and ECDSA have been widely described and are well understood, attacks on the Proof-of-Work mechanism blockchains use for gaining consensus were less well understood. It will be shown how within a medium term time frame, PoW could be attacked by a single quantum entity, potentially leading to 51% attacks against some of the most secure blockchain networks.

This thesis also explores the potential positive impacts that quantum devices could have on blockchain technologies. At present, only a handful of near-term, real-world applications exist for small, error-prone quantum hardware. One promising candidate is the concept of quantum cryptocurrency miners—devices analogous to classical ASICs but engineered specifically to mine Proof-of-Work networks. Introducing such miners can boost profitability and create a self-reinforcing feedback loop that attracts even more quantum miners. Furthermore, it will be quantified how their potential effect on the Bitcoin network's energy footprint, demonstrating that widespread deployment could reduce total consumption by up to 99.999%.

Throughout this thesis, it will be hypothesised what the blockchain industry could look like over the next two decades as quantum devices become more accessible and more computationally powerful. This includes both the positives and the negatives that this could bring for the industry. The importance of this is evident as there exists currently over $1 Trillion in market cap of cryptocurrencies. For most of these funds, the underlying ledgers that they are held on are at least in some way vulnerable. Blockchains due to their decentralised governance structure will be slower to move towards post-quantum security. Furthermore, they are entirely reliant and dependent on quantum vulnerable cryptographic primitives.

Chapter 1

Introduction

Blockchains and their corresponding cryptocurrency economies provide a valuable demonstration as to the effects of quantum computers on modern-day infrastructures. The primary motivation is that blockchains are built on a set of rules that are governed by cryptography. Any failure of this cryptography will render the blockchain useless, as the rules of the network that all users must follow can then also be broken. This is even more damaging as blockchains are immutable, once data has been added to the blockchain ledger it can no longer be removed (without breaking the blockchain rules). This is done to ensure there is only one unedited canonical history of all the transactions of data on the blockchain. This ledger and by extension the cryptographic principles it is built upon must be trusted by all users.

Blockchain technologies currently hold and secure vast amounts of wealth, currently exceeding $1 Trillion USD (4; 5; 6). This is protected and transacted by cryptographic protocols. Further to this, blockchains are generally distributed, meaning that no single individual controls the network, or decisions made on the network. This means that the issue of quantum attacks on blockchain technologies must be understood and accepted by a large group of individuals in order for change to be enacted. This work therefore aims to serve as a blueprint for the challenges faced by blockchains in relation to this emerging technology. However, it also describes some of the benefits that could also be possible through the use of quantum devices by honest actors. This thesis discusses at the possibility of increased profits for miners and also a massive reduction in the energy expenditure from the usage of blockchain technologies.

Blockchains were first introduced in 2008, by a person or people unknown, using the pseudonym Satoshi Nakamoto (7). Initially created as an alternative to fiat currencies, it aimed to be utilised as a mechanism for payment without the control or interaction with any central authority or bank. Blockchain technologies have diversified into a popularly used technology across a wide range of industries (8; 9; 10; 11; 12; 13; 14).

The initial goal of this research was to apply the methodology performed by Aggarwal et al. (15), on other blockchains. Aggarwal et al. analysed the vulnerabilities of Bitcoin against quantum attack. In this thesis, the primary vectors for an attack against the Bitcoin cryptocurrency were described. This primarily focused on the use of Shor's algorithm (16) against the digital signature algorithms employed to give ownership to users over their cryptocurrencies. This research demonstrated that Bitcoin is highly vulnerable to these attacks. By investigating the vulnerabilities of other cryptocurrencies, the overall scale of the vulnerabilities for the entire blockchain industry could be analysed. Due to the popularity of Bitcoin, many other blockchains utilise the same technologies. However, a focus on how differing protocols can affect the scale of the vulnerabilities was analysed.

From this research, the impacts of quantum devices on blockchains' consensus algorithms can be studied, specifically Proof of Work (PoW). If a single quantum entity could gain control of the consensus of a blockchain, then the damage that could be done would be more wide-reaching than targeting users' digital signatures. This could lead to the compromise of the entire network. While this kind of attack was considered to be infeasible and of little concern when compared to the attack on digital signatures, it will be demonstrated that this attack is, in fact, possible in a medium term time frame by a single quantum entity.

The concept of quantum cryptocurrency miners will also be presented. This is under the assumption that not every user of quantum computing infrastructure will be a bad actor. This could present an opportunity for increased profit, due to the speed-up gained from running a PoW using a quantum algorithm. This could create incentive for further quantum devices to enter the network.

Finally, other ways in which quantum devices can benefit the blockchain ecosystem will be explored. PoW is (by design) highly energy consumptive. This has long been one of the major criticisms of those blockchains that utilise it for their consensus algorithms. Quantum devices are reversible, meaning that there would be a decrease in the energy consumed when using them on PoW. This thesis will investigate to what extent using quantum devices will decrease the energy consumption of the largest network, Bitcoin.

Throughout this chapter, it will be presented how some of the core principles that are necessary for the remainder of the thesis. Within this, it will be introduced what the cryptographic principles that blockchains utilise are. What are blockchains, and how do they work? Finally, an introduction to quantum computing and the quantum algorithms that are applicable to blockchains is given. Firstly, we will present the current state of the research in this area.

1.1 Scope and Organisation of the Thesis

Scope. This thesis studies how *Cryptographically Relevant Quantum Computers* (CRQC) will impact today's public blockchains. The analysis is limited to protocols that (i) authenticate transactions with elliptic-curve signature schemes (ECDSA, Schnorr, or EdDSA) and (ii) employ consensus mechanisms whose difficulty or energy profile could change under quantum search (with a primary focus on Proof-of-Work, and Proof-of-Stake considered only for comparison). Permissioned or purely classical post-quantum blockchains lie outside the core scope except where they provide useful baselines. This focus allows analysis of both the *risks* (signature forgeries, consensus attacks) and the potential *benefits* (energy savings, profitability) that quantum technology presents to the most widely deployed public chains.

Organisation of the thesis.

Chapter 2 – Contribution and Previous Works formalises the three central contributions of this dissertation, (i) a cross-chain quantum-vulnerability survey, (ii) a Grover-based mining analysis, and (iii) an exploration of quantum hardware for energy efficiency, and positions them against the closest prior literature.

Chapter 3 – Blockchain Background details the five principal blockchains analysed (Bitcoin, Ethereum, Litecoin, Monero and Z-Cash), documenting their cryptographic primitives, address formats and consensus parameters. *Contribution:* a unified data set of protocol specifics that enables later cross-chain comparisons.

Chapter 4 – Quantum Attacks on Blockchains models Shor-style key-recovery and Grover-style search attacks across those blockchains, quantifying signature-breaking timelines and identifying protocol features that accelerate or mitigate risk. *Contribution:* the first analysis spanning multiple chains and attack vectors.

Chapter 5 – Quantum Advantage on Proof of Work develops a profitability model for a miner equipped with a quadratic Grover speed-up, showing how consensus fairness shifts and when single-entity 51 *Contribution:* concrete hardware and difficulty thresholds for destabilising major PoW networks. Introduction to the concept of Quantum Bitcoin Miners

Chapter 6 – QCM Energy Saving applies Landauer-limit calculations and realistic error-correction overheads to estimate the energy consumption of quantum miners, demonstrating orders-of-magnitude savings relative to classical ASICs under several deployment scenarios. *Contribution:* a quantitative argument that quantum mining can reduce PoW energy demand by $\geq 99.999\%$.

Chapter 7 – Conclusion and Future Work summarises findings, discusses limitations (e.g. NISQ-era constraints, adoption barriers) and outlines open research directions for quantum-secure and quantum-enhanced blockchains.

1.2 Cryptographic Principles

When talking about blockchains in relation to quantum computation, the cryptography used by blockchains must be considered. A major implication of quantum computation is their ability to break some of our most well — known and used cryptographic protocols. The areas that will be discussed are hash functions, digital signature schemes and zero-knowledge proofs.

An overview will also be presented on elements of computation complexity that will be critical later in the thesis.

1.2.1 Hash Functions

Hash functions are mathematical algorithms. They take an input, this input is then converted to a series of bytes. For use in cryptography, this output or digest is required to be securely random. Furthermore, two unique inputs should not return the same output. Furthermore, they are one-way functions, trivial to create the digest but computationally very hard to retrieve the input from the digest. The hash functions that will be considered throughout this thesis are assumed to have the following properties:

- **Deterministic** — For every unique input when using the same hash function, the same digest should be the output.

- **Pre-Image Resistant** — This is a critical security feature required of all hash functions. It should be computationally infeasible to reverse a hash function, meaning, given a digest the input should not be retrievable. In short, hash functions are one-way functions.

- **Second Pre-Image Resistant** — Second pre-image resistance is where it is computationally infeasible that given any specific input, a second corresponding input can be found that has the same hash digest. This is similar to collision resistance. However, the difference is that a collision cannot be found for a specific input value.

- **Collision Resistant** — Collision resistance requires it to be computationally difficult to find any two inputs that return the same hash digest.

- **Avalanche Effect** — The avalanche effect is such that changing any part of the input should entirely change the output. This must be to such an extent, that given two digests that originated for two very similar but not identical inputs should be completely discernible from each other.

Some additional desirable properties are that the hash function should accept an input of any size and that it is efficiently computable.

Hash functions have many uses within computer science, including use in data structures such as hash tables. However, for this thesis, it focuses on their uses within cryptography. Hash functions are used to check both the integrity of data and in cryptographic systems such as digital signatures.

Throughout this thesis, when talking about hash functions it is generally referring to SHA (Secure Hash Algorithm) family of cryptographic hashes, mainly SHA-256 (17) and SHA-3(keccak) (18). The algorithm for SHA-256 is based on a Merkle-Damgård construction (19; 20). This involves the process of performing various operations on fixed size message blocks iteratively. SHA-256 works as following:

1. Preprocessing
    - Padding — The input is padded to ensure that it is of some multiple of 512-bits
    - Parsed — The input is split into 512 bit chunks
2. Processing
    - Block splitting — Each 512-bit block is split into 16 separate blocks. These blocks are expanded to an array of 64 32-bit blocks. This is done by various bitwise operations.
    - Initialisation — Eight 32-bit hash variables are initialised using the array of 32-bit blocks

- Main Hashing Loop — 64 rounds of various functions that update the 8 32-bit hash variables.
- Hash Values — An intermediate hash value is calculated by taking the hash variables and adding them to the initial hash variables

3. Post-Processing

   - Final Hash Value — After all the 512-bit blocks have been processed, the final hash value is created by concatenating the eight 32-bit hash variables to give a 256-bit hash digest.

This flow is also shown in figure 1.

SHA-3 part of keccak family of hash functions works differently using a sponge construction (21). SHA-3 hashes are constructed as follows:

1. Initialisation

   - Splitting — The input is split into fixed size blocks
   - Array Initialisation — A 5 by 5 array of 64-bit words is created

2. Absorbing

   - XOR — Each of the message blocks is XORed with part of the Array (rate part)
   - Permutation function — Several rounds of operations are performed. These operations perform bitwise functions.

3. Squeezing

   - Output Hash — The output hash is taken from a portion of the array.
   - If the output is larger than the rate part, then the permutation function is repeated. Once the has is the correct length, it is output as the hash digest.

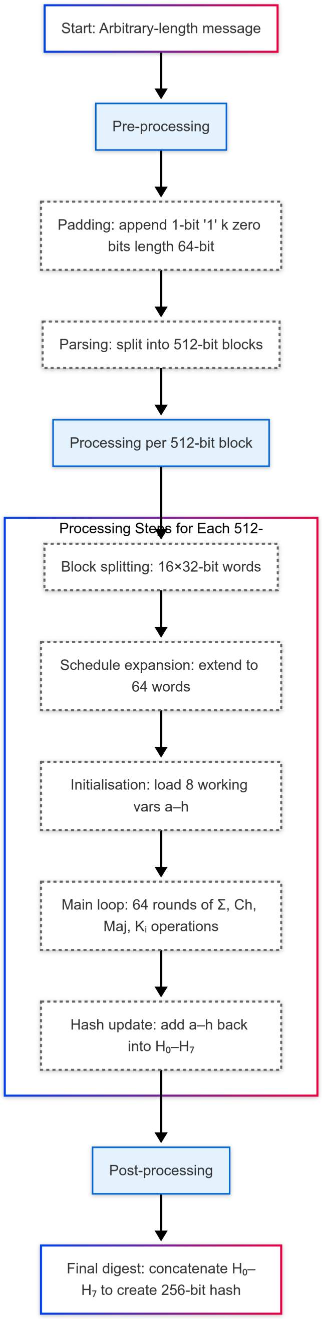
Start: Arbitrary-length message
Pre-processing
Padding: append 1-bit '1' k zero bits length 64-bit
Parsing: split into 512-bit blocks
Processing per 512-bit block
Processing Steps for Each 512-
Block splitting: 16×32-bit words
Schedule expansion: extend to 64 words
Initialisation: load 8 working vars a–h
Main loop: 64 rounds of Σ, Ch, Maj, K_i operations
Hash update: add a–h back into H_0–H_7
Post-processing
Final digest: concatenate H_0–H_7 to create 256-bit hash

Figure 1: Flow-chart of the SHA-256 hashing process. The diagram follows a message from initial padding and block parsing, through the per-block compression loop—comprising schedule expansion, working-variable initialisation, 64 rounds of logical mixing, and intermediate hash updates—and finishes with the concatenation of the eight 32-bit state words to produce the 256-bit digest.

Figure 2 also shows the creation of SHA-3 hash digest.

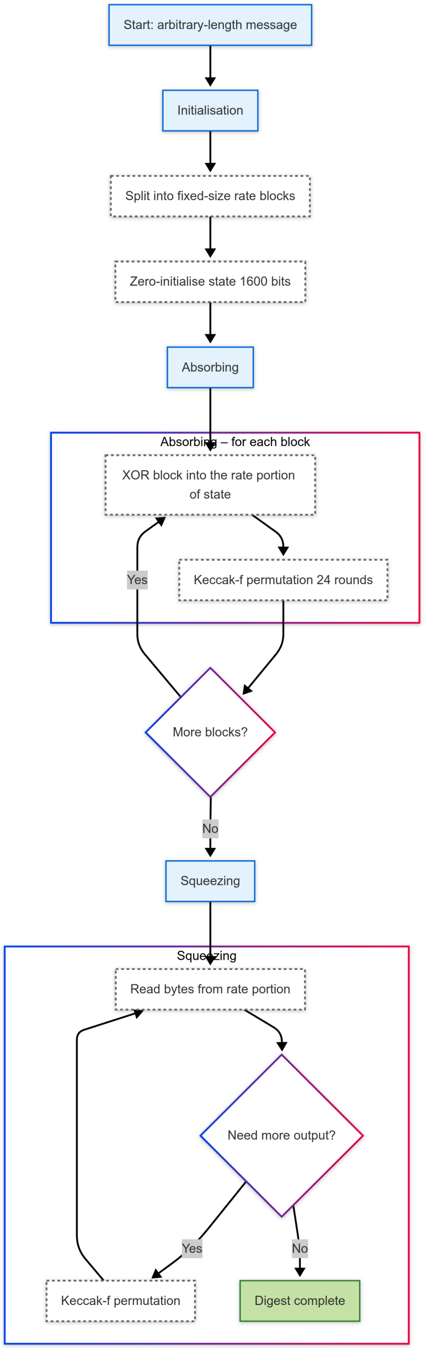
Start: arbitrary-length message
Initialisation
Split into fixed-size rate blocks
Zero-initialise state 1600 bits
Absorbing
Absorbing – for each block
XOR block into the rate portion of state
Keccak-f permutation 24 rounds
Yes
More blocks?
No
Squeezing
Squeezing
Read bytes from rate portion
Need more output?
Yes
No
Keccak-f permutation
Digest complete

Figure 2: Flow-chart of the SHA-3 (Keccak) hashing process. Starting with input-block splitting and a 5 × 5 array (1600-bit state) initialisation, each block is **absorbed** by XORing the rate portion of the state before every round of the Keccak-f permutation. Once all blocks are absorbed, **squeezing** begins: bytes are read from the same rate part to form the output digest; if more output is required, the permutation is invoked again and the process repeats until the desired length is reached.

Other hash functions will differ from the mechanisms laid out here, however hash functions utilised for cryptographic purposes will obey the rule laid out earlier. Both the SHA-256 and SHA-3 hash functions are secure to this day and provide all the requirements that have been laid out.

Hash functions provide the backbone to many Proof-of-Work systems that are used throughout the blockchain industry. This is another feature of hash functions that are important to blockchains, they allow the creation of cryptographic games.

1.2.2 Digital Signatures

Digital signatures are asymmetric cryptographic schemes that allow a user to digitally sign something, proving using their public key that they were, in fact, the one that signed the data (22). It allows a user to provide authentication to the the data that they are sending. Generally, digital signatures have three key parts:

- Key Generation
- Signing
- Verification

Key generation is the step where a user creates a private key and its corresponding public key.

When signing, the user creates a proof that can be verified using the public key that they did in-fact sign a message using a private key. This allows the user to demonstrate that they own the private key without revealing it.

Verification is necessary as external users need to be able to confirm that a signature provided by the user is, in fact, valid. Furthermore, they must do this without knowing anything about the private key.

The most common digital signature scheme is RSA names after Rivest, Shamir and Adleman (23). RSA relies on the hardness of prime factoring. This is given a value, n of the hardness of finding two prime numbers p and q that multiply together to find n. The most efficient classical method to solve this problem is the General Number Field Sieve (24). This solves it at a sub-exponential rate at approximately $O(2^{n^{\frac{1}{3}} \times logn^{\frac{2}{3}}})$.

Key generation for RSA takes place as follows:

1. Two large primes selected p and q
2. Calculate the modulo point $m = p \times q$
3. Calculate $\phi(n) = p - 1 \times q - 1$
4. An exponent is created e such that $1 < e < \phi(n)$. The greatest common divisor of e and $\phi(n)$ must be 1.
5. The private key can be calculated as k where $k \times e \equiv 1 \ mod \ n$

This will give the public key as the pair n and e and the private key as the pair n and k.

Step 3 calculates Euler's Totient Function, which is the number of positive integers that are co-prime, too n.

Once a public private key pair have been generated, the user can digitally sign messages. This is done by:

1. Message $(m)$to be digitally signed must be hashed $(H(m))$. Any hash function that is cryptographically secure as laid out in sub-section 1.2.1 will

work. The signer and the verifier have to both have an understanding of what hashing algorithm should be used. This hash should be considered an integer.

2. The signature is calculated as $S = H(m) \times k \ mod \ n$.
3. m and S are sent to the verifier.

Once the verifier receives the message m and signature, S they can perform the following protocol to verify that the signature is, in fact, valid:

1. Verifier recreates $H(m)$ as $H(m')$ and considers it an integer value.
2. Calculate the decrypted hash value $H(m_d) = S^e \ mod \ n$
3. Check that $H(m_d) = H(m)$
   (a) if true: accept the signature
   (b) else reject the signature

Another example of a digital signature scheme is the Elliptic Curve Digital Signature Algorithm and its variations. These algorithms are more applicable to the remainder of this thesis due to their use in just about every blockchain technology currently in use. These schemes will be explored in section 3.1 where standard ECDSA will be displayed with relation to Bitcoin, as well as some variants, used by blockchains.

Digital Signatures have many uses across the internet where provable identity is necessary. Outside of blockchains digital signatures are utilised by almost everyone that uses the internet on a daily basis, often without the user knowing. Examples of their uses are email security, document authentication, software distribution, financial transaction and healthcare records (25; 26; 27). One of the most integral uses of digital signatures is in TLS for communication across the internet it ensures the privacy, integrity, and authenticity of the data transmitted between two communicating applications, such as a web browser and a web server(28).

1.2.3 Computational Complexity

Computational complexity is a branch of theoretical computer science that focuses on classifying computational problems according to their inherent difficulty and quantifying the resources needed to solve them (29). The primary resources considered are time (how many steps it takes to solve a problem) and space (how much memory it requires). Complexity theory aims to understand the limits of what can be computed efficiently and to identify problems that can be solved within these limits versus those that cannot.

At the core of computational complexity is the classification of problems into complexity classes. The most famous of these classes are P, NP, and NP-complete. Problems in P are those that can be solved in polynomial time by a deterministic Turing machine, meaning they are efficiently solvable. NP problems, on the other hand, are those for which a proposed solution can be verified in polynomial time by a deterministic Turing machine (30). An NP-complete problem is one that is both in NP and as hard as any problem in NP, meaning that if an efficient (polynomial time) algorithm can be found for any NP-complete problem, all NP problems can be efficiently solved.

A major open question in computational complexity theory is the P vs NP problem, which asks whether every problem that can be verified quickly (in polynomial time) can also be solved quickly (in polynomial time) (31). This problem is one of the seven Millennium Prize Problems for which the Clay Mathematics Institute has offered a $1 million prize for a correct solution. The consensus among most computer scientists is that P is not equal to NP, implying that there are problems in NP that are inherently more difficult to solve than to verify, although this has not been proven.

Beyond P and NP, there are other complexity classes that help to map the landscape of computational difficulty. For example, PSPACE consists of problems that can be solved using a polynomial amount of memory, regardless of the time it might take. EXPTIME includes problems solvable in exponential time. These classes help us understand the spectrum of difficulty and what is required for solving different problems. This gives us a more detailed view of how computationally feasible a problem is.

A complexity class that is of interest to us is the BQP (Bounded-Error Quantum Polynomial Time) complexity class. This means that a problem can be efficiently solved by a quantum computer (32). Specifically, a problem is in BQP if there exists a quantum algorithm that can solve it in polynomial time with a probability of error of less than 50% for all computations. It must be noted that this error rate is not referring to the error rate of the device itself, but the error rate of the algorithm. This means that a quantum computer can solve these problems correctly most of the time within a reasonable amount of time. BQP includes problems that are believed to be difficult for classical computers to be solved efficiently.

A key tool in analysing computational complexity is Big O notation, which describes the upper bound on the time complexity of an algorithm in terms of the size of the input (n). Big O notation allows us to express how the runtime of an algorithm grows as the input size increases. This gives us a general estimate on the run time of a problem without considering low level details. For instance, an algorithm that runs in $O(n^2)$ time grows quadratically with the input size, meaning that if the input size doubles, the running time increases fourfold. This notation provides a way to compare the efficiency of different algorithms and is essential for understanding the scalability of computational solutions.

Computation complexity is necessary to understand the impact that the use of quantum algorithms will have on the cryptographic principles utilised in blockchain technologies. By reducing the complexity of a problem it is possible that, the problem that was previously infeasible on a classical device, could, in fact, be solved trivially on a quantum device. Or even a problem that is solvable in a specific period of time on a classical device could be solved far quicker than intended on a quantum device.

1.2.4 Zero-Knowledge Proofs

Zero-Knowledge proofs allow a user to send a proof that demonstrates they possess specific knowledge without the need to reveal the information concerned (33; 34).

Zero-Knowledge proofs rely on the following properties (35):

- **Completeness** — The verifier should be able to confirm the validity of the proof. This is under the condition that the protocol for generation and verification are followed correctly.
- **Soundness** — A dishonest prover cannot trick the verifier if the proof created is false.
- **Zero-Knowledge** — All information of the information being proved should remain completely hidden. The verifier should leave the interaction no more

knowledgeable about the data than at the start, aside from the fact that the prover is honest.

The common metaphor for Zero-Knowledge proofs is the Ali-Baba Cave analogy. In this example, Alice is the prover and Bob is the verifier. Alice has access to a secret key. This secret key within the cave allows her to enter through door A and pass through and exit door B. Without the key, a user is incapable of doing this. To achieve this, Alice enters the cave through either entrance, Bob cannot see this part. Bob will then request Alice to exit through one of the two doors. If the door selected is the same as the one Alice entered through, then there is no need to use the key, she can exit through the same door. However, if the opposite door is called by Bob, if Alice has the secret key, she can pass through the cave and leave by the opposite door. This process can be repeated multiple times by Bob and Alice until such a point that Bob is satisfied that Alice does, in fact, have the key. If Alice fails at any point, Bob can be certain that Alice does not have the key. Bob will continue until such a point that the probability of Alice not having the key is very unlikely. With this example, where the probability of Bob picking the door that Alice entered through being $\frac{1}{2}$ the probability of Alice knowing the secret after 10 successful rounds is $0.999 = 1 - (\frac{1}{2})^{10}$. This would give more than enough confidence that Alice holds the key without knowing anything about the key.

This analogy fulfils the rules of zero knowledge proofs as:

- **Completeness** — Alice can always come out of the correct entrance if she has the key.
- **Soundness** — Alice would only have a probability of $\frac{1}{2}$ of getting a single challenge correct if she does not have the key.

- **Zero-Knowledge** — This series of challenges does not reveal anything about the key, only strengthening the knowledge of Bob whether Alice does have the key or not.

Zero-Knowledge proofs are utilised by some blockchain technologies. This is done to hide the transacted amount of cryptocurrency or even hide the users' pseudonym on the blockchain. While this is not used by most networks, its importance will be shown in Chapter 4.

1.3 Blockchain Technologies

The concept of a digital currency was first introduced by David Chaum in 1983 (36) as eCash. While it is not a blockchain technology, it could certainly be considered the world's first cryptocurrency. It was not until 2008 with the invention of Bitcoin (7) that the blockchain revolution truly began. It is estimated that over a billion people have used cryptocurrencies in some way (37). Blockchains, are distributed ledgers. These ledgers immutably track data with a single true canonical history. Users send transactions that contain some data, generally this can be any data, but most frequently it is the blockchains native cryptocurrency.

There are three groups of blockchains:

- **Public** — Public blockchains are the most commonly discussed, such as Bitcoin and Ethereum. They are created in such a way that all data stored on the chain is immutable. Anybody can access the blockchain, and no one user has ultimate authority over the blockchain.
- **Private** — Private blockchains, widely adopted in industry—, typically function without a native cryptocurrency. Platforms such as R3 Corda and Hyperledger illustrate this design. Because such networks can be configured

and governed by a single organization, a central authority can potentially alter or even remove data from the ledger.

- **Permissioned** — Permissioned blockchains are a hybrid of the two. Examples of which are private instances of public blockchains. Permissioned blockchains may restrict who has access to the network, however will generally keep the core concept of blockchains that once it is deployed, there is not a single entity in charge that is capable of making invalidated changes to the ledger.

For the remainder of this thesis, we will be only talking about public blockchains.

1.3.1 Transactions and Blocks

Data is stored by blockchain in the form of transactions and blocks. Transactions are data structures that allow a user to send data (cryptocurrency) from their account to another user/users. They signal to other members of the network, what they are sending, where they are sending it to and that they are, in fact, allowed to send what they are trying to send.

Miners (can also be called validators) are specialist nodes on the network. These nodes package transactions to the network into blocks. They are responsible for checking that the transactions are valid. Examples of ways in which a transaction may not be valid are, the user does not prove they are eligible to send the data they are, they are trying to send more than is available. Transactions are broadcast by the sender to other nodes on the network.

Details of the structure of these transactions and blocks are provided where relevant in chapter 4.

1.3.2 Proving Identity

Identity and ownership and the ability to prove them are critical aspects of blockchain technologies. With no central authority to check who individuals are, and they are who they say they are, this needs to be done by other means. On blockchains, a users' identity is generally in the form of the users' public key or the hash of that public key.

ECDSA is the most widely used mechanism for generating digital signatures on blockchains. This is chosen due to two reasons. Firstly, this was first implemented by Bitcoin. Due to a lot of technology re-use, this meant that many proceeding blockchains also utilised ECDSA as their digital signature scheme. Secondly, ECDSA provides much smaller signatures than other similar schemes such as RSA (256-bits compared to 2048-bits), while providing the same level of security. Due to the nature of every transaction requiring a signature and there being hundreds, potentially thousands of transactions per block, smaller signatures reduce the demand on nodes holding the ledger locally.

ECDSA relies on the discrete logarithm problem. This problem can be defined as, over a cyclical group G given two values, g and y. Find a value x that satisfies the equation: $g^x mod p = y$. This is a trapdoor problem. Given g and x it is trivial to compute y. However, solving for x is exponentially difficult with an increasing size of the G group. The discrete logarithm problem and its mathematical difficulty will be important throughout this thesis.

1.3.3 Consensus Algorithms

One of the primary problems that arise when designing a blockchain from scratch is how does the blockchain network create trust in the system when users cannot trust each other? This is the major hurdle that previous iterations such as ECash struggled to overcome. The solution to this is consensus algorithms. All public blockchains have to have some mechanism by which the users can trust that every transaction that is added to the chain is, in fact, valid. Table 1 shows some major consensus algorithms that are currently used by public blockchains. However, even from this small set, an extreme majority use Proof-of-Work or Proof-of-Stake. Blockchain consensus algorithms have two primary goals. Firstly to ensure that there is a definite canonical history of the blockchain, and secondly to ensure that a dishonest minority cannot disrupt the network.

A further goal of consensus algorithms is to allow the blockchain to throttle the number of blocks produced. If we consider a situation where any user can produce any block (valid or invalid) at any time, how would any of the user know the truth of the blockchain? As stated previously, blockchains have one canonical history, one thread of blocks. By throttling the number of blocks or restricting who can add a block at a specific block height, this order can be created.

A 51% attack denotes any situation in which a single entity—or a colluding coalition—gains control of a majority of the total consensus power in a blockchain network (e.g., more than half of the cumulative hash-rate in Proof-of-Work, or more than half of the staked weight in Proof-of-Stake). Possessing that majority lets the adversary reliably outpace honest participants in extending the canonical chain, thereby giving it the unilateral ability to reorder recent transactions, exclude selected transactions from inclusion, or perform "double-spend" attacks by first spending coins on the public chain and then privately rewriting history to a longer chain that invalidates the earlier spend. Although the attacker cannot fabricate coins ex nihilo or rewrite blocks buried deeply in the ledger without prohibitive cost, the economic and reputational damage caused by sustained control over consensus—especially loss of payment finality and censorship resistance—undermines the very trust that consensus algorithms are meant to establish.

PoW will be the main mechanism for consensus that will be considered throughout this thesis. It relies on the miner to perform some computationally hard problem for which, in order for the block to be valid, the miner has to produce a proof. By doing this, the miner of the block is rewarded in the blockchain's native token. Solving the problem will cost the miner time and money. The time it takes means that the number of valid blocks are restricted depending on the difficulty of the problem. The capital expended means that the user is incentivised to act honestly, as any badly created blocks will be invalid. PoW will be discussed in detail in chapter 5.

1.3.4 Failures and Successes of Blockchain Technologies

Blockchains and the cryptocurrencies associated with them have been a divisive topic since their inception. Cryptocurrencies were originally conceived as a decentralised alternative to government-backed fiat currencies (44), designed to enable peer-to-peer value transfer without reliance on traditional financial intermediaries. Through the use of pseudonym address, users have some level of anonymity when using cryptocurrencies. This, of course, lends itself well to black market activities. This was not more acute than the use of cryptocurrencies on the Silk Road (45; 46). Silk Road was a website that allowed users to purchase illegal goods anonymously, the primary payment method for these goods was Bitcoin. Lawrence Lessig stated in 2000 (47) that '*Code is Law*'. This has been found to be highly applicable to blockchain technologies. Blockchain protocols have been designed in such a way that there is no one person in control, and that there is also no need to trust any individual on the blockchain. It is, in fact, the code and mathematical functions that blockchains are built on that give users trust in the network. This new trust-less technology, however, does have, like many other technologies, the ability to be misused such as in the case of Silk-Road.

One major criticism that will be discussed extensively in Chapter 6 is the excessive energy consumption of PoW-based blockchain technologies (48; 49; 50). Proof-of-Work is designed to keep users honest by ensuring that they have to perform some hard mathematical problem, which in turn costs computational power, therefore electricity and therefore money. This use of electricity is argued to provide both the security and the value that underpins PoW blockchains.

Despite these negatives, blockchains have proven to be a formidable force in the financial industry as well as many projects trying to use the technology for good. It also allows those that do not have access to banks to have a mechanism for storing funds. They provide a haven for residents of nations where hyperinflation is rampant, thereby preventing the depreciation of their income.

1.4 Quantum Computing

Quantum computers exploit quantum-mechanical phenomena to execute certain algorithms with provable asymptotic speed-ups over the best-known classical counterparts, even though both models can in principle compute the same set of functions. The area of Quantum mechanics and quantum computing is an extremely deep and expansive field. Within this section, brief overview of quantum devices and the functionality that they have above classical devices will be given. However, we will restrict ourselves for this section and the remainder of the thesis to the aspects necessary for quantum computers interactions with blockchains.

1.4.1 Quantum Mechanics and Quantum Information

Quantum mechanics is a branch of physics that deals with the behaviour of particles at the atomic and subatomic levels (51; 52). Quantum mechanics provides a framework for comprehending phenomena that occur on minimal scales, in contrast to classical mechanics, which accurately describes the macroscopic world. Examples of such are the behaviours of electrons, photons, and other fundamental particles (53). The introduction of principles describes these particles exhibiting both wave-like and particle-like properties, depending on how they are observed.

One of the fundamental principles of quantum mechanics is the Heisenberg Uncertainty Principle (54), which asserts that certain pairs of physical properties, such as position and momentum, cannot be precisely known simultaneously. This implies that the more accurately we know the position of a particle, the less accurately we can know its momentum, and vice versa. This uncertainty is not due to measurement errors, but is a fundamental property of nature, reflecting the limitations of our ability to predict the behaviour of quantum systems.

Quantum entanglement is another integral aspect of quantum mechanics (55). When particles become entangled, their states become linked such that the state of one particle instantly influences the state of another, regardless of the distance separating them. This phenomenon has been experimentally verified and has profound implications for information theory and quantum computing. Entanglement challenges our classical understanding of locality and causality and has led to new technological developments, such as quantum cryptography, which promises secure communication channels that are theoretically impervious to eavesdropping.

1.4.2 Superpositions and Quantum Devices

The superposition principle says that a quantum system can exist in multiple states at the same time until it is measured. This is famously illustrated by Schrödinger's cat thought experiment, where a cat in a sealed box can be simultaneously alive and dead until an observer opens the box and observes its state (56). This principle underpins the functionality of quantum computers, which use quantum bits or qubits that can represent both 0 and 1 simultaneously, potentially allowing for massively parallel computations.

In quantum computing, superposition is leveraged to perform parallel computations (57; 55). Because a qubit in superposition can represent multiple states simultaneously, a quantum computer with n qubits can represent 2^n states at once. This exponential growth in representational capacity allows quantum computers to process vast amounts of information more efficiently than classical computers for certain problems. For example, quantum algorithms like Shor's algorithm for factoring large numbers and Grover's algorithm for searching unsorted databases exploit superposition to achieve significant computational speed-ups over their classical counterparts.

The power of superposition is further amplified when combined with the previously discussed quantum principle: entanglement. Entangled qubits exhibit correlations that can be used to solve complex problems more efficiently. In a quantum computer, operations on one entangled qubit instantaneously affect its partner, no matter the distance between them. This interconnectedness, combined with superposition, enables quantum computers to perform complex calculations at unprecedented speeds, offering potential breakthroughs in fields such as cryptography, materials science, and artificial intelligence.

1.4.3 Subgroup Finding Algorithms

Subgroup finding algorithms allow us to find smaller significant subsets from a large group. Shor's algorithm is a quantum subgroup finding algorithm (16). It can factor large integers and solve the discrete logarithm problem in polynomial time. It utilises quantum Fourier transforms and coset sampling (55) to efficiently find hidden subgroups within these problems. This is done in the following steps :

1. Create a superposition of all possible states. Hadamard gates are applied to every qubit in the input register, one qubit per bit of the search space.

2. Perform the function and entanglement. The modular exponentiation function is performed: $f(x) = a^x \; mod \; N$ on the state in super position to create and entangled state. N is the value we aim to find solution for.

3. Apply the quantum Fourier Transform on the input register. This will highlight the periodicity.

4. The state can now be measured and value p can be retrieved. p is then used to derive the factors of N

So this would mean that N is a public key and p while it is not the factors of, N i.e. not the private key. It can be used to trivially find them. This can be used to factor or solve the discrete logarithm problem in $O\left(\log^2 N \log\log N \log\log\log N\right)$. This can be generalised to $O\left(\log^3 N\right)$.

This will have a dramatic impact, as almost every public-key cryptography system utilises problems that can be solved in polynomial time by this algorithm. Examples of the cryptography that will be broken by this are, Elliptic curve cryptography, RSA, ElGamal and Diffie-Hellman. Current predictions suggest that a quantum computer capable of breaking these cryptographic systems could arrive in 2035, just over a decade away (58; 15).

It is needless to say that this will have a dramatic impact on blockchain technologies. This will be discussed further in Chapter 4.

1.4.4 Quantum Search Algorithms

Grover's algorithm is the search algorithm that will be discussed throughout this thesis. It is an amplitude amplification algorithm. This means that it iteratively applies a series of operations. This will over a series of iterations amplify the answer being searched for while decreasing the amplitude for incorrect answers (59). This means over iterations, the probability upon measurement for the answer to be correct increases. This is quite unique in the fact that if the algorithm is stopped early, there is still a chance of the correct state being measured despite the algorithm not running to the end.

The process of Grover's algorithm is as follows:

1. Set up Oracle $\mathfrak{O}$ that marks solutions that fit a given criterion
2. Set up n qubits for the bit size of the search space.
3. Initialize the system to a uniform superposition of all possible values by applying Hadamard transform $H^{\otimes n}$
4. Perform Grover's algorithm iterations, until the probability of finding a valid nonce is approximately 1. Each iteration consists of:
    (a) Apply $\mathfrak{O}$ over the search space
    (b) Apply Hadamard transform $H^{\otimes n}$
    (c) Perform phase shift such that $|0\rangle \to |0\rangle$, $|x\rangle \to -|x\rangle$ for $x > 0$

(d) Apply Hadamard transform $H^{\otimes n}$

The iteration must take place $\sqrt{2^n}$ times to have an approximate probability of 1 of finding the correct answer.

Search spaces are the collective group of all feasible solutions to a given problem. A solution space is the set of correct values. Given an unstructured search space (N), a search algorithm looks for a member of the solution space (M). Using classical infrastructures for most problems, the optimal time complexity is $O(\frac{N}{M})$. Grover's Search Algorithm (60) when ran on a quantum device can perform a quantum search of the search space is $O(\sqrt{\frac{N}{M}})$ steps.

The impact of Grover's algorithm is more subtle than that of Shor's algorithm. While it can not be utilised to break cryptosystems, it can certainly be employed to speed up problems that have no efficient algorithm but are designed to be solved. While this is a very specific use case, it directly applies to the solving of PoW.

1.4.5 Notes on the Current State and the Future of Quantum Devices

Quantum computing is currently in an exciting yet early stage, with significant advancements being made both in theoretical research and practical implementation. Major technology companies such as IBM (61; 62; 63), Google (64), and Microsoft (65; 66; 67) have been at the forefront of this progress. IBM's Quantum Experience and Qiskit framework allow researchers and developers to run quantum algorithms on real quantum hardware via the cloud, democratizing access to quantum computing resources. Google's notable achievement with its Sycamore processor, which demonstrated “quantum supremacy” by solving a specific problem faster than the best-known classical supercomputers, marked a significant milestone in the field.

Despite these advancements, quantum computers are still far from being able to solve practical, large-scale problems. The primary challenge lies in the stability and coherence of qubits, the fundamental units of quantum information. Qubits are highly susceptible to decoherence, a process where quantum information is lost to the surrounding environment, making error correction a critical area of research. Current quantum computers, often referred to as Noisy Intermediate-Scale Quantum (NISQ) devices (68), have a limited number of qubits and are prone to errors, restricting their practical utility to specific types of problems that can tolerate high error rates. Chapter 6 will discuss how these small-scale devices for specific applications could be utilised on blockchain technologies.

Efforts to scale quantum computing are ongoing, with various approaches being explored to increase qubit count and fidelity. Superconducting qubits, trapped ions, and topological qubits are among the leading technologies being developed. For instance, Honeywell (69) and IonQ (70) are making strides with trapped ion technology, which offers advantages in terms of coherence times and connectivity between qubits. Meanwhile, companies like Microsoft are investing in topological qubits, which are theoretically more robust against errors. There are also advancements being made in quantum error correction codes (71; 72; 73) and quantum algorithms, aimed at making quantum computers more reliable and versatile.

Quantum computers are expected to revolutionize many fields, such as materials science (74; 75), by simulating molecular structures that are infeasible for classical computers. Additionally, optimization problems in logistics (76; 77; 78), finance (79; 80; 81), and drug discovery (82; 83) are anticipated to benefit significantly from quantum computational power. Governments and private sectors worldwide are recognizing the strategic importance of quantum computing (84; 85; 86; 87; 88; 89). This will potentially lead to increased funding to accelerate its development and deployment.

Name	**Used By**	**Description**
Proof-of-Work (7)	Bitcoin, Litecoin	Competing miners aim to solve some difficult cryptographic problem. By solving the problem, the miner can publish a block and potentially receive a reward.
Proof-of-Stake (38)	Ethereum 2.0, Cardano	Validators stake their native cryptocurrency. These validators are chosen randomly to publish the next block. Depending on the protocol, the higher the stake potentially may give you a greater chance of being selected. Dishonest or unreliable behaviour may lead to the loss of a portion of your stake.
Delegated Proof of Stake (39)	EOS, Tron	A variation on PoS. Stakers select a group of block producers with their votes. These block producers will then create the blocks.
Practical Byzantine Fault Tolerance (40)	Stellar	Leaders on the network send messages to other nodes on the network so that all nodes come to an agreement on the state of the network.
Delegated Byzantine Fault Tolerance (41)	NEO	Token holders elect consensus nodes. These consensus nodes then validate transactions and publish blocks.
Proof of Authority (42)	VeChain, PoA Network	Trusted nodes are the only members of the network that can publish blocks.
Federated Consensus (43)	Ripple	A pre-selected group of nodes works together to publish blocks to the chain. These nodes are selected depending on how reliable / how honest they are.

Table 1: This table shows a selection of some of the most common consensus algorithms used by public blockchains

Chapter 2

Contribution and Previous Works

2.1 Thesis Contributions

This thesis is among the first to systematically examine both the *threats* that quantum computing poses to blockchain systems and the potential *advantages* that quantum technologies might offer for improving those systems. In doing so, it begins to shine light on the looming quantum vulnerabilities of blockchains while also exploring ways that quantum computing could be leveraged beneficially. The main contributions of this research are as follows:

1. **Broad Analysis of Quantum Vulnerabilities in Blockchains:** The thesis provides a comprehensive assessment of how quantum adversaries could impact a range of existing blockchain protocols. Prior studies have mostly focused on Bitcoin or individual aspects of the problem; in contrast, this work takes a wider view by analyzing multiple major public blockchains (the top five by market capitalization at the time of study, among others) for quantum-related vulnerabilities. It examines the susceptibility of each blockchain's cryptographic primitives (such as digital signature schemes and hashing algorithms) to quantum attacks and highlights which platforms and components are most at risk. This broad survey begins to address the gap in

literature regarding the *industry-wide* implications of quantum computing on blockchain technology.

2. **Investigation of a Novel Quantum Attack on Proof-of-Work:** The thesis investigates a previously underexplored attack vector against Proof-of-Work (PoW) consensus using quantum computing. Specifically, it explores how quantum algorithms (notably Grover's search algorithm) could give malicious miners a computational advantage in Bitcoin's (and similar cryptocurrencies') mining process. While earlier research largely assumed that PoW mining is relatively safe from significant quantum speed-up in the near term, this work re-examines that assumption in depth. It analyzes a scenario in which a quantum-equipped miner could substantially increase its probability of finding new blocks, a threat that has been mostly disregarded in prior literature. By quantifying the potential advantage and its consequences (such as lowered security thresholds or mining centralization risks), the thesis sheds new light on how PoW-based blockchains might be destabilized by quantum-capable adversaries.

3. **Exploration of Quantum Advantages for Blockchain Efficiency:** Beyond threats, this work explores the counterintuitive possibility that quantum computing could be *beneficial* to blockchain systems under certain conditions. In particular, it examines how the deployment of quantum cryptocurrency miners might improve network efficiency by increasing mining throughput and reducing energy consumption for solving PoW puzzles. The analysis considers a future scenario where quantum hardware is widely available to miners: block generation could become more energy-efficient overall, potentially mitigating one of the biggest criticisms of PoW (its high energy usage). This thesis presents what is, to the best of our knowledge, the first quantitative discussion of how integrating quantum computing into

blockchain operations might lead to positive outcomes like energy savings or improved profitability for honest participants. By doing so, it opens a novel line of inquiry into *quantum-enhanced* blockchains, complementing the more common focus on quantum-resistant designs.

Overall, these contributions expand the understanding of blockchain security in the quantum era by not only identifying where and how current systems are vulnerable, but also by considering strategic opportunities that quantum technology might unlock for future blockchain innovation.

2.2 Previous Work

Blockchain security in the presence of quantum computing is a relatively new and underexplored research area. Early discussions on the topic began to emerge around 2015, when Campagna and Chen (90) highlighted the need for quantum-safe cryptography in the context of the Bitcoin ledger. It was already well understood by that time that sufficiently powerful quantum computers would be capable of breaking widely used cryptographic schemes—most notably RSA and the Elliptic Curve Digital Signature Algorithm (ECDSA)—through Shor's factoring algorithm (16). Such knowledge implied a serious threat to blockchain systems, which rely heavily on those cryptographic primitives for securing transactions and identities. Despite this looming threat, however, the literature up to that point remained sparse, and there was a clear gap calling for deeper analysis of how exactly quantum capabilities might compromise blockchain protocols.

The first in-depth academic analyses of blockchain vulnerability to quantum attacks appeared in 2017. Aggarwal et al.(15) and Tomamichel et al.(91) were among the pioneers to rigorously investigate the risks that quantum computing poses to cryptocurrencies. Aggarwal and colleagues, for example, examined Bitcoin's security under quantum adversaries and demonstrated that once large-scale quantum computers become available, they could use Shor's algorithm to swiftly break Bitcoin's ECDSA-based digital signatures. This would enable an attacker to forge transactions (e.g. steal funds by deriving private keys from public addresses), undermining the fundamental trust and security of the blockchain. The authors projected that Bitcoin and similar cryptocurrencies might become vulnerable to such signature-forging attacks within as little as a decade if quantum technology continues to progress at its current pace. Their study also touched on the prospect of quantum-accelerated mining, concluding that Bitcoin's Proof-of-Work puzzle (hash-based mining) is comparatively more resistant to drastic quantum speedups in the near term (because Grover's algorithm provides, at best, a quadratic speedup for brute-force searching). Nevertheless, they noted that a quantum "advantage" in mining could eventually pose a concern as quantum hardware improves. Around the same time, Tomamichel et al. reached similar conclusions about the urgency of transitioning blockchains onto quantum-safe cryptographic footing. These early works sounded the alarm that, although blockchain technology was born in the classical computing era, it would need significant modifications to survive the advent of quantum computing.

Following these foundational studies, research interest in quantum-and-blockchain grew gradually in 2018 and 2019. Several works expanded the analysis of specific vulnerabilities and began to sketch out defenses. For instance, Fedorov et al.(92) published a commentary in *Nature* highlighting how quantum technology threatens blockchain security while hinting that it could also help make blockchains more secure (for example, through quantum random number generation or quantum cryptographic techniques). Matsuo (93) emphasized the pressing *need for* *quantum-safe security* in blockchains, surveying potential attack points (like signature schemes and key exchange protocols) and advocating for the integration of post-quantum cryptography sooner rather than later. Nouri et al.(94) likewise examined the prospects of quantum attacks on blockchain technology and reinforced the view that without proactive upgrades, many cryptocurrency systems could eventually be compromised by quantum-capable attackers. These studies collectively underscored that the vulnerability was not limited to Bitcoin alone—any blockchain relying on traditional public-key cryptography (or certain hash functions) could become insecure in the post-quantum era. They strengthened the case that the blockchain community must prepare by investing in quantum-resistant solutions.

In parallel with threat assessments, researchers also began exploring how to build quantum-resistant or quantum-secured blockchains. Efforts in 2018–2019 proposed new designs aimed at preemptively mitigating quantum threats. For example, Yu (95) and Ikeda (96) each presented frameworks for blockchain systems that would remain secure against quantum adversaries. These proposals typically involve swapping out vulnerable cryptographic components for post-quantum algorithms and sometimes even leveraging quantum technologies for added security. An illustrative milestone in this direction is the work by Allende et al. (97), who demonstrated a prototype of a quantum-secure blockchain on the Ethereum-compatible LACChain network. Their implementation incorporated post-quantum digital signature schemes and utilized sources of quantum entropy for key generation, establishing quantum-safe communication channels for transactions. This provided a proof-of-concept that integrating quantum-resistant cryptography into an existing blockchain is feasible and can protect the system from known quantum attack methods. Another notable example is the emergence of entirely new platforms like the *Quantum Resistant Ledger (QRL)*. QRL is a cryptocurrency and blockchain network designed from the ground up to withstand quantum attacks (98). It replaces vulnerable signature algorithms with the eXtended Merkle Signature Scheme (XMSS), a hash-based digital signature that is believed to be secure against quantum adversaries (99). The QRL project, launched in the late 2010s, was one of the first operational blockchains to implement a post-quantum signature scheme, thereby offering a practical testbed for quantum-safe blockchain technology. These pioneering projects (both in academia and industry) mark the beginning of a shift towards "quantum-proofing" blockchain infrastructure.

Most recently, the urgency and interest in this field have increased markedly, in part due to milestones in the wider cryptographic community. The U.S. National Institute of Standards and Technology (NIST) concluded its multi-year post-quantum cryptography standardization competition in 2022, selecting a suite of new quantum-resistant algorithms for digital signatures and encryption (100). This development has provided concrete options for blockchain developers to upgrade their protocols in anticipation of quantum threats. Indeed, in the past year or two, major figures in the blockchain industry have publicly discussed plans to transition prominent networks to post-quantum cryptography. For example, Ethereum co-founder Vitalik Buterin in early 2024 outlined a roadmap for migrating Ethereum's signature scheme to a post-quantum variant (101). Such plans are non-trivial and may take years to implement, but they signal that the community is starting to take the quantum threat seriously. At the same time, a critical question remains largely unanswered in the literature: *What will be the holistic impact of quantum computing across the entire blockchain ecosystem?* While large, resource-rich projects (like Bitcoin or Ethereum) can marshal the expertise to implement quantum-safe upgrades, many smaller or newer blockchain projects might struggle to do so in a timely manner. These less-resourced networks could be left dangerously vulnerable once quantum attacks become practical, potentially fracturing trust in blockchain technology as a whole. The present thesis begins to address this overarching question by examining a broad spectrum of blockchains under quantum threat and by considering both negative and positive implications of quantum computing for blockchains. In doing so, it builds upon the prior works surveyed above and pushes the discussion a step further – toward understanding not just how to defend against quantum attacks, but also how the advent of quantum computing might transform the blockchain domain, for better or worse. This combined consideration of threats and opportunities distinguishes our work and contributes a novel perspective to the growing body of literature at the intersection of quantum computing and distributed ledger technologies.

2.3 Comparison with Previous Works

Early works such as Aggarwal et al. (15) and Tomamichelet al. (91) established Bitcoin's vulnerability to quantum attacks but confined their analyses to a single platform. Later surveys by Matsuo (93) and Nouri et al. (94) broadened the discussion yet remained largely qualitative and fragmentary. In contrast, this dissertation conducts a systematic, cross-platform evaluation of signature schemes and consensus primitives, revealing patterns of quantum exposure that the earlier studies did not capture.

Aggarwal et al. also suggested that Grover's quadratic speed-up would be insufficient to destabilise Proof-of-Work in the near term. Our focused analysis revisits this assumption, modelling a quantum-enabled miner and showing that even a quadratic advantage can yield a disproportionate share of blocks as hardware scales, an effect not quantified in prior literature.

Where previous work treated quantum computing mainly as a threat, occasional references such as Fedorov et al. (92) and Allende et al. (97) hinted at constructive uses limited to stronger randomness or post-quantum signatures. This dissertation extends that idea by exploring how quantum accelerators could directly reduce Proof-of-Work energy costs and improve throughput, opening an energy-positive perspective absent from earlier security-centric studies.

Chapter 3

Blockchain Background

3.1 Blockchains Studied

The blockchains that were analysed were at the time of publication the five largest public blockchains based on their market capitalisation, technical innovation, and security. While this has changed to this date, many of the findings can still be applied to the top blockchains in the present day. A further reason that these blockchains were selected was in part due to the differences in their protocols, to analyse the effect that this would have. While there are some blockchains aiming towards quantum resistance, they are not considered in this study, as the current offering of quantum resistant blockchains have not seen a mainstream uptake in usage.

This chapter, will individually look at each cryptocurrency and discuss areas where they differ from other networks. This will focus primarily on the digital signature scheme, consensus algorithm, any zero-knowledge proof obfuscation mechanisms and how transactions are performed on the network. These are the critical elements that determine how vulnerable blockchains are to quantum devices.

Within this section, all the blockchains studied follow the structure laid out in Section 1.3.

3.1.1 Bitcoin

Bitcoin was first introduced by an author/authors writing under the pseudonym Satoshi Nakamoto (7). It presents a peer-to-peer electronic cash system that is designed to be completely decentralised and is completely secured by its cryptographic principles. It has been since its inception been the most widely used blockchain and the largest by market capitalisation (102). This dominance has only truly been questioned by the Ethereum network.

Transactions on the Bitcoin blockchain are represented by Unspent Transaction Outputs (UTXOs). These are a denomination of a specific number of bitcoin and can be considered a token. They are associated with a specific address made up of the hash of the public key. When a transaction is sent, one or more UTXOs are used by the sender as an input. The sender will then spend these UTXOs and create new ones as the output. These outputs are how ownership of Bitcoin is transferred when they are referenced in the input of a transaction.

Table 2 lays out the transactions fields for transactions sent on the bitcoin network (103). These structures can differ slightly depending on the type of transaction. For example, Seg-Wit transactions have additional fields for a marker and a flag, as well as a field in each input for the witness. For each transaction created, there can be multiple of the input fields and multiple of the output fields.

Bitcoin uses ECDSA to sign all of its transactions (104). There are three phases to ECDSA, these are, key generation, signing, and verification.

Field	**Sub-Fields**	**Size (Bytes)**	**Description**
Version	Version	4	This denotes the transaction format version, this can be changed to denote a non-standard format
Input Count	Input Count	1	Specifies the number of transaction inputs (users UTXOs)
Inputs	Previous Output Hash	32	This is the hash of the transaction where the input UTXO is located
	Output Index	4	The specific index within the transaction where the input UTXO is located
	Script Length	1	Length of the unlocking script
	Script Signature	Variable	Proof of ownership / ability to use the input as the owner. This is generally the digital signature associated with the UTXO and the public key of the user.
	Sequence	4	This allows the sender to lock a transaction to a specific period from the spending of the input UTXO, it also allows the user to denote that this UTXO can be used as a fee.
Output Count	Output Count	1	This is the number of output UTXOs there will be
Outputs	Value	8	The number of satoshis (1 × 10−8BTC) that are being sent to this output
	Script Length	1	specifies the length of the Script Public Key
	Script Public Key	Variable	This field specifies the conditions that have to be met to unlock this output and spend it. This contains a Public key hash (of the person the transaction is being sent to) or a script hash.
Locktime	Locktime	4	Defines the earliest time at which this transaction can be added to a block.

Table 2: This table shows the transaction structure for a standard Bitcoin transaction

Key pairs are generated in ECDSA firstly by creating a private key k, this value should be $1 \leq k \leq n - 1$ where is a prime value. From this, the corresponding public key can be calculated as, $P_k = k \times G$ where G is the elliptic curve generator.

Data from the transaction itself is used to create the signature, this allows verification later. This is in the form of the hash of the transaction data d. Not all the data seen in table 2 is used as the signature for the transaction itself is part of the data. The elements used for the hash are:

- Version
- Input Count
- Input Script Signature — Only for the input that is currently being signed. The signature itself is replaced with the public key associated with that UTXO
- Other Input Script Signatures — All other Script Signatures for other input UTXOs are set to 0
- Output Count
- All Outputs
- Locktime

An ephemeral key pair is generated in the same manner as during key generation, giving the ephemeral private key m and the corresponding ephemeral public key P_m, this public key has the co-ordinates $(P_m x, P_m y)$. The first part of the signature is the x-co-ordinate of the ephemeral key, calculated by $r = P_m x \ mod \ n$. The second part of the signature is calculated as $s = m^{-1} \times (d + k \times r) \ mod \ n$, this gives us the full digital signature for the transaction relating to d as $S = (r, s)$.

Given the digital signature S as well as the transaction hash d and the public key of the sender, P_k all of which are packaged in the transaction, any node can check the validity of the transaction. This is done by calculating $a1 = d \times s^{-1}\ mod\ n$ and $a2 = r \times s^{-1}\ mod\ n$. P'_m must now be calculated as $P'_m = a1 \times G + a2 \times P_k$. The x co-ordinate of this point on the elliptic curve should be equal to, r subject to mod n. If this is the case, then the verification succeeds. This is, as discussed in subsection 1.3.2 reliant on the hardness of the discrete logarithm problem.

Bitcoin introduced Nakamoto consensus (7; 104), this consensus algorithm works through the use of PoW as well as the longest chain rule. We introduced PoW in subsection 1.3.3. Bitcoin utilises an adapted version of the HashCash (105) which was originally designed as a mechanism to prevent denial of service and email spam. To achieve this, senders of emails were required to perform a small amount of work in the form of some computational problem. While this work was insignificant to an honest user, it became prohibitive to any malicious users who were performing an excessive number of actions. This was adapted to Bitcoin's Proof-of-Work (and most others) work on the hardness of finding hash values that are smaller than a defined value. To achieve this, the miner produces a hash of the block header, shown in table 3. The miner will be targeting some specific value, a target of 2^{256} would be the case where any hash is acceptable and 1 being that only 1 value in the entire SHA-256 search space would be correct. The mining algorithm for bitcoin is as follows:

1. Miner prepares a candidate block consisting of the block header and the set of transactions
2. Miner initialises or changes the nonce

3. Miner hashes the header and nonce together and then hashes again such that $H(H(header||nonce))$

4. Miner checks if $H(H(header||nonce)) \leq target$

   (a) if $4 = True$ miner broadcasts newly found block

   (b) else miner repeats 2-4

If we assume that the rules for hash functions described in subsection 1.2.1 hold, then the most efficient mechanism for solving the PoW problem in Bitcoin on classical infrastructures is a brute force method. One noticeable issue is the size of the nonce space. At only 4 bytes or total usable search space of 2^{32}, as the current bitcoin mining pool at the time of writing performs, $3.38 \times 10^{20} H/s$ could check that entire space in approximately $1.2716107 \times 10^{-11} s$. Table 4 shows the slightly different transaction structure of the coinbase transaction. The coinbase transaction is the first transaction in the list of transactions in a block. This is how a miner pays themselves a reward if they are successful. Within the slightly different structure there is a message section, this can be used as an extra nonce space. However, transactions are not included in the block header. The way that this is turned into an extra nonce space, any changes to the coinbase transaction space will change the Merkle root. This means, by using the Avalanche effect of hash functions, this space allows an additional 2^{256} of nonce space. As this is larger than the maximum search space of Bitcoin PoW, $2^{256} - 1$ this allows miners to have a virtually unlimited space to target PoW solutions. Comparatively, it would take the entire bitcoin network at the current hash rate 1.0866008×10^{49} years to search the entire 2^{256} search space, in perspective, this is in approximately 7.8710145×10^{39} times the current age of the universe.

The value that controls the complexity of the PoW problem is referred to as the difficulty. The more computational power that is acting upon the network,

Field	**Sub Field**	**Size (Bytes)**	**Description**
Block Header	Version	4	Provides the version number used by the miner
	Hash of the Previous Block	32	The hash of the previous block in the chain.
	Merkle Root	32	The hash root of all the transaction included in this block.
	Timestamp	4	Unix Timestamp
	Target Difficulty	4	The difficulty value that the miner was targeting (this will not vary miner to miner)
	Nonce	4	The integer changed by the miner to find a valid hash
Transaction Counter	Transaction Counter	Variable	Number of transactions in the block
Transactions	Transactions	Variable	List of all transactions found in the block (see table 2)

Table 3: This shows the structure of a standard block produced by a miner on the bitcoin blockchain

Field	**Sub-Fields**	**Size (Bytes)**	**Description**
Version	Version	4	This denotes the transaction format version, this can be changed to denote a non-standard format
Input Count	Input Count	1	This is always set to 1 for a coinbase transaction
Inputs	Previous Output Hash	32	For coinbase this is all zeros as there is no originating transaction
	Output Index	4	This set to 0xFFFFFFFF
	Coinbase Data Size	1-9 Bytes	This sets the size of the coinbase data field
	Script Signature	Variable	This is any data that miner wishes to include. However, it is often used as an extra nonce space.
	Sequence	4	This is set to 0xFFFFFFFF
Output Count	Output Count	1	This is how many outputs there will be. How many address' the coinbase transaction will be sent to
Outputs	Value	8	The number of satoshis that are rewarded
	Script Length	1	specifies the length of the Script Public Key
	Script Public Key	Variable	This field specifies the conditions that have to be met to unlock this output and spend it. This contains a Public key hash (of the person the transaction is being sent to) or a script hash.
Locktime	Locktime	4	This is set to 0

Table 4: This shows the structure of coinbase transactions. The main differences occur in the input section, as these are newly created tokens, there is no originating UTXO. Some of this space is used for message space for the miner

the greater the difficulty is. Block time is a part of many blockchains protocols as it ensures that blocks are produced in a timely manner, without allowing a large influx of blocks. A large influx of blocks could potentially lead to conflicting blocks also known as forks. Once every 2016 blocks (20160 minutes == 2 weeks). If 2016 blocks are reached before a 2-week period, the difficulty will be increased and if it takes longer it will be decreased. This tuneability of the difficulty of the problem is critical, as the computational power acting upon the network will not be static.

The longest chain rule defines what is the true canonical history. It is possible that two blocks can be produced at the same height, at which point there is a fork. If this occurs, consensus can carry on both chains. Eventually, one (most likely the one with the most computational power acting upon it) will become a longer chain. At this point, network resources will be diverted onto that chain. This is reliant on the fact that no one entity has more than 50% of the network's computational power. Due to the sheer amount of computational power that acts upon the Bitcoin network, this would be unlikely and prohibitively expensive.

3.1.2 Ethereum

Ethereum was created by Buterin et al. (106). It has a corresponding cryptocurrency called Ether. It is designed to allow the use of smart contracts and distributed applications (107). Ethereum recently transferred over to Proof-of-Stake (38), however at the time of publication this was not the case. Ethereum is often referred to as a global computer. There is a particular emphasis on the running of code as well as its use as a cryptocurrency. This ability to run code on the blockchain while possible on some preceding blockchains like Bitcoin was possible, it was not possible to the same extent.

Ethereum used to use PoW, however since the transition to Ethereum 2.0 this has been depreciated and Proof of Stake has taken its place. Proof of Work in Ethereum worked similarly to that of Bitcoin, it was some hard hashing problem. It, however, used a different algorithm called EthHash (108). EthHash was designed to be more resistant to specialist mining equipment, aiming to keep the mining pool more open. It did this by forcing the miners to solve memory hard problems. This involved the use of a Directed-Acyclic-Graph (DAG). This DAG is generated approximately every 5 days. Over time, the size of the DAG also increases in size. This affects specialist hashing devices such as ASIC miners more dramatically than other devices. Miner on Ethereum would create a block header. The structure of an Ethereum block can be seen in Table 5(107). The process for the Ethereum mining algorithm is as follows:

1. Miner prepares the candidate block
2. Miner initialises or changes the nonce value
3. The miner creates a Mix Hash using SHA3 hash of the block header, nonce, and a counter.
    (a) 64 pseudo random data points are selected from the DAG using the mix hash. This is then concatenated together
    (b) This new value is then hashed using SHA3
4. The hash output of the EthHash function is then checked to see if it less than or equal to the target value
    (a) if True: Publish block
    (b) else: Repeat steps 2-4

Step 3a is the part of the algorithm that is memory intensive, as to mine a block the miner will have to make many queries to the DAG.

Field	**Sub-Field**	**Size (Bytes)**	**Description**
Block Header	Parent Hash	32	Hash of the block being appended to
	Ommers Hash	32	Hash of all the uncle blocks of the parent
	Beneficiary	20	Ethereum address of the miner
	State Root	32	The root of the Merkle tree after the new transaction are added
	Transactions Root	32	Root of the Merkle tree of transactions in the block
	Receipts Root	32	Root of the Merkle tree for the transaction receipts
	Logs Bloom Filter	256	This bloom filter allows fast verification of any log events that took place as a result of smart contract or dApp interactions caused by the transactions in the block
	Current Difficulty	1-4	Difficulty that the block was mined at
	Block Number	1-4	Block number in the chain (should be the parent +1)
	Gas Limit	1-4	Limit for the total amount of Gas that can be used by all the transactions in the block
	Gas Used	1-4	The actual amount of Gas used in the block
	Timestamp	4-8	Unix timestamp
	Data	32	Any additional data added by the miner
	Mix Hash	32	Cryptographic value that is used by the miner to complete the PoW
	Nonce	8	Value used to create the PoW
Ommers	Ommers	N/a	An array of all the current uncles of the block
Transactions	Transaction	N/a	Array of all the transactions that have been incorporated into this block

Table 5: Table showing the block structure created by and Ethereum miner during PoW

Ethereum 2.0 updated to PoS in 2022. All perspective validators are required to stake at least 32 Ether into a smart contract. Once you have become a validator, you can then contribute to the consensus of the network. The PoS is split into Epochs, one Epoch is made up of 32 slots. A slot is 12 seconds, this can be considered the block time as a single validator is selected in each 12 second slot to produce a block.

The steps for the PoS consensus algorithm on Ethereum 2 are as follows (109):

1. Committee Selection — Committees are formed giving each validator a different job, this is achieved based on the validators position in the validator set combined with a source or randomness called RANDAO (110).

2. Block Proposal — During a slot, the selected validator for that slot collects transactions from the transaction pool and packages them into a block.

3. Attestations — Other validators validate or invalidate the proposed blocks for each of the time slots in an epoch.

4. Aggregation — The attestations are aggregated together to form a single attestation.

5. Inclusion — The confirmations for this block are added to a proceeding block, confirming if it is valid or invalid.

6. Finalisation — Once over two thirds of a validator set have agreed to a block, then it is finalised.

PoS was implemented primarily as a mechanism to prevent centralisation. In theory, any user capable of purchasing or owning the required staking amount can partake in consensus on the blockchain. This has, as a by-product, led to a reduction in the energy usage by the blockchain.

To run code on the Ethereum network, dedicated smart contract languages such as solidity were developed (111). This is all made possible by the Ethereum Virtual Machine (EVM) (107). The EVM is the runtime environment for the smart contracts. The EVM can be divided into 8 distinct components:

- Execution Environment — This is the context within which compiled smart contract code behaves when it is executed. This information can be taken from the transaction interacting with the contract, as well as the original transaction that set up the contract. These contexts include:
  - Interacting Party — Who is the individual that is calling the smart contract, for example, the contract creator may be able to perform an action that another user can not.
  - Origin — Who created the contract.
  - Gas Price — How much the user willing to pay in for the Gas used
  - Gas Limit — What is the limit of Gas the user has set for execution for the contract interaction
  - Value — Amount being sent to the contract by the user
  - Block Information — Information about the current block.
  - Data — Users may look to execute specific functions within a smart contract.
  - Code — This is the code from the smart contract that is being executed during a transaction interaction
- World State — The world state holds the current state of all Ethereum addresses on the network. The world state is organised as a Merkle–Patricia tree (112; 113). Whenever a contract is deployed or a transaction invokes an existing contract, every update to this state must be executed atomically and deterministically, so that all nodes reach the same result.

- Stack — Temporary storage of values for the execution of contract code, and works on a last-in-first-out basis.
- Memory — This contains the data that is needed while executing a smart contract. This data will be lost at the end of contract execution.
- Storage — This is the storage of smart contract state data on the blockchain. This allows variables and the contract state to persist over multiple executions of a contract.
- Gas — This is the mechanism that the EVM uses to prevent over use of resources. Almost every action has a Gas cost. This prevents malicious users from occupying most of the network's resources.
- OpCodes — Ethereum Smart contract run on a series of Op-Code. These Op-Codes provide the necessary functionality to make Ethereum Turing complete. These Op-Codes include arithmetic, logic, stack manipulation and other mechanisms for interacting with the World State.
- Calls — Calls between contracts allow complex actions to be performed. The EVM allows this through messages and calls.

These elements allow the EVM to provide a secure and complex environment to allow the Ethereum blockchain to run complex code and have a massively diverse environment of applications.

Ethereum works on an account-based system. Ethereum user accounts are called Externally Owned Accounts (EOA). This means that instead of using UTXO's stored in blocks to keep track of transactions, Ethereum accounts and the Ethereum global state keep track of account balances. When a transaction is sent by a user, instead of pointing to an output where their key pair received a transaction, they use their account balance and if the transaction the value of it will be deducted from the global balance, changing the state. This means that the same key pair is used multiple times by a user, unlike with UTXO based blockchains, where a key is generally only used for one spending action.

The transaction structure for Ethereum is one of the many ways in which Ethereum differs from many other blockchains.

Ethereum, like Bitcoin and many other blockchain technologies, implements ECDSA as their digital signature scheme, this is done in the same manner as explained for Bitcoin in subsection 3.1.1. All the SHA-256 hashes are replaced by a single round of SHA-3 hashing. However, there is a major difference as can be seen in Table 6 there is no field in the transaction structure for the Ethereum users' public key. In Ethereum, there is a process of public key retrieval. Instead of the public key, the transaction contains the value P'_m where $P'_m = a1 \times G + a2 \times P_k$. This can be rearranged such that $P_k = (P'_m - a1 \times G) \times a2^{-1}$. Given that we know the values for P'_m, $a1$ and G we can calculate P_k. This will give two possible points on the elliptic curve. However, this can trivially be checked by hashing both possibilities for P_k and one will match the Ethereum account. Additionally, this can be checked using a third part of the signature provided by the user v. This will have the value of 28 or 29 which will direct the validator to one of the two points on the curve to retrieve the public key.

3.1.3 Litecoin

Litecoin was one of the earliest blockchains, created by Charlie Lee in 2011 (114). It was created with faster transaction throughput at the forefront as a rival to Bitcoin. It is a direct fork of bitcoin with some minor alterations. It mirrors Bitcoin in almost every way and can be considered a case study for the re-use of technology that arose as the result of the success enjoyed by Bitcoin.

Field	**Size (Bytes)**	**Description**
Nonce	1-4	This is used to prioritise transactions; however, it also used to ensure that transactions are unique. Because of the transaction structure, it is possible to send two transactions with the same transaction has. An EOA cannot send two transactions with the same nonce.
Gas Price	1-4	The amount that a sender of a transaction is willing to pay in Gwei (1 × 10−9ether) for each unit of Gas.
Gas Limit	1-4	How much Gas the sender is willing to use. This will affect how fast the transaction goes through, but also must be at least the cost of the actions performed by the transaction.
To	20	The Ethereum address that is being sent to or the contract address
Value	32	The amount in Wei (1×10^{-18} ether) to be sent.
Data	Unlimited	This field can contain any kind of data. Generally, the larger the size, the more it will cost. This is where compiled byte-code for contract deployment is added
Signature (v,s,r)	65–68	Elements of the signature for the transaction. 32-Bytes for s and r and 1-4-Bytes for v

Table 6: This table shows the transaction structure for an Ethereum transaction. A lot of the sizes are variable, however the values given are the general sizes that will commonly be used

While the differences are minor when compared to Bitcoin, some include:

- **Block Time** — The target block time between publishing of blocks for Litecoin is 2 minutes 30 seconds
- **Consensus** — Litecoin employs a memory hard consensus algorithm called Scrypt, while still based on hash problem PoW.
- **Supply of Tokens** — The total supply of Litecoin tokens is 84 million compared to Bitcoins 21 million

Litecoin follows the same block and transaction structure as Bitcoin. Therefore, tables 3 and 2 apply to Litecoin as well as Bitcoin. One Litecoin also has a smaller denomination called Satoshis, these have the same conversion rate of 1 LTC $= 1 \times 10^8$ Satoshis. Furthermore, the ECDSA implementation seen in subsection 3.1.1 also applies.

One of the primary differences between Bitcoin and Litecoin is Litcoin's use of the Scrypt hashing algorithm for PoW (115). Scrypt is a memory hard algorithm. This aims to reduce the use of specialist mining equipment such as ASIC miners. It is a Key Derivation Function (KDF). The Scrypt algorithm works as follows:

1. Create Block Header — Block is created as described in table 3.
2. Nonce is initialised or changed
3. Compute Scrypt Hash
    - Key mixing — The ROMix function takes in two inputs, these are the block header and the current blockchain difficulty. The KDF is applied to them to generate a key.

- Memory Hard algorithm — A large memory buffer is generated as an array. This is pseudorandomly generated according to the key. Over several iterations, read-write functions are performed. This step requires a large amount of memory.
- Hash output — From the values returned from ROMix KDF is used again to output a key

4. The output of step 3 is added to the block header as the scrypt hash
5. Check to see if it has output is less than or equal to target value
   - if True: Broadcast complete block
   - else: return to step 2

Litecoin, disregarding some minor changes at the protocol level. mirrors Bitcoin almost exactly, This technology reuse is a key feature of blockchain technologies. This technology reuse will prove important when analysing the quantum resistance of blockchain technologies in general.

3.1.4 Monero

Monero is a completely privacy-focused blockchain (116). The aim is to provide end to end anonymity to the user. All transactions are obfuscated on the network and through the use of various cryptographic techniques, who the user is sending Monero tokens too is hidden also. It was originally a fork of one of the earliest blockchains, Bytecoin (117). Monero however, because of its privacy-focused element, has led to the raising of concerns about its use for illicit activities as the obfuscation mechanisms are cryptographically secure (118).

Monero employs a PoW-based consensus algorithm called RandomX (119). RandomX is designed to be run on personal devices rather than specialist mining infrastructures. Random programs are run inside a virtual machine, this reduces the effectiveness of ASIC miners. The block header shown in table 7 is used to produce a valid hash in RandomX. RandomX PoW works as follows:

1. The block header created by the miner is used to create a random program.
2. Execution of the program is done using a dataset that is created or retrieved from another user by the miner before mining.
3. The program is executed by the miner. It performs the specified operation on the dataset, giving an output.
4. The output given is hashed using a Blake-2b hash (120)
5. Miner checks if hash is less than or equal to the target value
    - if true: broadcast block to other nodes
    - else: return to step 1

As part of Monero's focus on user privacy, it employs a different digital signature scheme. This scheme is called Ring Signatures (121). It allows the sender to mask their identity, so from an external position, another user will not know who sent the transaction. They are based on One-Time Ring Signatures (122) Ring signatures like ECDSA work using elliptic curve cryptography. This means that they rely on the hardness of the discrete logarithm problem. It uses an Ed25519 elliptic curve, this is a twisted Edwards curve (123). Ed25519 elliptic curve cryptographic scheme is generally faster for key generation and verification, for example, Bernstein et al. (124) showed that Ed25519 was twice as fast as secp256-k1 for scalar multiplication function when ran on the same infrastructure.

Key generation for EdDSA is completed by choosing a random value k, this is the private key, this is then multiplied by the base point of the Ed25515 curve G,

Field	**Sub Field**	**Size (Bytes)**	**Description**
Header	Version	1-2	The major version number of Monero that is being used
	Minor Version	1-2	The minor version number of Monero that is being used
	Timestamp	4-8	Unix timestamp
	Previous Block Hash	32	Hash of the preceding block that this one will be attached to
	Nonce	4	Value used to find a valid block hash
Coinbase Transaction	Coinbase Transaction	Variable	For more detail on this see \ref{}
Number of Transactions	Number of Transactions	1-4	Value indicating how many transactions are packaged into the block
Transaction Hashes	Transaction Hashes	32 for each transaction	A 32 byte is added to this field for every transaction incorporated into the block
Transactions	Transactions	Variable	Serialised structure of all the transactions in the block

Table 7: This table shows the structure of a Monero block produced by a miner

this gives $P_k = k \times G$ with P_k being the user's public key.

Ring signatures as laid out by Rivest et al (122). allow a signer to create a signature on behalf (or under the pseudonym) of a set of other signers without the need for all the private keys. This ring signature can be used to completely obfuscate who the sender was, it, however, does allow verification that it was signed correctly and fairly by a user. This scheme relies on the hardness of the Diffie-Hellman problem, this can also be considered the Discrete Logarithm problem as if you can efficiently solve one you can also solve the other. Further details of the specific ring-signature implementation used by Monero or the cryptographic operations of ring signatures in general are outside the scope of this thesis.

Monero implements a Zero-Knowledge proof scheme called Bulletproofs (125). While this thesis will not explore the intricacies of Bulletproofs, they are utilised to hide transactions and validate that they abide by the rules of the blockchain protocol (No overspend, no invalid spend and no double spend). They work through the use of Pedersen commitments (126) and the use of inner-product proof that allow a user to prove a value is within a specific range without revealing the value. This specific mechanism was adopted by Monero was due to it allowed transactions to be up to 80% smaller and allowed faster verification.

3.1.5 Z-Cash

Z-Cash (127) like Monero employs zero-knowledge proofs to obfuscate transactions on the network. Since analysing the blockchains discussed here, privacy-based blockchains have decreased in popularity due to governmental and policy pressure (128; 129), Z-Cash and Monero are both still major stakeholders in the blockchain industry.

Z-Cash however differs from Monero in that it can transact publicly visible tokens to private tokens and vice versa. To achieve this, it utilises a Zk-Proof mechanism called Zk-SNARKS (Zero-Knowledge Succinct Argument of Knowledge) (130).

Z-Cash itself implements a variation on the Bitcoin PoW problem called Equihash (131). This problem is a memory-based problem rather than a computational power problem. This is done to discourage the use of ASIC miners, thereby providing a more decentralised mining network where specialised mining devices are not required.

In this problem, rather than solving for a hash digest with a specific form as in Bitcoin, the miner is required to find a set of values that have a specific relationship. This problem is based on the difficulty of the Generalised Birthday Problem (132). The generalized birthday problem involves finding specific elements from multiple lists that sum to a particular value. Within the problem, the miners have a dataset, and they are required to find a specified number of related values from the data set. As the data set becomes large, the problem becomes harder. The problem can be more specifically described as a miner is given a set of binary strings of a particular length. The miner must find subsets of these strings, that XOR to the same value.

Equihash has two key parameters, n and k. n is the length of the binary strings. k is the number parameter that determines how many hashes must be XORed together to find a valid string of all zeros. Specifically 2^n hashes. n represents the space complexity while k represents the time complexity.

Given n and, k the mining process runs as follows:

1. Miners create a list of potential binary strings given the Block Header and value n
2. Binary strings are paired together
3. The miner iterates through the list k times, reducing the list of binary strings
4. If the miner has by the end of the search found a subset of 2^k binary strings that when XORed together produce a string of all zeros, they have succeeded.
5. Else return to step 1

The initial values for n and k for Z-Cash were set to 200 and 9 respectively.

Z-Cash originates from a project which was a hard fork of Bitcoin called zerocash; therefore the same ECDSA algorithm is used in Z-Cash as laid out in section 3.1.1.

While this thesis will not explore ZK-SNARKS in depth, they are zero knowledge proofs that rely on the hardness of a public key created in a 'ceremony' between multiple members of the network. As long as there is one honest actor that takes part in the ceremony when creating the trusted set-up. Upon the completion of the ceremony to create the trusted key, as long as one member destroys their element of the key, their public key will be safe. Using this shared key, a prover (transaction sender) can create a proof that allows any user to verify the transaction that they are sending is valid. Using the shared key, any member of the network can verify a transaction is valid. The trusted key element of this process relies on the hardness of the discrete logarithm problem.

3.1.6 Others

Throughout our analysis, a cursory analysis of some more minor blockchains was considered. The aim of which was to highlight the reuse of technology and that the methodology could be applied to almost every blockchain currently existing. Two blockchains employing Zk-proofs were considered, as well as three Bitcoin hard forks.

Grin

Grin (133) is a blockchain protocol that is privacy focused, similar to Monero and Z-Cash. It utilises a Zk-proofs protocol called Mimblewimble (134). Since the time of writing, this blockchain has seen almost no uptake. This is likely due to a persistent 51% attack (135). Mimblewimble is a protocol reliant on the hardness of the discrete logarithm problem. Grin utilises a Graph-Theoretic PoW mechanism called the Cuckoo Cycle (136). Grin's Mimblewimble protocol uses Schnorr signatures to prove ownership over tokens, which are more space efficient than ECDSA. However, like ECDSA, Schnorr signatures are reliant on the hardness of the discrete logarithm problem.

BEAM

BEAM (Bringing Mimblewimble to Everyone, Anywhere, Manifoldly) is another privacy-based blockchain that utilises Mimblewimble for its security. BEAM implements an Equihash PoW mechanism as in Z-Cash with the settings for n and k set to 150 and 5 respectively.

Bitcoin Cash

Bitcoin Cash is a direct hard for from Bitcoin, with some minor changes. Bitcoin Cash maintains ECDSA signatures that are implemented in Bitcoin; however, they also allow the use of Schnorr signatures. Outside of this, many elements such as the block time and the consensus protocols remain the same.

Bitcoin Gold

Bitcoin Gold was a hard fork of the Bitcoin blockchain that started at a similar time to Bitcoin Cash. While there are no changes to the ECDSA scheme used in Bitcoin, Bitcoin Gold does implement an Equihash PoW mechanism similar to Z-Cash.

Bitcoin SV

Bitcoin SV is also a hard fork of the Bitcoin blockchain with little to no changes from the Bitcoin core protocol.

Chapter 4

Quantum Attacks on Blockchains

In this chapter, analyse the quantum vulnerability of 10 large blockchains in market capitalisation, while focusing on performing a more rigorous analysis on 5 of the 10. The background of these blockchains were presented in Chapter 3. Through this analysis it will be demonstrated that the issue of quantum vulnerability is an industry-wide problem. However, any changes, even slight ones, to the protocol will cause changes in the level of vulnerability of the network. This analysis shows the danger approaching for almost all blockchains with billions of dollars worth of cryptocurrency potentially vulnerable.

This chapter is based upon work published in the **Elsevier** journal **Array** in July 2021 (1). This work looks at the current state of the blockchain industry regarding their vulnerabilities to quantum aggressors. This analysis was done using a similar technique that were applied to the Bitcoin blockchain by Aggarwal et al. (15). The aim of this research is to highlight the danger that is posed to the blockchain industry as a whole. This is further compounded by the decentralised governance that is one of the key aspects of all public blockchains.

4.1 Preliminaries

Since 2008, blockchains have only increased in popularity and notoriety. They do, however, provide an interesting case study for the ever-growing threat of quantum computing technologies. Blockchains are designed with decentralisation in mind, meaning that no one single entity has authority over the system. This allows blockchains to have some advantages over centralised systems, for example the ability to be trust-less, no single point of failure and significantly less censorship. However, a major disadvantage is that due to there not being a single point of authority, there is no one key decision maker. This means that the ability to make changes swiftly is dramatically reduced (137). Additionally, the cryptographic principles integrated within blockchains are so entrenched, that the failure of these systems has no fall back. This is because the system is trustless meaning that no user has to trust another user, they are only required to trust the cryptographic principles and the rules of the blockchains that they underpin.

Within this chapter, it will be what the current state of the blockchain industry and how vulnerable it would be to a quantum attacker. Furthermore, it will look to some extent about what would be required to upgrade these blockchains to being post-quantum resistant and in some cases if that is even possible.

4.2 Results

Our results are presented in Tables 8 and 9.

Table 8 details the vulnerabilities of each blockchain in detail. As can be seen from the table, there were almost uniform vulnerabilities for all the blockchains. While there were slight differences in the mechanism of attack, all of them were vulnerable to the exponential speed up of Shor's algorithm against ECDSA. While there were some vulnerabilities in all the PoW-based blockchain, this was considered a minor issue when compared with the best-case scenario of a quantum being able to trivially steal transactions that are broadcast to the network. At worst, an attacker could take over whole accounts or even create their own cryptocurrency in the case of Z-Cash. There were some ameliorating factors for the blockchains that utilise Zero-Knowledge proofs, as the value of a transaction and the user identity additional attacks must be performed by a user to un-obfuscate those transactions.

Table 9 shows an analysis of how vulnerable each of the major blockchains were in comparison with each other. As can be seen from the table and as previously discussed, all the major blockchains have a major vulnerability to quantum attack. The ratings were created as follows:

- Green (✓) — Non-quantum vulnerable as far as we are aware.
- Orange (–) — Quantum vulnerable, however, not considered a major concern for attack. Generally, this is due to either a long-term timescale, or it would require an unrealistic amount of quantum computational power.
- Red (✗) — Major vulnerability that if left unresolved could severely weaken or even break the blockchain.

These ratings were determined by; what the potential financial gain of an attacker would be, were there any protocol level mitigating factors, and what is the timescale that would be required for a quantum device to be capable of doing this.

Overall, it is clearly demonstrated that quantum vulnerability is rife across almost all current blockchain networks. There are hundreds of blockchain protocols and projects that rely upon these major blockchains. An example of this is Ethereum. There are 1385 other cryptocurrencies outside of Ether that are reliant on the Ethereum blockchain (138). This includes major tokens such as

Blockchain	**Risk Level**	**Target**	**Vulnerabilities**
Bitcoin	High	Transactions declared to the network	Transactions declared to the network are vulnerable to quantum attack, specifically regarding their signature scheme. The main form of attack identified is against transactions declared to the network which have not yet been incorporated into a block. Using the public key declared by the sender of a transaction, a quantum attacker can find the private key. This allows them to duplicate the transaction with whichever output location they desire.
Ethereum	High	Re-use of public keys	Ethereum is designed on an account-based system, within which reuse of public keys is common. The attack mechanism we have identified can target accounts that have previously declared transactions to the network, while still retaining some Ether tokens in the account. By solving the public key to gain the private key using Shor's algorithm, a quantum attacker could forge transactions in a user's name, by generating a valid transaction signature.
Litecoin	High	Transactions declared to the network	As Litecoin shares a majority of its technical structure with Bitcoin, it is equally vulnerable to quantum attack. The most damaging attack technique as in Bitcoin is against transactions declared to the network that have not yet been added to the blockchain.
Bitcoin Gold	High	Transactions declared to the network	Due to the similarities with the Bitcoin cryptographic elements, Bitcoin Gold shares the same vulnerabilities.
Bitcoin Core	High	Transactions declared to the network	Due to the similarities with the Bitcoin cryptographic elements, Bitcoin Core shares the same vulnerabilities.
Bitcoin Cash	High	Transactions declared to the network	Due to the similarities with the Bitcoin cryptographic elements, Bitcoin Cash shares the same vulnerabilities.
Monero	Medium	Obfuscated transactions and transactions declared to the network	The signature scheme used in Monero EdDSA is vulnerable to quantum attack as it relies on the discrete logarithm problem. However, Monero gains some resilience to quantum attack through the anonymity of its users as well as the amounts being transacted. Although the Bulletproof protocol used in Monero to achieve this obfuscation of transacted amounts is vulnerable to quantum attack, an attacker would be reliant on luck to select a transaction of significant value. Furthermore, due to a recent change in the consensus protocol implemented on Monero where RandomX was introduced, it would also have further resistance to quantum attacks attempting to perform a 51% attack utilizing Grover's algorithm.
BEAM	Medium	Obfuscated transactions and transactions declared to the network	BEAM's signature scheme, as well as the obfuscation technique Mimblewimble, are vulnerable to quantum attack. Quantum attack could both, intercept transactions broadcast to the network and remove anonymity from hidden transactions. However, as with Monero, the hiding of transaction and account values removes some incentive for a quantum attacker.
Grin	Medium	Obfuscated transactions and transactions declared to the network	Grin's signature scheme, as well as the obfuscation technique Mimblewimble, are vulnerable to quantum attack. Quantum attack could both, intercept transactions broadcast to the network and remove anonymity from hidden transactions. However, as with Monero, the hiding of transaction and account values removes some incentive for a quantum attacker.
ZCash	Very High	Public parameter generated during the Zk-SNARK ceremony	ZCash is highly vulnerable to quantum attack against both its consensus algorithm and its signature scheme. However, the most damaging attack found against ZCash is the vulnerability of its zero-knowledge proof protocol ZK-SNARKS, as this obfuscation method requires a trusted set-up and therefore the production of a public parameter, which is a public key. If a quantum attacker gains the private key to this public parameter, they will be able to generate tokens at will.

Table 8: **Blockchain Quantum Vulnerability Overview:** This table shows a summary of the blockchain vulnerabilities discussed in this thesis. The table shows, from left to right, the blockchain in question, the level of risk established here, the particular underlying cryptographic technology at risk, and a summary of the attack. This table is quoted from (1)

Blockchain	**Subgroup-Finding algorithm (Shor's)**	**Amplitude Amplification (Grover's)**
Bitcoin	✗	–
Ethereum	✗	–
Litecoin	✗	–
Monero	✗	✓
ZCash	✗	–

Table 9: **Vulnerabilities of Key Blockchain Technologies:** This table shows the vulnerabilities of key cryptocurrencies against two forms of quantum attack. An ✗denotes the blockchain has strong vulnerabilities against quantum attacks: due to the exponential quantum advantage for such attacks, as soon as quantum computers exist with sufficient memory, these could be used to effectively attack the blockchain in question. A – denotes that the blockchain has an intermediate level of vulnerability: while a quantum advantage exists, this is only quadratic in nature; hence it will take longer for quantum technologies to advance to the point of becoming a threat. Finally, a ✓means that the cryptocurrency is currently considered safe from quantum attacks. This table is quoted from (1)

Tether(139) and Chainlink(140; 141) that have a combined total circulating value of approximately $129 million.

4.3 Vulnerability Analysis

Each of the analyses conducted was done in three steps, firstly analysis of the digital signature, this is particularly relevant regarding the flow of the transactions. Secondly, weaknesses to the consensus algorithm, due to the static nature of blockchain technologies this is the only part that will differ in-between the original paper from the time of writing and this thesis. A brief overview of the changes and their possible impacts will be given. Finally, blockchain protocol-specific vulnerabilities or strengths will be investigated.

The analysis considers the two quantum algorithms laid out in 1.4, Shor's algorithm and Grover's algorithm. Shor's algorithm is applicable for any areas where cryptography relies up on the hardness of integer factoring, discrete logarithm or Diffie-Hellman problems. This is generally the digital signature schemes and some of Zk-proofs implementations. Grover's algorithm is applicable to unstructured search space problems, while this could be applicable to all the problems discussed here, our focus will be mainly on the PoW mechanisms. While the effects of these quantum algorithms are well known, our focus for the analysis is more the mechanism for attack, and how minor changes to blockchains protocol change their vulnerability. We aim to give a ranking of the more and less vulnerable blockchains, however due to technology reuse this is generally minimal.

It is well established that elliptic-curve cryptography (ECC) schemes—including ECDSA, Schnorr, and EdDSA—succumb to Shor's quantum discrete-logarithm algorithm. Because every blockchain analysed in this dissertation (e.g. Bitcoin, Ethereum, Litecoin, Monero and Z-Cash) authenticates transactions with one of these ECC variants, a sufficiently powerful fault-tolerant quantum computer would enable an adversary to derive private keys directly from the on-chain public keys. The high-level attack workflow is captured in the following pseudocode, which assumes availability of Shor's quantum subroutine for solving the elliptic-curve discrete-log problem. Pseudocode for how a quantum device would retrieve a private key from a public key is found in 1

Protocol 1 Quantum extraction of an ECC private key via Shor's algorithm

```
1: **Input:** public key $Q = kG$ on curve $\mathcal{E}$ (order $N$, base point $G$)
2: **Output:** secret key $k$

3: Prepare two quantum registers $|0\rangle|0\rangle$.
4: Apply Hadamard gates to the first register to obtain the uniform superposition $\frac{1}{\sqrt{N}}\sum_{x=0}^{N-1}|x\rangle$.
5: Compute $xG$ in superposition: map $|x\rangle|0\rangle \rightarrow |x\rangle|xG\rangle$.
6: Apply the Quantum Fourier Transform (QFT) to the first register.
7: Measure both registers, yielding a sample pair $(x, xG)$.
8: Repeat steps 3–6 until enough independent samples are collected.
9: Classically post-process the samples (e.g. via continued fractions) to recover $k$.
10: **return** $k$.
```

Consequently, this dissertation proceeds to a protocol-level deep dive of each selected blockchain, examining how their specific implementations of ECDSA, Schnorr, or EdDSA—as well as ancillary design choices such as address formats, key-reuse policies, and signature-aggregation mechanisms—modulate their exposure to quantum attacks and shape the feasibility of post-quantum migration paths.

4.3.1 Bitcoin

As previously stated, Bitcoin's vulnerabilities have been thoroughly documented by Aggarwal *et al.* (15). Nevertheless, for completeness, our own analysis of the Bitcoin blockchain will be presented, employing the methodology developed by Aggarwal et al.

Before a Bitcoin address has been spent from, the underlying public key remains hidden, so it is not yet susceptible to a direct quantum attack. A standard Bitcoin output (P2PKH) locks the funds to *the hash of* the recipient's public key rather than to the public key itself; the key is revealed only when the coins are later spent. Because Shor's algorithm does not threaten cryptographic hash functions—and Grover's algorithm offers only a quadratic speed-up against them—these unspent outputs enjoy a strong interim layer of protection. The same design pattern, and therefore the same caveat on quantum vulnerability, holds for every blockchain analysed in this thesis.

Blockchains in comparison with other real-world systems are generally immutable, this key point gives some resilience to cryptocurrency transactions as once a transaction is finalised only with a 51% or similar attack could a bad actor reverse / steal a transaction. While this may be possible on some smaller blockchains, the larger PoW-based blockchains, especially Bitcoin, are highly resilient due to the large amount of computational power acting on the network. With this said, in chapter 4 we investigate the possibility of a quantum attack on a blockchains PoW mechanism.

As discussed in section 1.4 Shor's algorithm has an exponential advantage when compared with any known classical algorithms when trying to solve factoring, discrete logarithm or Diffie-Hellman problems. For ECDSA, Shor's algorithm can solve a public key for its private key in polynomial time of $O(n^3)$. This is compared to $O(2^n)$ on classical infrastructures.

Utilising the quantum device's exponential advantage on Bitcoin's ECDSA scheme, the primary method for attack of Bitcoin is for a quantum attacker to target transactions. These transactions must have been broadcast to the network, but have not been added to the ledger (reached finality). Transactions on the bitcoin network are not automatically added to the blockchain, there is a waiting period where they are held in the transaction pool. How fast they are added is determined by the reward payment for the miner they sender adds, as well as how busy the network is. Generally, transactions will take at least 10 minutes to reach finality (the block time for Bitcoin) but often could take longer, possibly even hours. Once a transaction has been created, the user includes their public key as seen in table 2.

If a quantum attacker can solve the user's public key for the private key before the transaction is finalised, they could create a competing transaction. Furthermore, they could add a block reward as an additional incentive for miners to prioritise their transaction over the legitimate users. By doing this, a user would essentially hijack a transaction and be able to direct the output to whatever location they wish to. Additionally, they could steal any other UTXO's associated with the public private key pair. Generally, however, there is limited key reuse in Bitcoin, meaning that an attacker would only be able to gain access to the cryptocurrency controlled by the specific key they attack rather than all cryptocurrency owned by the individual. This does give the user some protection from attack, however upon a quantum device capable of solving an ECDSA public key for its private key in under 10 minutes, all transactions on the network are vulnerable.

Possibly, the most damaging attack on the Bitcoin blockchain, however, is against some of the oldest and most important transactions on the network. When the Bitcoin protocol was originally implemented, it was only possible to pay to public key. This means that all outputs were directly to public key. This is evident if you take the first ever mined block. The block was mined by Satoshi Nakamoto with a payment of 50BTC (approximately $1.7 million). This transaction is unspent; however, the public key is available. It is estimated that Satoshi Nakamoto owns about 1 million Bitcoin (Approximately $34 Billion). Most of these UTXO's are secured only using a public key rather than the hash of the public key. These UTXO's would provide a high-value target for quantum attack. Furthermore, there is no time restriction on these UTXO's meaning that a significantly slower quantum computer could perform these attacks when compared to the hijacking of transactions. This attack would do untold damage to the Bitcoin blockchain. Furthermore, if the keys for Satoshi Nakamoto's accounts are permanently lost, there would be no mechanism by which they could be secured against quantum attack. This means the only possible response is to permanently lock them. This would have a serious effect on the cryptocurrency, as this amount of Bitcoin represents about 5% of the current supply of Bitcoin and about 4.7% of the total supply.

Within this chapter, we will not consider in depth detail for quantum attack against PoW, as it is covered in detail in chapter 5. However, PoW used by Bitcoin can be mined by a quantum device with a quadratic speed-up when compared to a similar scale classical device. This coupled with the large amount of computational power acting upon the network means that a midterm quantum attack on Bitcoin is not feasible. However, with sufficiently, it could be possible to mount a 51% attack on any PoW-based blockchain.

4.3.2 Ethereum

Despite the use of relatively similar cryptographic mechanisms, the variance in the protocol of Ethereum and Bitcoin means that the vulnerabilities change. Ethereum, as discussed in 3.1.2 utilises a different structure for storing information on the blockchain. Bitcoin and many other blockchains utilise a UTXO based structure, while Ethereum uses an account-based structure. Ethereum EOA accounts hold a balance, and this balance is stored in the world state. This means that a user sends and receives transactions to the same account consistently. Ethereum accounts do not have to reference where they received cryptocurrency, rather just the account and prove they have ownership over the account.

Accounts are secured by a public / private key pair that can be used by a user indefinitely. However, this also means that once a single transaction is sent from this account, the public key is now in the public domain. While this is not at risk from classical devices due to the difficulty of solving a public key for a private key due to the hardness of the discrete logarithm problem, a quantum device that is capable could take control of the balance of an entire account. This could have potentially serious ramifications outside of a user losing their account balance. This could lead to smart contracts deployed by the user being compromised, potentially leading to large projects failing and non-compromised users losing access to tokens. This could lead to some Ethereum ecosystems failing as a result of a single compromised account.

The recent development of Ethereum moving to PoS could potentially mitigate any risk of 51% attack against the network. However, a potentially major flaw in this upgrade is the requirement of all PoS nodes to submit their public key to a central register. This means that every staking node could be vulnerable to attack, potentially leading to a malicious user taking control of consensus on the network.

4.3.3 LiteCoin

Analysis of Litecoin shows the impact of technology re-use across blockchain networks. Litecoin mirrors most of the cryptographic primitives and even protocol structure that is seen in Bitcoin. Due to the use of ECDSA in Litecoin, the vulnerabilities and attack vectors found in Bitcoin present the most urgent threat here as well. This is seen across many blockchains within the industry and becomes a recurring theme throughout our analysis.

The PoW mechanism for LiteCoin is based on a difficulty problem. While there is a memory hardness parameter, a quantum device running Grover's logarithm can still gain a quadratic speed-up compared to classical devices for the computation hard portion of the task. This however in this research is not considered a primary form of attack.

As Litecoin was designed with higher transaction throughput in mind when compared to Bitcoin, it has a lower block time. This block time is 2 minutes. This reduced block-time means that Litecoin has some more resilience to attack, as a quantum attacker has a shorter period of time to break the ECDSA public key and form a competing transaction.

Despite this advantage, the difference between Bitcoin being vulnerable and Litecoin being vulnerable would be minimal. If Bitcoin is vulnerable to quantum attack, Litecoin would be at best be vulnerable in the short term.

4.3.4 Monero

The primary mechanism for attacking Monero was found to be similar to that of Bitcoin. Once a transaction is broadcast to the network, if a quantum device can solve the broadcast public key for the private key, then a competing transaction can be created. This is a major vulnerability for Monero. However, as like Bitcoin it utilises UTXO based transaction structures, public keys are frequently recycled and generally not used for multiple UTXO's. Due to the use of RingCT in Monero it would also be impossible to detect when this attack is being performed. When compared to Bitcoin, it is clear when a competing transaction is created. However, due to the obfuscation of who creates a transaction in Monero, this would be significantly more difficult. However, the user will be required to perform additional computation to unmask the users account that they are targeting within the ring.

Introducing zk-proofs to blockchain technologies presents a different proposition when compared to previous analysis. Transactions on the Monero network are obfuscated, this means that a potential attacker would not know the value of the transaction. This would mean that a slightly more complex attack mechanism would be required. Bulletproofs are vulnerable to quantum attack, as it is reliant on the difficulty of the discrete logarithm problem. An attacker could therefore attack UTXO's blindly, although this may prove unprofitable for the attacker when comparing the value of the hijacked transactions to that of the cost to run the quantum device. This could also lead to detection before an attack of significant value is performed. Alternatively, an attacker could target many transactions, unmasking the values held within. This step would allow the user to remain anonymous. Upon finding a transaction that is of high value, that transaction could be targeted.

The RandomX consensus algorithm utilised by Monero could potentially be more quantum resistant. RandomX is based on the execution of random programs. Grover's algorithm would therefore have limited impact on this consensus algorithm as these programs are done over special instruction sets.

The use of zk-proofs does add some resilience to the Monero network, as there would be more computational steps for a quantum attacker to succeed. However, due to the underlying cryptographic primitives being vulnerable, they would still be vulnerable to attack.

4.3.5 Z-Cash

Zcash is vulnerable to quantum attacks in three key areas:

Firstly is the consensus algorithm used by Z-Cash is vulnerable, as shown by Grassi *et al.* (142) developed a quantum algorithm for that k-xor problem (a generalized birthday problem) that surpasses Wagner's classical algorithm. This quantum algorithm offers improved time and memory complexity of $(O\left(2^{n/(2+\lfloor\log_2(k)\rfloor)}\right))$ versus Wagner's $(O\left(2^{n/(1+\lfloor\log_2(k)\rfloor)}\right))$. This would give quantum devices an advantage against classical ones when acting on Z-Cash's PoW. However, it is not considered a primary threat.

Secondly is the digital signature scheme utilised by Z-Cash is reliant on the discrete logarithm problem, and therefore transactions broadcast to the network will be vulnerable to hijack. Like with Monero the obfuscated transactions have some resilience, however this is negligible.

Finally, is the most damaging attack presented in this chapter. Quantum attack on the global public parameter of Z-Cash's zk-SNARKs. This public parameter is created in such a way that no single individual should ever hold the corresponding public key. The security of this is reliant on the assumption that there is at least a single user who destroys their part of the key and also that there is no mechanism for recovering the private key from the global public key. However, this process is reliant on the hardness of the discrete logarithm problem. This means that a quantum attacker could target the global key and solve it for the private key. Utilising this key, an attacker could create Z-Cash tokens at will. Even more damaging for the network is that no-one would even see it happening, as the created tokens could be obfuscated.

Due to the potential for a quantum attacker to be able to print tokens at will, this is the most damaging attack that has been analysed. This could render the blockchain and its underlying economy completely redundant and with little way to detect or counteract it.

4.3.6 Others

While the analysis that has been performed on the blockchains in this subsection is not as in depth as the other, it serves to demonstrate the technology reuse across the industry. This reliance on these key cryptographic principles has left almost all networks vulnerable to attack from Shor's algorithm on their digital signature schemes.

Grin

Both the signature scheme and the obfuscation mechanism that are used by Grin are vulnerable to quantum attack from Shor's algorithm, as they are reliant on the hardness of the discrete logarithm problem. However, like other privacy-based blockchains, the obfuscation of the value of transactions means that there is some gained resistance. Grover's algorithm will show some speed-up for the cuckoo cycle PoW mechanism; however, this is not considered a primary attack vector.

BEAM

Like other privacy-based blockchains, BEAM is vulnerable in both its digital signature scheme and obfuscation mechanism, as both are reliant on the hardness of the discrete logarithm problem.

Bitcoin Cash

As Bitcoin Cash utilises most of the same infrastructure, being a hard fork of Bitcoin, it will suffer from the same vulnerabilities. It does, however, focus on increased throughput of transactions, meaning that transactions may be waiting in the transaction pool for inclusion in a block for less time. However, this will only have a negligible effect when a quantum attacker is targeting a transaction.

The PoW difficulty parameter and hash power acting upon the network is also considerably lower (143); therefore, it is more susceptible to Grover's algorithm attacks than Bitcoin.

Bitcoin Gold

Bitcoin Gold as a direct hard fork of the Bitcoin blockchain re-uses much of the technology. Therefore, it shares all the same vulnerabilities. Bitcoin Gold does utilise a slightly different consensus algorithm; however, this will see some speed-up from Grover's algorithm. It, however, is not considered a major attack vector.

Bitcoin SV

Bitcoin SV as a hard fork of Bitcoin Cash which is in turn a hard fork of Bitcoin, shares the same vulnerability as Bitcoin. It does, however, have a shorter block time. While this will reduce the time period over which a quantum attack can attack, it will not make a difference eventually, and therefore it is quantum vulnerable.

Chapter 5

Quantum Advantage on Proof of Work

Grover's algorithm as described in Section 1.4 is the original quantum search algorithm. This means it can search through an unstructured search space with a significant speed-up when compared to a brute force search. This has the potential for a wide range of uses, as this encompasses all problems that can be turned into search spaces. Within the context of blockchain technologies, the particular use case that is considered is the use of Grover's algorithm on a blockchains PoW where a quadratic advantage is gained.

Grover's algorithm can search an unstructured search space with a quadratic speed-up when compared to brute force methods. Compared with the exponential speed-up achieved by Shor's algorithm, Grover's quadratic speed-up has a more modest practical impact, even though it applies to a broader range of search-type problems. For example, if using Grover's algorithm to search for a 256-bit private key from a public key, it would take up to $\sqrt{2^{256}}$ operations (60). Meanwhile, for a quantum computer, if the problem is based on integer factoring, or similar problems, Shor's algorithm can find a solution in at most $(log\ 2^{256})^3$ operations (16). Examples of these can be seen in Fig. 3.

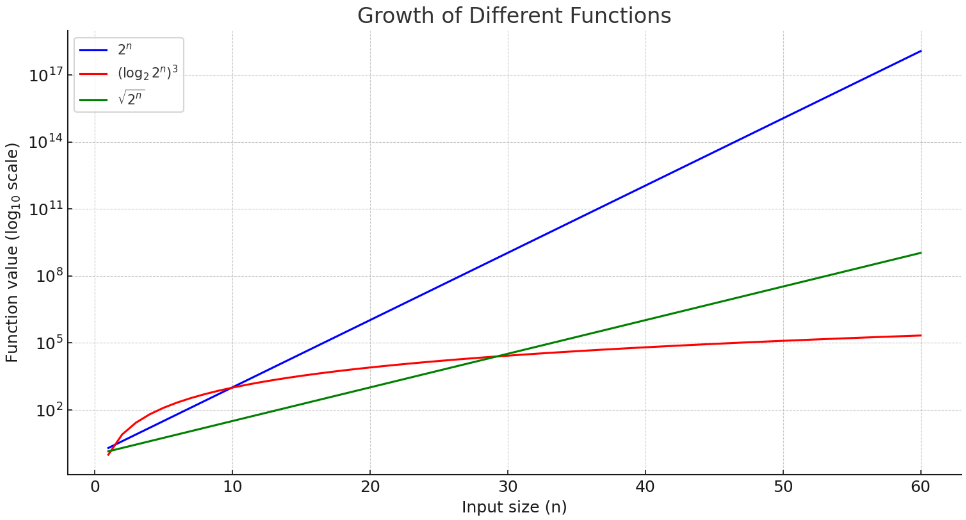

Figure 3: **Input of n vs. Output** This graph shows for inputs of n up to 60. It plots for the complexities 2^n, $\sqrt{2^n}$ and $(log2^n)^3$. These values represent a brute force attempt at solving a problem, a quadratic advantage and an exponential advantage, respectively.

Previous research stated that for blockchain technologies, the primary target for attack was assumed to be Shor's algorithm for targeting digital signatures (15; 1). Furthermore, quantum attack on Proof of Work (PoW) was not considered feasible in the short to medium term. However, this assumption was incorrect, as the PoW problem **must** be solvable within a reasonable time frame on classical infrafastructures. This is unlike the example of finding a private key from a public key. While the hash rate on major blockchains such as Bitcoin is very high, a quadratic advantage over the opposition is significant. Within the next two decades, it is conceivable that a single quantum-capable adversary could amass enough hashing power to launch a quantum-assisted 51% attack against Bitcoin, the blockchain with the largest aggregate computational power.

This led to the question under what circumstances would a quantum device be able to perform a 51% attack on the Bitcoin network. The reason for choosing the Bitcoin network is that it has the most computational power acting upon it at any one time. This means that if a quantum device is capable of attacking that network, it would be capable of attacking any PoW-based network. Bitcoin is also the largest blockchain by capitalisation and has the highest value block reward of any blockchain.

This chapter will cover the results presented in the paper 'Quantum Advantage on Proof of Work' which was published in September 2022 (2).

While a single user with a sufficiently powerful quantum device could attack PoW, smaller scale devices could also be used to generate increased profits. This chapter considers the use of quantum devices not just as a threat to PoW, but as a more efficient version of miners. It therefore introduces the concept of quantum cryptocurrency miners. These devices are more akin to the classical ASIC devices used to mine cryptocurrency currently. These devices would not necessarily run Grover's algorithm to completion, rather to a specified probability. This would be within a reasonable time frame, controlled by the blockchain protocol's block time. It is considered what the impact of the introduction of quantum devices into a PoW network would do, and what would be the effects of a continued influx of these new mining devices. It will also be discussed what the potential profit calculations for mining pools that switch to quantum-based infrastructure and what the requirements of for greater profitability would be. This analysis could prove useful for mining pool looking to convert to quantum-based mining infrastructures in the future.

To calculate the risk and rewards of using quantum devices on the PoW mechanism of blockchains, firstly PoW must be clearly defined.

5.1 Defining Proof of Work

Early cryptocurrencies faced the task of solving the problem of ensuring that a dishonest minority of stakeholders cannot impact an honest majority. The Byzantine Generals problem (144) is a game theoretic approach that describes the difficulties faced by decentralised systems and the impacts of unreliable or malicious nodes. In the context of cryptocurrencies, it is an unreasonable expectation that all nodes are both honest and present at all times. This problem was the critical element that was solved by Satoshi Nakamoto (7) when creating Bitcoin. The central idea of this was that malicious nodes must be punished for bad actions, and therefore it is in the interest of mining nodes to act honestly. PoW does this by providing mining nodes with a problem to solve. This problem requires computational power to solve, this requires electricity and therefore costs money. Proof-of-work ensured that for users of the network, it is in the best interest of miners to act honestly. Dishonesty by miners causes any blocks that they create to be rejected by the network, and any rewards therefore gained to be lost. More critically, however, the dishonest miner will be worse off as a result of the energy expenditure from solving PoW. To analyse the impact quantum devices will have on a PoW system, a generalised version of PoW must be defined. This allows us to analyse all PoW systems, not just the Bitcoin implementation.

While most blockchains utilise some kind of hash-based problem, in reality any difficult problem could be utilised as long as it follows the equation.

$$TC_v << TC_s \tag{1}$$

Where TC_v is the time to verify a solution and TC_s is the time to solve the problem. In the case of hash-based problems, the time complexity to verify is $O(1)$ while the time to solve is $O(2^n)$ where n is the number of leading bits required to be 0. While other problems may have differing time complexities, as long as the time complexity to solve is much greater than that to verify, it can be used in PoW. Some blockchains include some element of space complexity as well (131) however this is not necessary for a PoW algorithm.

Secondly, we require the problem to be scalable. If the search space for the problem is static, then as more computational power enters the network, the problem becomes increasingly easy and eventually even trivial. Blockchains generally have a static block time, for example 10 minutes between new blocks in Bitcoin. This 10-minute block time is created by a tunable difficulty of the problem. Giving us the implication:

$$(CP \uparrow \implies PD \uparrow) \wedge (CP \downarrow \implies PD \downarrow) \tag{2}$$

Where CP is an increase in the computational power acting to solve the problem and PD is an increase in problem difficulty.

For most PoW-based blockchains, the average time it has taken for each block to be produced is checked intermittently. At this point, if this average time is less than the required block time, then the difficulty is increased. If the average time is higher than the block time, then the difficulty is decreased.

Using these two statements, it allows us to create the following definition of PoW (2):

Definition (Proof of Work)**.** *A computational problem can be considered as a PoW problem if it satisfies the following two requirements*

1. *The computational complexity of the problem must satisfy equation 1,*

2. *The difficulty of the problem must be easily* tuneable *with a parameter to satisfy equation 2.*

It must be noted that this definition does not consider any of the rules and restrictions for the blockchain protocol. This includes how a miner or user determines which chain to follow or what constitutes a bad actor.

5.2 Quantum Advantage for PoW

As previously mentioned, research, particularly by the papers by Aggarwal et al. (15) and Kearney et al. (1) did not consider Proof of Work to be a feasible target for a quantum attacker. The exponential speed-up gained by Shor's algorithm on the cryptographic principles that underpin the digital signature algorithms appeared a stronger attack vector. In comparison, the mere quadratic speed-up that Grover's algorithm offers against proof-of-work hashing was considered too modest to pose a realistic threat in the near term. However, this did not fully consider the fact that the PoW problem is inherently solvable by a classical device. Specifically, it must be solvable by the combined effort of the network computational power in around the designated block time. The difficulty will scale with the size of the network. This is in stark contrast to the difficulty of the problems that underpin ECDSA, which are solvable in sub-exponential time on a classical device. While it is true that at its most difficult PoW would be approximately as difficult as solving ECDSA, it is unlikely that PoW problems for any blockchain would ever reach this difficulty due to the computational power that would be required to solve it within the block time.

By using Grover's algorithm, the Bitcoin implementation of PoW can be reduced in time complexity from $O(2^n)$ to $O(\sqrt{2^n})$. This significant advantage could potentially allow a single quantum entity to control more than 51% of the computational power. To calculate what the requirements of this would be, the following assumptions must be made:

1. The quantum attacker is the only quantum device acting upon the network; therefore they are the only ones with a quadratic advantage.
2. Bitcoin's computational power will follow Moore's law (145)
3. From the present day and into the future, quantum devices will obey Moore's law.

These assumptions can be considered reasonable. For 1) there will be a point where there is a single quantum device acting on the network. It will take a reasonable amount of time for other users to catch up with the technology. 2) Bitcoin's computation power has roughly doubled every 18 months or so. Due to price fluctuation and therefore incentives for miners, it does frequently crest and fall; however, it has followed a generally upwards trend. 3), this is probably the most difficult to justify due to the infancy of the technology; however, the industry is generally in an upwards trajectory currently exceeding the quantum equivalent of Moore's law.

Next, the initial state of both systems must be considered. At the time of the analysis, the Bitcoin network had a total hash rate of $130 \times 10^{18} H/s$. Moore's law is then applied to this, meaning that the hash rate will double every 18 months. This assumption we can see has held as the hash rate at the time of writing is approximately $400 \times 10^{18} H/s$ over 2 years later.

Secondly, the quantum device is assumed to be starting at a clock speed of $40 MHz$ and follows Moore's Law, doubling every 18 Months. This device would take at least 512-qubits to search a 256-bit search space. This assumes there is

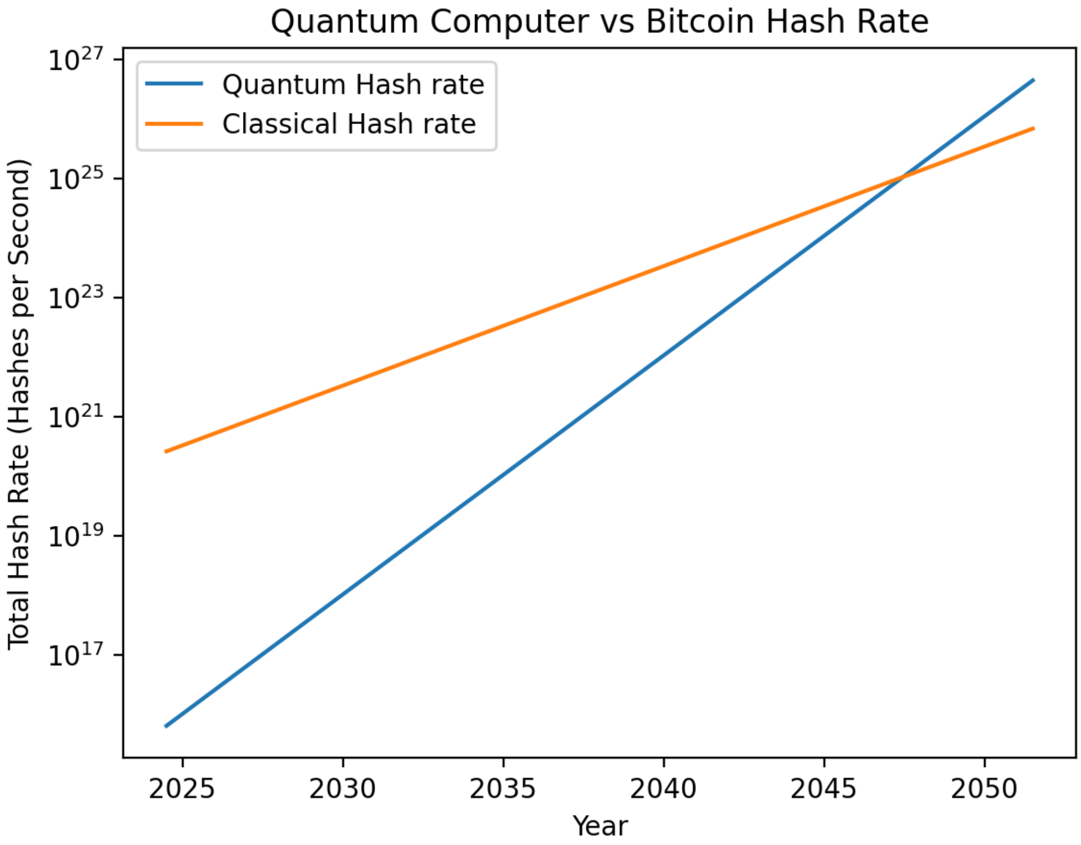

Figure 4: **Bitcoin network hash rate *vs.* single quantum computer.** The graph shows the hash rate growth over time of the entirety of the Bitcoin network, compared to that of a *single* quantum computer. Future data-points are extrapolated from current hash-rates, and assumes growth-rates for both quantum and classical technologies in line with current Moore's Law trends. See the main text for further details.

no error correction.

Figure 4 shows that with the quadratic advantage, we estimate a single quantum entity could mount a 51% attack against the bitcoin network by 2047.

51% attacks are specifically mounted against PoW-based blockchains. They are where a single malicious entity controls more than 50% of the network's computational power. This majority allows the attacker to disrupt the network in several ways, including:

- Double Spending — The miner can reuse previously used UTXO's. This would allow them to spend their cryptocurrency twice. This would mean that the blockchain is no longer immutable.
- Transaction Prevention — Transactions from other users could be blocked and be prevented from being included in the blockchain.
- Chain reorganisation — With a prolonged 51% attack, the miner could change the previous canonical history of the blockchain by creating a new longest chain. This could take a long time at greater block depths.

In general, however, it would erode the trust in the network. While this work only considers Bitcoin as the largest and most computationally powerful network, smaller PoW-based networks are significantly more vulnerable to this form of attack. Therefore, this timeframe could be significantly shorter for other, less popular networks.

5.3 The Profitability of Quantum Cryptocurrency Mining

While there is the possibility of a malicious actor utilising a quantum device to attack a blockchains PoW mechanism, they could also be used as a more efficient mining mechanism. PoW could potentially be a near term use case for a small scale quantum device, as will be discussed more in Chapter 6. In order for a quantum device and the outlay costs associated with them to be worthwhile, they must be more profitable than that of their counterpart classical devices.

5.3.1 Profitability Calculation

To calculate the profitability of a quantum device when compared to that of its classical counterpart, an equation is created to find the ratio to compare the profit of a network of quantum devices against a comparable classical device. Firstly, the probability of a classical device mining a block must be calculated. This value will be P_C.

$$P_C = \frac{H_C t}{\frac{\eta D}{t}} \tag{3}$$

The value H_C is the hash rate of the classical device, t is the block time in seconds (600s) for Bitcoin. D is the difficulty of the blockchain network and η is the hash in bits (Most blockchains use a 256-bit hash function, e.g. SHA-256).

$\frac{\eta D}{t}$ is the calculation for the total network hash rate of any one network. This can then be simplified to:

$$P_C = \frac{H_C t^2}{\eta D}, \tag{4}$$

H_C is derived from the hash rate of any one device divided by the total network hash rate (146). For quantum devices the probability for calculating a block is done similarly, however due to the use of Grover's algorithm the difficulty of the problem can be considered quadratically smaller. Due to this advantage, the probability of mining a block on any quantum device based on a given equivalent hash rate can then be defined as:

$$P_Q = \frac{H_Q t^2}{\eta\sqrt{D}} \tag{5}$$

Where P_Q is the probability of mining a block from a quantum device and H_Q is the equivalent hash rate of that device.

Together, P_Q and P_C give us the probabilities of a quantum or classical miner respectively to mine a specific block. This can then be adapted over a time period to calculate the amount of block rewards that a miner would expect to gain. For ease of comparison between blockchains, we also introduce a function f which converts the cryptocurrency to USD. Finally, it must be considered what the block reward is. Block rewards for cryptocurrencies vary. However, specifically for Bitcoin, the value of reward started at 50 BTC and decreases by half every 210,000 blocks(147). This means the reward is currently 6.25 BTC, with a further decrease expected some time in April 2024 to 3.125 BTC(148).

This allows us to calculate the total income for a classical miner:

$$I_C = f\left(\frac{T}{t} \cdot P_C B\right), \tag{6}$$

where I_C is the income for a classical miner across the timespan T, and B is the block reward for the considered blockchain.

And for a quantum miner:

$$I_Q = f\left(\frac{T}{t} \cdot P_Q B\right), \tag{7}$$

where I_Q is the income for a quantum miner across T.

To calculate the profit of a device, the costs associated with the device must be considered. These are the set-up and the operating costs. The set-up costs are the cost of purchasing the device and are considered a one-time cost. Therefore, the longer the device is operational, the less this cost matters. The operating cost is the cost for a device to operate for a single block time. This is important when mining cryptocurrencies, as the consumption of energy is critical to the operation of PoW. This will be expanded upon in Chapter 6 how relevant energy consumption is for a quantum miner.

From this, the profit returns for classical miners can be determined as:

$$R_C = I_C - (T \cdot O_C) - S_C, \tag{8}$$

where R_C is the profit, O_C is the operating costs and S_C is the setup costs for the classical device. The profit calculation for quantum miners is as follows:

$$R_Q = I_Q - (T \cdot O_Q) - S_Q, \tag{9}$$

where R_Q is the profit, O_Q is the operating costs and S_Q is the setup costs for the quantum device.

From these two equations, a profit ratio (G) can be calculated:

$$G = \frac{R_C}{R_Q}. \tag{10}$$

For the above equation once, $G > 1$ this means it is more profitable to mine Bitcoin on a quantum device rather than classical infrastructures. It must also be noted that this equation does consider other quantum devices being present in the network. As more quantum devices would increase, the value D corresponding with the hash rate for the total network would increase.

Eq. 10 can be expanded, using the previous equations, too:

$$G = \frac{f\left(T \cdot \frac{H_C t}{\eta D} \cdot B\right) - (T \cdot O_C) - S_C}{f\left(T \cdot \frac{H_Q t}{\eta \sqrt{D}} \cdot B\right) - (T \cdot O_Q) - S_Q} \tag{11}$$

These equations give us the ability to calculate the profitability of classical devices compared to quantum devices, as well as the ability to calculate the profitability of each independently of each other. They can be used as a tool for prospective quantum miners to calculate the profitability of a device. These equations can be used on any PoW-based blockchain, meaning that analysis could be done on the most profitable networks for a quantum mining pool to target.

5.3.2 Scenarios and Forecasts

Using equation 12 derived from equation 9 the point at which a quantum device becomes profitable can be calculated.

$$f\left(T \cdot \frac{H_Q t}{\eta \sqrt{D}} \cdot B\right) - (T \cdot O_Q) - S_Q > 0 \tag{12}$$

Forecasts can therefore be made on the future of quantum devices as cryptocurrency miners. When predicting the future of two emerging technologies, some assumptions must be made.

First, it is assumed that quantum devices are not going to, certainly in the near term, be accessible by most of the population. It is assumed they will be the domain of nations and large tech firms such as IBM, Google, and Microsoft[1].

[1] This assumption will be challenged in 6 as we discuss the use of NISQ devices as small

Therefore, the assumption is that most users will access quantum computation through the use of cloud-based quantum devices (149; 150). This will mean a wider range of individuals will have access to them, and secondly, it also means that set up cost for quantum mining pools will be almost 0. Therefore, it can be assumed that our value $S_Q = 0$.

Secondly, IBM's quantum computing timeline (58) must be considered to be accurate. This means the first quantum device capable of running Grover's algorithm on a 2^{256} search space will exist in approximately 2025. This will mean the block reward throughout our calculations will be $B = 3.125BTC$.

Assumptions must also be made about the cryptocurrency price value for the function f. Within the analysis, calculations are performed on a range of values. High ($23,536.12) and Low values ($10,385.49) from at the time of writing the paper (102) as well as predicted high ($31,000) and optimal high($100,000). The higher values were chosen as over time the value of Bitcoin has trended upwards. It must be noted that at the time of writing this thesis, the value is closer to the optimal high value.

The difficulty of the blockchain must be calculated. This was done by taking historical difficulties of the Bitcoin blockchain and projecting them into the future using a polynomial curve of best fit. This gives us a difficulty value (D) of $D = 4.2903 \times 10^{18}$. The time over which the quantum device will run will be set as 1 year, therefore $T(s) = 31536000$.

scale cryptocurrency miners. Nonetheless, it should be noted that for the majority of users, cloud-based quantum infrastructures may be the sole means of utilizing this technology in the near to medium term

$H_Q(MHz/s)$	$f(USD)$	O_Q
40	23,536.12	6,258.27
40	10,385.49	2,761.51
40	31,000.00	8,242.92
40	100,000.00	26,590.06
640	23,536.12	100,132.28
640	10,385.49	44,184.12
640	31,000.00	131,886.68
640	100,000.00	425,440.90

Table 10: This table shows the income generated by a quantum Bitcoin miner, over the period of a year in USD in relation to a specified quantum computer clock speed (first column) and a fiat currency conversion (second column). In the third column O_Q is calculated with $I_Q = 1(USD)$

Our final assumption is the speed of the quantum device. One of Google's current quantum devices has a clock speed of 40MHz/s (64). Therefore, this will be our starting low value. We also consider quantum devices after 4 Moore's cycles for a second device at 640MHz/s.

These values are then used to calculate the maximum O_Q value at which, if we are below, a profit will be made. These results are detailed in Table 10 and Fig. 5. For example, in case where $f = 100,000USD$ and the quantum device is $640MHz/s$ then as long as $O_Q < 425,440USD$ per year, then we have made a profit.

5.3.3 The Effects of Introducing Quantum PoW Technology

The introduction of quantum devices to a PoW mining pool will have dramatic consequences for the makeup of the mining pools currently acting on the network. Upon the introduction of a profitable quantum device to the network, the cycle shown in Fig. 6 will begin.

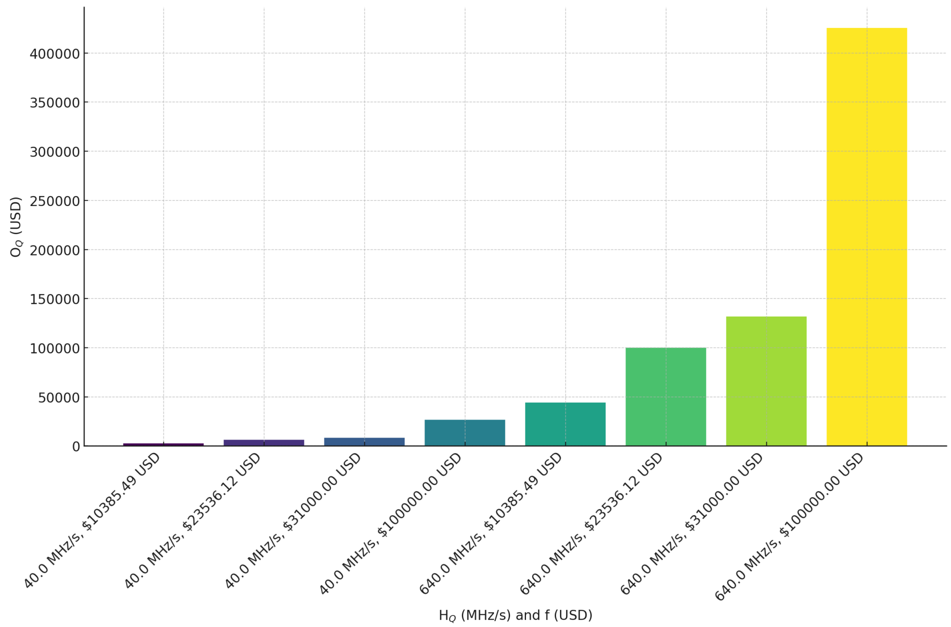

Figure 5: **Maximum Operating Costs (O_Q) as a Result of Price of Bitcoin (f) and Speed of Quantum Device (H_Q)** This chart lays out the details shown in 10. Colours are for clarity

Introducing a quantum device to the network will cause an increase in the difficulty parameter due to the increase in hashing power. This increase in hashing power will mean that the overall advantage for a quantum device is higher. This will also mean a higher profit margin for quantum devices. This increases the incentive for miners to transition to quantum-based infrastructures. This will lead to a further increase in quantum devices on the network. This will cause the cycle to repeat.

While this cycle will be slow at first, as more quantum devices enter the network the difficulty will spike. This will eventually mean that classical devices become non-competitive on the network as the difficulty parameter increases.

This will also lead to the PoW mechanism becoming secure against quantum attack, as if everyone has a quadratic speed-up, then no miner has a distinct advantage.

Overall, this raises the question which will occur first. A single quantum entity that can perform 51% attacks on PoW, or the transition to decentralised quantum mining infrastructures becoming standard across the network? Within this thesis and with the current state of the industry, it is difficult to answer this question. However, within chapter 6 is discussed a near term use case of small scale quantum computers that could be used as a profitable use case for quantum devices.

5.4 Discussion

The impacts for blockchain technologies laid out in this chapter are potentially more damaging than those found in Chapter 4. With the ability to solve PoW problems faster than the rest of the network combined, a single quantum entity could completely control and cripple a network. While we chose to concentrate only on Bitcoin, this applies to all blockchains with PoW-based consensus algorithms. Many other blockchain systems use PoW (151). In some cases, 51% attacks with comparatively modest quantum devices may be possible. This means that the timelines for attack could be much shorter. Mining is a financially expensive activity, and optimal profits are a major consideration for those with large amounts of computational power. Therefore, many miners choose to only focus their mining on major blockchains. This was, until recently, a disregarded threat to blockchain technologies. This type of attack raises serious concerns about the security of PoW and blockchain networks in general. Particularly in how they might have to adapt their consensus algorithms to preserve integrity.

It must be reiterated, within the confines of PoW there is no solution to the

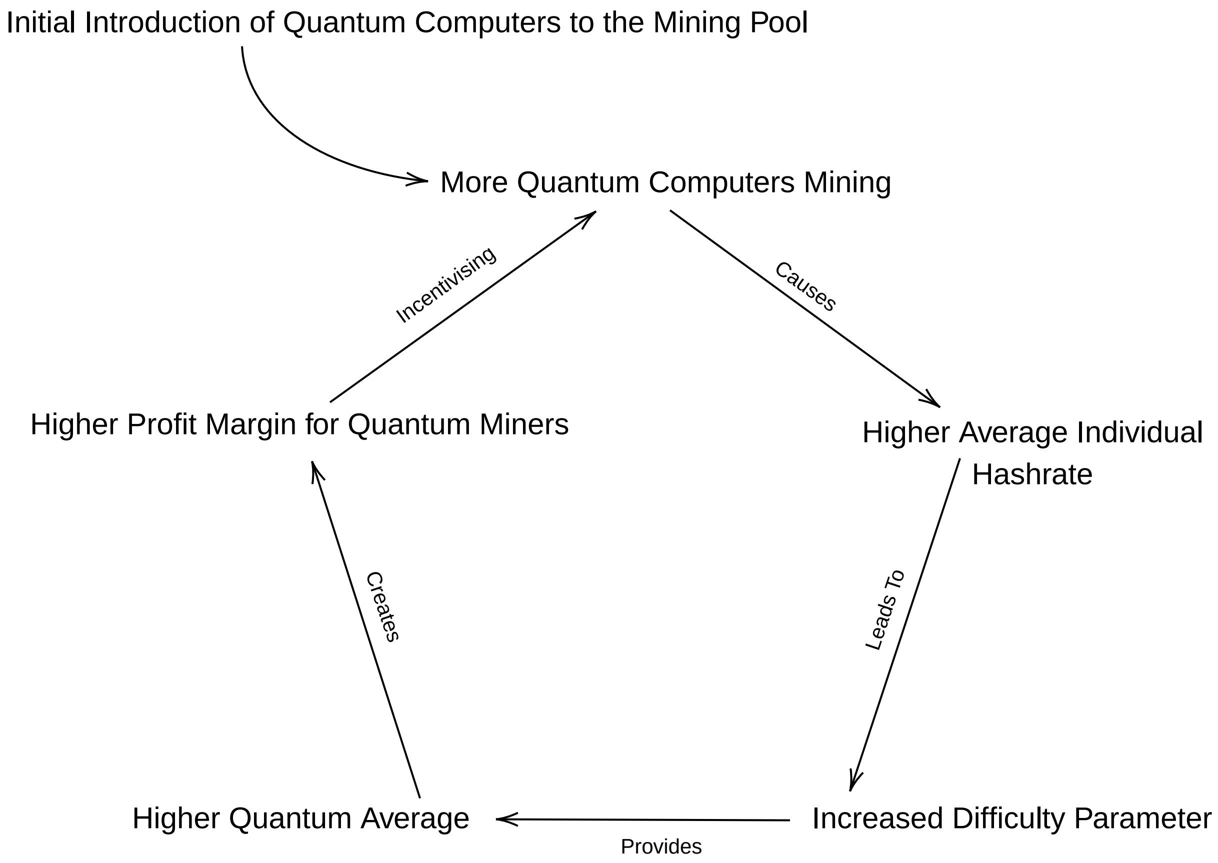
Initial Introduction of Quantum Computers to the Mining Pool
More Quantum Computers Mining
Causes
Higher Average Individual Hashrate
Leads To
Increased Difficulty Parameter
Provides
Higher Quantum Average
Creates
Higher Profit Margin for Quantum Miners
Incentivising

Figure 6: **Self-propagating cycle of increasing quantum advantage on PoW networks.** Adding quantum miners to the cryptocurrency network increases the network's hash-rate. An increased hash-rate will raise the difficulty parameter. An increased difficulty parameter increases the relative quantum-advantage. This, in turn, increases the profitability of quantum-mining, which in turn motivates the introduction of more quantum miners.

advantage that quantum devices will have upon it. Therefore, the only two solutions that are possible are to transition away from PoW or accept that at some point quantum devices will enter the network.

On the contrary, the novel notion of utilizing cryptocurrency miners exemplifies a compelling future application scenario for quantum devices. Although they have demonstrated the ability to solve intricate problems at speeds that are unfeasible on conventional devices, their practical applications in the upcoming decade are constrained. A specialised quantum circuit that applies Grover's algorithm to cryptocurrency mining could therefore represent a realistic short-to-medium-term application, even on today's noisy intermediate-scale quantum (NISQ) hardware. As Chapter 6 will demonstrate, the case becomes even stronger when the substantial energy savings of quantum-based miners are taken into account.

Chapter 6

Quantum Cryptocurrency Miners Allow for Massive Energy Savings

Chapter 5 demonstrated that quantum mining devices have the potential to give PoW miners a quadratic speed up in their capability to mine, and thereby give those with this speed up a greater profit margin. It is not unrealistic to believe that this could be a very early use case for quantum devices. This raised the question of, are there any other properties of quantum devices that would prove advantageous to PoW miners?

PoW is the first consensus algorithm utilised by blockchains allowed them to flourish. Providing the ability for blockchains to transact cryptocurrency without the need to trust other nodes on the network allowed them to be truly decentralised. While initially this system scaled well, as blockchains became more popular and more profit was possible, the amount of computational power being dedicated to PoW mining increased dramatically. This has led to the Bitcoin blockchain that uses amounts of energy comparable to those of medium-sized countries (152). We live in a world where carbon emissions and global warming are at the forefront of almost all business decisions. As a modern technology designed to challenge the paradigm's of modern business security through the consumption of energy has been widely criticised (48). This energy consumption is only set to increase as the value of the market as a whole increases. While some blockchains such as Ethereum have moved from PoW to PoS most blockchains are still entrenched in using PoW.

This chapter will show that through the use of quantum devices, potentially up to a 99.999% energy saving could be realised. This would reduce the energetic consumption by approximately the annual energy consumption of Sweden. While quantum devices certainly provide a threat to the underlying integrity of blockchains and more precisely PoW, there is potential for them to be used in cooperation. This chapter also describes how small scale, purpose built quantum devices could provide a near term use case for quantum computing in general. As a whole, the field lacks near term use cases and this could potentially provide one that is profitable as well as providing a marked energy reduction for mining cryptocurrencies.

The research presented in this chapter presents some of the results from the paper (3).

6.1 Blockchains and Energy Consumption

PoW-based blockchains have come under serious scrutiny for their excessive energy usage. While consensus algorithms are critical to blockchain technologies and their functionality, due to the popularity of blockchains such as Bitcoin, the profit available from the expenditure of energy to mine Bitcoin is extremely high. While there is some argument that PoW utilises renewable energy and even promotes increased investment in renewable energy (153), it can equally be argued that the renewable energy used by them could be utilised elsewhere as there is currently only a limited amount of renewable energy. Despite this argument, reduction in the energy consumption of PoW-based blockchains while maintaining the security of the networks can only be considered a positive for the industry and the environment.

6.2 Results

Presented in this section is the key takeaway results of our research. Considering several quantum miner infrastructures and present the results in tables 13 and 14. This is considered against the Bitcoin network, as it is the largest network in terms of computational power acting upon it. The quantum infrastructures we consider are comparable to modern-day ASIC. This ASIC would have a probability of mining a block of approximately 0.00007%. This was calculated by taking the hash rate of the device and comparing it to the total hash rate of the network.

The classical device that is considered as our baseline for comparison is the Antminer S19 XP (154). The cited hash rate of this device is 140 Tera Hashes (TH) / s. At the time of writing, the total network hash rate was approximately 200 million TH/s. This value has increased dramatically since. This gives the probability of successfully mining a block for this device in isolation as $7 \times 10^{-}7$.

Three quantum devices are considered, firstly with no quantum error correction, secondly with 1-layer Shor code error correction and finally 2-layer Shor code error correction. This error correction greatly affects the tolerated gate fidelity. The maximum error rates for each device are shown in table 11.

These devices and their corresponding error rates consider a quantum device that is running Grover's algorithm for enough Grover's iterations so that, it has a probability of approximately 1 of finding the solution. As stated, we only need

Infrastructure	**Physical Error Rate**
Non-ECC NISQ	1×10^{-10}
1 Layer ECC Quantum	1×10^{-8}
2 Layer ECC Quantum	1×10^{-4}

Table 11: **Highest Possible Error Rate** This table shows the infrastructure in the first column and the highest possible gate error rate in order for the Grover's algorithm to run successfully.

a quantum device that matches the probability of 0.0000007. The results for Grover's algorithm ran to completion are shown in 13. By reducing to our target probability, we can reduce the number of logical quantum gates for a 2 layer-error corrected device from 6974925312 to 3383552. This is a 2000-fold decrease in the number of gates required. This also reduces the required gate error rate dramatically, as shown in table 12.

Infrastructure	**Physical Error Rate**
Non-ECC NISQ	1×10^{-8}
1 Layer ECC Quantum	1×10^{-5}
2 Layer ECC Quantum	1×10^{-2}

Table 12: **Highest Possible Error Rate for a Probability of 0.0000007** This table shows the infrastructure in the first column and the highest possible gate error rate in order for the Grover's algorithm to run to a probability of 0.0000007 successfully.

Both tables 13 and 14 show a dramatic reduction in energy consumption when mining with quantum devices. A device running two layer Shor code would consume 2.9428^8 less energy than a classical device, assuming the same level of inefficiency between quantum and classical devices. This means that the inefficiency of the device would have to be 5.0204×10^{11} times greater than that of a classical device to consume the same amount of energy.

When considering a quantum device without error correction, the results are even greater. Each block on the Bitcoin network requires approximately 2.256kJ to mine, this would be reduced to 2.44572×10^{-15}J. This is a 1.5758×10^{21} times energy advantage over its classical counterpart.

Infrastructure	**Landauer Theoretical Minimum (J)**	**Real-World Energy Cost (J)**
Classical	1.324	2258.69
Non-ECC NISQ	1.43×10^{-18}	1.43×10^{-15} (*est.*)
1 Layer ECC Quantum	3.75×10^{-10}	6.4×10^{-7} (*est.*)
2 Layer ECC Quantum	4.5×10^{-9}	7.68×10^{-6} (*est.*)

Table 13: **Full-Grover Miners Compared.** The first row shows known real-world energy consumption values for a classical ASIC-based miner. The top-left value is the theoretical optimal bound. The second value is the real-world energy costs calculated here—taking the ratio between these values, 1 : 1706, allows us to estimate real-world costs of quantum miners from their respective theoretical minimums. The values in this table are for quantum miners that run Grover's algorithm to completion, to obtain a success probability near one.

6.3 Methodology

Energy consumption of a given classical device is a known value. However, as the quantum devices that are described are theoretical, there are no known energy consumption values for them. Therefore, a methodology to estimate the consumption must be created. While the actual energy consumption is not known,

Infrastructure	**Landauer Theoretical Minimum (J)**	**Real-World Energy Cost (J)**
Classical	1.324	2258.69
1 Layer ECC Quantum	7.579×10^{-14}	1.293×10^{-10} (*est.*)
2 Layer ECC Quantum	6.063×10^{-13}	1.034×10^{-9} (*est.*)

Table 14: **Partial-Grover Miners Compared.** The first row shows known real-world energy consumption values for a classical ASIC-based miner. The top-left value is the theoretical optimal bound. The second value is the real-world energy costs calculated here—taking the ratio between these values, 1 : 1706 allows us to estimate real-world costs of quantum miners from their respective theoretical minimums. The values in this table are for quantum miners that run Grover's algorithm with enough iterations to obtain a success probability similar to the classical miners of 0.000070%

by following Landauer's Principle, we can calculate the theoretical minimal energy consumption of a device. This theoretical minimal is unlikely to ever be achieved, or even approached upon. The theoretical minimum energy consumption of a classical device and its known classical value can be used to calculate a ratio. This ratio shows the inefficiency of the classical device compared to its theoretical minimum. This can then be applied to the quantum theoretical minimal to give us a predicted energy consumption of the quantum device. This assumes that the quantum device is as inefficient as its classical counterpart. However, the quantum device would have to be orders of magnitude more inefficient than a classical device to consume the same amount of energy, to calculate PoW with the same probability.

6.3.1 Landauer's Principle and Quantum Error Correction

Within this subsection, a brief overview will be given of the two concepts that will be critical to calculating the energy consumption of quantum cryptocurrency

Infrastructure	**Equal Energy Cost Ratio**
Classical	1:1706
Non-ECC NISQ	$1 : 1.58 \times 10^{21}$
1 Layer ECC Quantum	$1 : 2.981 \times 10^{16}$
2 Layer ECC Quantum	$1 : 3.726 \times 10^{15}$

Table 15: **Equal Energy Cost Ratio.** This table shows what the ratio between the real-world energy cost and theoretical optimal cost would have to be for each device so that they all share the same energy costs. The ratio for the classical device is set to be the same as the actual calculated one. Then, for the various quantum devices, the required ratio to obtain the same energy cost as the classical device is shown. This table demonstrates that quantum devices can be several orders of magnitude less efficient than a classical device, and still retain an energy-cost advantage. These values are for the Partial-Grover implementations.

miners. These are Landauer's Principle and Quantum Error Correction.

Landauer's Principle

Landauer's Principle (155) is used to describe the relationship between computation and energy consumption. It describes how irreversible actions on information cause energy expenditure. Particularly, the deletion of data. This means the deletion of bits, from a state where the value of the bits is known to an unknown state, means an increase in entropy. This increase in entropy requires some minimal amount of energy.

The minimal amount of energy required to delete a single bit (E) is calculated through the following equation:

$$E \geq k_B T ln 2 \tag{13}$$

Where k_B is the Boltzmann constant and T is the absolute temperature of the circuit at which the erasure is taking place. The energy E is given off as heat.

The Boltzmann constant relates the average kinetic energy of particles in a gas with the temperature of the gas. It has a value of 1.380649×10^{-23} Joules per Kelvin (156).

This deletion can be any logically irreversible action. This principle impacts all computing, as our devices becoming smaller and faster means bit erasure becomes a major issue due to the increase in entropy in a smaller space.

This has given rise to reversible computing (157). By creating devices where actions are logically reversible, the device can theoretically be reset without the need for data deletion and therefore no energy consumption.

We utilise Landauer's principle to calculate the minimal energy consumption for our devices. This is used to calculate energy consumption values for quantum devices.

Quantum Error Correction

Quantum noise is a major concern when building quantum devices (55). This is caused by the interaction of the quantum systems with their environments. This causes decoherence of quantum devices. Decoherence leads to errors in the code, phase and quantum bit flips, or loss of quantum information. This can be combated is several ways; however, the focus will be on the correction of errors through Quantum Error Correction (QEC) (158).

QEC aims to mitigate these issues. Like classical error correction, it aims to protect data in the system from errors and allow successful and accurate computations. Each gate has a probability of performing its action successfully, this is called gate fidelity.

While we will not discuss quantum error correction in depth, for error correction of the devices proposed here we will assume the use of Shor Codes (159). Shor codes work through encoding each logical qubit to 9 physical qubits. These qubits are entangled in such a way they represent the information stored exactly on the original logical qubit. Ancilla qubits are then created that are initialised in a known state (either $|0\rangle$ or $|1\rangle$). These are then entangled with the physical qubits. Therefore, any errors in the physical qubits can be measured by through changes in the ancilla qubit. Through the use of measuring techniques there will be no effect on the quantum state as the physical and logic qubits are not measured. Both phase flips and quantum bit flip errors can be detected.

While devices with a gate fidelity of $\tilde{9}9.8\%$ have recently been created (70), replicating this over large circuits for long run times is increasingly difficult. Therefore, to reduce this error rate, error correction must be used by devices looking for longer and more complex runtimes.

6.3.2 Calculating Landauer Limit

Calculating the Landauer limit requires us to calculate the amount of data deletion performed by a device running the algorithm. It must be noted once again that the minimal energy consumption is not realistic or attainable, it, however, gives us a base from which we can compare devices. We must first calculate the Landauer limit for the classical device. We will assume that the only deletion occurs when performing hash functions. We must then calculate the Landauer limit for our quantum devices. For a perfect quantum device, this is trivial. As quantum computation is reversible, only the initial state, i.e. the number of the logical and Grover's ancilla qubits must be deleted. For PoW, this will be 512 qubits. Finally, we calculate for error correction. As the ancilla qubits used in Shor code must be read and deleted, this will add significant overheads when compared to a perfect device. This is multiplied by the physical qubits that entangle to give the same state as the logical qubits.

Further to this, we must also consider how many gate interactions there are. This will change when looking at quantum devices searching at lower probabilities.

To calculate for the classical device, we must first find the amount of data deletion when performing a single hash function. Kim et al. (160) show that there are approximately 8588 NAND gates used to produce a single hash function. Each NAND gate erases 0.625 bits. This fraction reflects that, averaged over all possible input–output combinations, each gate discards only part of a bit rather than a full bit of information. Therefore, 5368 bits worth of information are lost per hash function. Per block our chosen classical device produces approximately, 84000 TH/block. This would be a minimal energy consumption of 1.324J per block.

Calculating for a quantum device that is perfect, meaning that it requires no error correction, will only require 512 qubits of data to be deleted. This is linear with the probability of mining a block, and so is a static value. When the Landauer limit is applied, this gives us an energy consumption of 1.4336×10^{18}J.

When calculating for error correction, there are many elements that must be considered. Firstly, we must calculate the number of Grover's iterations (t) required. For 100% Grovers, this is calculated by

$$t = \sqrt{\frac{N}{M}} \tag{14}$$

.

Where N is the total search space and M is the answer space. The answer space is calculated as

$$M = \frac{N}{d} \tag{15}$$

d is the current PoW difficulty. For our calculation, we will use the difficulty 29.693×10^{12}.

The search space size can therefore be calculated as:

$$M = \frac{2^{256}}{29.693 \times 10^{12}} = 3.8996 \times 10^{63} \tag{16}$$

The number of Grover's iterations for a probability approaching 1 is therefore:

$$G = \sqrt{\frac{2^{256}}{3.8996 \times 10^{63}}} = 5.44916 \times 10^{6} \tag{17}$$

To calculate for a specific probability, we must use the equation

$$P(t) \approx sin^2((2t+1)\theta) \tag{18}$$

Where P(t) is the probability of finding an answer and θ is dependent on the initial amplitude. This is calculated by:

$$\theta \approx arcsin(\sqrt{\frac{M}{N}}) \tag{19}$$

This can be rearranged to calculate t such that:

$$arcsin(\sqrt{P(t)}) \approx ((2t+1)\theta) \tag{20}$$

$$t \approx \frac{arcsin(\sqrt{P(t)})}{2\theta} - \frac{1}{2} \tag{21}$$

We can therefore calculate the number of Grovers iterations needed to get a probability of 0.0000007 as:

$$t \approx \frac{arcsin(\sqrt{0.0000007})}{2 \times arcsin(\sqrt{\frac{3.8996 \times 10^{63}}{2^{256}}})} - \frac{1}{2} = 2643 \tag{22}$$

Now that we have both values for the number of Grovers iterations required, we can calculate the total number of gate interactions (m). This is calculated as:

$$m = t \times g \times d \tag{23}$$

where g is the number of gates per Grover's iteration and d is the circuit depth.

Therefore, we can calculate the number of error correction interactions to be 1.11598×10^{10} for complete Grovers, and 5412864 for partial Grover's.

Finally, we can calculate the total number of erased qubits in a single block through the equation:

$$B = m \times c^n + q \tag{24}$$

where c is the number of error correcting measurements, n is the number of layers of error correction, and q is the number of initial qubits. This gives us the values in table 16

This number of bits can be multiplied by the Landauer equation shown previously to give the lower limits shown in Tables 13 and 14 in the second columns.

Infrastructure	**Full Grover Bits Deleted**	**Partial Grover Bits Deleted**
1 Layer ECC Quantum	1.339×10^{11}	6.4955×10^{7}
2 Layer ECC Quantum	1.607×10^{12}	7.7945×10^{8}

Table 16: **Bits deleted during run of Grover's** This table shows the number of Bits of information deleted in both a full run of Grover's and a partial run with a probability of 0.0000007

6.3.3 Error Rate Calculations

We can calculate the maximum error rates required for the process to complete with a reasonable probability. While we will repeat some steps from subsection 6.3.2, this gives us the full process by which we calculate the error rates for our devices.

Preliminaries:

- M - Total answer search space
- $N = 2^{256}$ - Total search space for Proof of Work
- d - current difficulty
- G - Number of grovers iterations
- n - total number of gate interactions when running grovers algorithm to a prob. of approx 1
- a - ancilla qubit number
- k - search space qubit number
- E_P - Error rate in physical qubits

- E_L - Error rate in logical qubits
- S_P - Success rate in physical qubits
- S_L - Success rate in logical qubits
- P - Probability of algorithm running without error.
- c - Number of layers of error correction

Equation 1 is used to calculate M the answer search space for proof of work:

$$M = \frac{N}{d} \tag{25}$$

This is:

$$M = \frac{2^{256}}{29.693 \times 10^{12}} = 3.8996 \times 10^{63} \tag{26}$$

To calculate the total number of Grovers iterations needed for an approx, prob. of 1 we calculate:

$$G = \sqrt{\frac{N}{M}} \tag{27}$$

This is:

$$G = \sqrt{\frac{2^{256}}{3.8996 \times 10^{63}}} = 5.44916 \times 10^{6} \tag{28}$$

We can then calculate the total number of gates interactions:

$$m = (a + k) + (a + (4 \times k)) \times G \tag{29}$$

This is:

$$n = (256 + 256) + (256 + (4 \times 256)) \times 5.44916 \times 10^6 = 6974925312 \qquad (30)$$

We must now define the error rate. Our error rate will change depending on the number of layers of error correction. For example, E = E at 0 layers, $E = E^2$ at one layer and $E = E^4$ at 2 layers. For the physical qubits, we will use the equation $S_P = 1 - E_P$. However, to account for Error correction, a more generalised equation. As with the physical qubits for logical qubits $S_L = 1 - E_L$. The error rate per gate when using error correction E_L can now be defined as:

$$E_L = E_P^{2^c} \qquad (31)$$

Given the number of gate interactions, m we can then calculate the probability of success of the algorithm using the following equation.

$$P = S_L^m \qquad (32)$$

So, calculating with an error rate of 1 in 100 Billion assuming that Grover's is performed to a probability of approx. 1, the success probability is.

$$P = 0.99999999999^{6974925312} = 0.9326 \qquad (33)$$

At an error rate of 1 in 100 million with one layer of error correction, we get the following error probability

$$P = (1 - (0.00000001)^2)^{6974925312} = 0.9326 \qquad (34)$$

For 2 layers of error correction:

$$P = (1 - (0.001)^4)^{6974925312} = 0.993 \qquad (35)$$

Rearranged to make S the subject:

$$S_P = \sqrt[m]{P} \tag{36}$$

For a probability of success at 0.0000007 with no error correction (just physical qubits) the required success rate per gate would be as follows:

$$S_P = \sqrt[69749253 12]{0.0000007} = 0.999999997968124 \tag{37}$$

To calculate for error correction, the following equation must be used:

$$S_L = 1 - \sqrt[2^c]{1 - S_P} \tag{38}$$

1 Layer:

$$S_L = 1 - \sqrt[2^1]{1 - 0.999999997968124} = 0.999954924 \tag{39}$$

2 Layers:

$$S_L = 1 - \sqrt[2^2]{1 - 0.999999997968124} = 0.9932861 \tag{40}$$

We now wish to calculate how many grovers iterations it would take to find, with a probability of 0.0000007. We use the following equation:

$$P(t) \approx sin^2((2t + 1)\theta) \tag{41}$$

Where θ is dependent on the initial amplitude. This is calculated by:

$$\theta \approx arcsin(\sqrt{\frac{M}{N}}) \tag{42}$$

All of this can be rearranged to calculate t such that:

$$arcsin(\sqrt{P(t)}) \approx ((2t+1)\theta) \tag{43}$$

$$a = ((2t+1)\theta) \tag{44}$$

Rearrange to:

$$t = \frac{a}{2\theta} - \frac{1}{2} \tag{45}$$

$$a = arcsin(\sqrt{P(t)}) \tag{46}$$

therefore:

$$t \approx \frac{arcsin(\sqrt{P(t)})}{2\theta} - \frac{1}{2} \tag{47}$$

Using our previous numbers, this is:

$$t \approx \frac{arcsin(\sqrt{0.0000007})}{2 \times arcsin(\sqrt{\frac{3.8996 \times 10^{63}}{2^{256}}})} - \frac{1}{2} = 2643 \tag{48}$$

This would be the new number of Grover's iterations.

We can therefore calculate the number of gate interactions as:

$$n = (256 + 256) + (256 + (4 \times 256)) \times 2643 = 3383552 \tag{49}$$

If we assume that there is a success rate of 0.99999 for each gate, we can calculate:

$$P = 0.99999999^{3383552} = 0.967 \tag{50}$$

Meaning that with no error correction and a physical error rate of 1 in 100 million, there would be a 96.7% chance of no errors for a $P = 0.0000007$.

With 1 layer of error correction and an error rate of 1 in 10000 the following can be calculated:

$$P = (1 - (0.0001)^2)^{3383552} = 0.9326 \tag{51}$$

Meaning that at that error rate and a $P = 0.0000007$ there will be no errors 93.26% of the time.

Finally, with 2 layers of error correction:

$$P = (1 - (0.01)^4)^{3383552} = 0.9667 \tag{52}$$

meaning that with an error rate of 1 in 100 there would be no error rate 96.67% of the time for a probability of $P = 0.0000007$

6.3.4 Efficiency Ratios

Efficiency ratios must be calculated for the classical devices, which we can then apply to the quantum minimal energy consumption. To achieve this, we must first take the energy consumption of the network as a whole. This was from information provided by the Cambridge Bitcoin Electricity Consumption Index (CBECI) (152). At the time of writing, Bitcoin had a total annual energy consumption of 126.7 TWh per year. This accounts for approximately 0.57% of the worlds' energy consumption. It must be re-iterated that this is the energy consumption for just one of the hundreds of PoW based cryptocurrencies. This amount of energy is comparable to that of Norway (124.3 TWh). It must also be noted that these value have dramatically increased since the research was originally performed.

By taking 0.00007% of the total energy consumption value, we can calculate the actual energy consumption of our ASIC device. We use ASICs as the classical baseline as they are purpose-built chips whose architecture is optimised solely for the hash computations used in proof-of-work, allowing them to achieve far higher throughput and energy efficiency than general-purpose CPUs or GPUs. This then reduced from a period of one year to 10 minutes for a single block equals an energy consumption of 2258.69J. The ratio (R) can be calculated as:

$$R = \frac{A}{M}$$

Where A is the actual energy consumption and M is the minimal energy consumption. This gives us:

$$R = \frac{2258.69}{1.324} = 1706$$

Or an efficiency ratio of 1:1706. We will also consider the manufacturer's stated energy consumption of 3.010kW, which works out to 502W/block. This would give us an efficiency ratio of 1:379.

Using the data in Tables 13 and 14, we can calculate how inefficient a quantum device would have to be as energy consumptive as the classical infrastructure. This can be found in table 15.

6.3.5 NISQ Devices

As described within this section, for a probability of 0.0000007 the number of gates can be reduced 2000-fold. This reduction means that algorithms can be run significantly faster, therefore giving the device less time to decohere. An example is that if we can run the device at 1×10^7 gates per second, as demonstrated by Noiri et al. (161). The algorithm will be completed in approximately $\frac{1}{3}$ of a

Protocol 3 This pseudocode displays the algorithm a quantum cryptocurrency miner would use to mine Bitcoin. This is quoted from (3)

```
Miner's Block Input:
    1. Version Number
    2. Hash of Previous Block
    3. Merkle Root Hash
    4. Time Stamp
    5. Difficulty Target $D$
    6. Nonce $i$
    7. Block Size
    8. Transaction Counter
    9. Transactions $T_1 \dots T_n$
Output Block $B$ such that $\mathfrak{H}(B) \leq B_{target}$
Steps
  1. Create block $B$ from Miner's Block input.
  2. Set $B_{target}$ through the calculation $2^{256} - (D \cdot 2^{32})$
  3. Set up Oracle $\mathfrak{O}$ that marks solutions where $\mathfrak{H}(B) \leq B_{target}$
  4. Set up $n$ qubits for the $n$ bit search space(generally $2^{256}$ or 32-Byte value for $i$)
  5. Initialize the system to a uniform superposition of all possible nonce values by applying Hadamard transform $H^{\otimes n}$
  6. Perform Grover's algorithm iterations, until the probability of finding a valid nonce is approximately 1. Each iteration consists of:
      (a) Apply $\mathfrak{O}$ over the search space
      (b) Apply Hadamard transform $H^{\otimes n}$
      (c) Perform phase shift such that $|0\rangle \rightarrow |0\rangle$, $|x\rangle \rightarrow -|x\rangle$ for $x > 0$
      (d) Apply Hadamard transform $H^{\otimes n}$
```

second. This opens up the possibility of creating NISQ devices. These specialised quantum devices can be considered akin to ASIC devices. They are designed in such a way they are capable of running a single algorithm. By running the device to an accepted probability, the error rate tolerance is much higher. This time frame of a third of a second is well within the range demonstrated by Bruzewicz et al. (162).

It is unlikely that full scale quantum devices are likely to be available for many more years, with error rates and the number of qubits required being a prohibitive factor. However, the use of quantum devices for small-scale algorithms that can be run quickly, such as on PoW, could be possible with minimal error correction. This presents an exciting near term use case for quantum devices, that could also be profitable.

Protocol 2 shows a simple pseudocode algorithm of how a NISQ cryptocurrency miner would operate.

6.4 Discussion

Within this section, we have expanded on the potential impacts that quantum devices can have on PoW based cryptocurrencies. In chapter 5 while showing the potential vulnerabilities, we looked at the utilisation of quantum devices for greater profit compared to classical devices. A reduction in energy consumption as demonstrated in this chapter only compounds the potential for greater profitability.

It is a fair assumption that quantum devices will be in the same order of magnitude as inefficient as classical devices, therefore a potential energy saving of 126.6TWh per year could be possible, or approximately 99.999% total energy saving. This would save enough energy to provide energy for Sweden for a year every year (163).

Even if this assumption does not hold, quantum devices would be required to be several orders of magnitude more inefficient than classical devices to consume as much energy. This is demonstrated in 15.

This presents a significant and profitable use case for NISQ devices. NISQ devices are designed to solve a single problem more efficiently than classical devices. As stated in chapter 5 PoW is not an infeasible problem even on classical devices. Therefore, with the gained efficiency of utilising quantum devices, and the reduced energy consumption, along with the fact that this is a relatively simple problem to design a circuit for, this presents potentially one of the most compelling use cases.

Chapter 7

Conclusion and Future Work

Within this thesis, we have looked at the effects that quantum devices of the future will have upon blockchain technologies. Both positive and negative. Chapters 4 and 5 discuss the vulnerabilities that blockchains have to quantum attack. While Chapters 5 and 6 proposed ways that quantum devices could actually benefit blockchain ecosystems.

We have shown that through the use of Shor's algorithm, user accounts for almost all blockchains existing are currently vulnerable to hijack. We demonstrated that the mechanism for quantum attack is affected by the blockchains' protocol; however, vulnerabilities persist for all. Given a sufficiently powerful quantum device with as few as 512 qubits, Shor's algorithm could be ran against elliptic curve based public keys to retrieve the private keys. In the context of blockchain this could have impacts ranging from a quantum attacker being able to target transactions as they are being broadcast to the network, to quantum attackers being able to create cryptocurrency from nothing.

This is a complicated problem within blockchain due to how integral a blockchains' cryptography is intertwined with its rules and protocols. Converting a blockchain to post-quantum will require community buy in from the blockchains stakeholders and tough decisions may have to be made.

An exponential speed against the underlying mathematical principles that underpin the cryptography of a blockchain would be extremely damaging. Therefore, this research heeds as a warning to the entire industry that action must be taken to avoid the potential of entire blockchain ecosystems being attacked by bad quantum actors. It is no secret that cryptocurrencies are not well liked by many governments around the world (164; 165; 166) and as governments are one of the major actors investing in quantum infrastructure (84; 85; 86; 87; 88; 89), it is not infeasible to see a situation in which a government may look to attack blockchain technologies.

Further to the attacks on a blockchains digital signature scheme, we have also shown the potential for a single quantum entity to attack a blockchains Proof-of-Work mechanism. PoW is vulnerable to a quadratic speed-up when run using Grover's algorithm. While there is a trend away from blockchains using PoW, it is still and stands to remain to be the most common mechanism for generating consensus on blockchains.

While it was originally thought the PoW was not a strong attack vector, we have shown that it could be possible for a single quantum entity to hold as much mining power as the rest of the network, therefore, having the potential to mound a 51% attack by the year 2048. This is because PoW has to be inherently solvable. Furthermore, there are no PoW mechanisms that are not vulnerable to a speed-up from Grover's algorithm.

This research then led to the question of what if quantum devices are used by honest actors on a PoW mechanism. We show that it is possible for greater profitability for miners that convert to quantum infrastructures due to the speedup gained. Furthermore, we show that the introduction of quantum devices on the network would cause a feedback loop. This could eventually lead to quantum devices being the only profitable and consistent way to mine cryptocurrency.

Finally, we showed how quantum devices could counteract one of the greatest problems of PoW-based blockchains. Their energy consumption. Through the use of quantum devices, a potential energy saving to the equivalent of some medium-sized countries could be possible.

Overall, we also show that quantum cryptocurrency miners present the opportunity for a near term, profitable use case for small scale error-prone quantum devices.

7.1 Future Work

One of the major questions that must be investigated is the process by which the current major blockchain technologies will transition to post-quantum secure cryptography. This process will take significantly longer for blockchains due to their decentralised governance. When it is compared with the process of a centralised system performing a similar upgrade, there are significantly more hurdles to overcome. The first and most obvious one is the cryptographic principles are so deeply imbedded into blockchains, as it's what makes blockchains what they are. By replacing them, there could be unintended side effects, such as what happens to those that do not have access to their cryptocurrency or are currently locked out. Users could be locked out for long periods of time, either due to their keys being held in 'cold storage' or even as a result of the blockchain protocol. An example could be a founder of a company creates a cryptocurrency token. They, as a result of their work, get a share of the token. As a result of the governance structure, those tokens could be locked within a smart contract for a period of time. The question would be, what if that period of time exceeds the time threshold of pre-quantum, i.e. there exists powerful quantum devices. Any transaction they perform after that fact could be vulnerable to attack. This is just one of many scenarios the stakeholders at blockchains need to be aware of and plan for. This is even further complicated by the scale of the industry with thousands of blockchains and hundreds of thousands of projects, how will it be possible to ensure they are all secure?

Further to this would be the re-analysis of the Ethereum blockchain. Since the original analysis laid out in chapter 4, there have been many changes to the Ethereum network. Furthermore, when considering the Ethereum network, we did not consider other attacks that could be performed on the Ethereum state machine. Examples of which could include attacks on the smart contract system, or on major smart contracts and DApps.

Another area that has yet to be studied is the attack vectors and how to upgrade Bitcoin and other blockchains P2P network protocols. While most post-quantum discussions focus on breaking ECDSA signatures or accelerating proof-of-work, the peer-to-peer layer that links Bitcoin nodes is an equally plausible target. A quantum-capable adversary that can derive private keys in minutes—or even seconds—could pair that ability with classic Sybil, eclipse, or mempool-flooding tactics to censor, reorder, or front-run transactions before they reach miners. Because Bitcoin's wire protocol still relies on unauthenticated gossip and (for now) unencrypted TCP links, a well-funded attacker could spin up thousands of counterfeit peers, occupy victims' connection slots, and exploit their quantum advantage to rewrite or invalidate transactions in flight. Despite these threats, rigorous modelling of quantum-assisted network attacks is sparse: there are no large-scale simulations that combine realistic Internet latencies, node-density assumptions, and projected quantum key-recovery times. Exploring these questions, how fast a 'quantum Sybil' could gain the majority of the relay capacity, what bandwidth or qubit budgets are required, and which protocol tweaks would neutralize the advantage, remains an open and important direction for future work.

Finally, an area of study that has been overlooked is the impact that quantum devices will have on smaller projects. While the market is dominated by Ethereum and Bitcoin, there are thousands of other smaller projects which could be more vulnerable, or less able to upgrade.

Overall, this field is novel and the research on how and when the industry will begin to shift towards quantum resistance is yet to be seen. However, as quantum devices are looming ever closer and it is becoming increasingly critical for the blockchain industry to act as there is a problem that requires proactivity rather than reactivity.

List of Acronyms

Acronym	Definition
ASIC	Application-Specific Integrated Circuit
BFT	Byzantine Fault Tolerance
CPU	Central Processing Unit
DAG	Directed Acyclic Graph
DPoS	Delegated Proof of Stake
ECC	Elliptic Curve Cryptography
ECDSA	Elliptic Curve Digital Signature Algorithm
EVM	Ethereum Virtual Machine
EdDSA	Edwards-curve Digital Signature Algorithm
FPGA	Field-Programmable Gate Array
GPU	Graphics Processing Unit
NISQ	Noisy Intermediate-Scale Quantum
NIST	National Institute of Standards and Technology
PBFT	Practical Byzantine Fault Tolerance
PKI	Public Key Infrastructure
PoA	Proof of Authority
PoS	Proof of Stake
PoW	Proof of Work
QCM	Quantum Computing Miner

Acronym	**Definition**
QFT	Quantum Fourier Transform
QRL	Quantum Resistant Ledger
RSA	Rivest–Shamir–Adleman cryptosystem
SHA	Secure Hash Algorithm family
SHA-256	Secure Hash Algorithm 256-bit
SHA-3	Secure Hash Algorithm 3
UTXO	Unspent Transaction Output
dBFT	delegated Byzantine Fault Tolerance

Bibliography

[1] Joseph J Kearney and Carlos A Perez-Delgado. Vulnerability of blockchain technologies to quantum attacks. *Array*, 10:100065, 2021.

[2] Dan A Bard, Joseph J Kearney, and Carlos A Perez-Delgado. Quantum advantage on proof of work. *Array*, 15:100225, 2022.

[3] Joseph Kearney and Carlos A Perez-Delgado. Quantum blockchain miners provide massive energy savings. *arXiv preprint arXiv:2306.03321*, 2023.

[4] CoinMarketCap. Cryptocurrency prices, charts and market capitalizations, 2024. Accessed: 2024-05-16.

[5] Fortune Business Insights. Blockchain technology market size — industry forecast [2032]. 2024. Accessed: 2024-05-16.

[6] Grand View Research. Blockchain technology market size & growth report, 2030. 2023. Accessed: 2024-05-16.

[7] Satoshi Nakamoto. Bitcoin: A peer-to-peer electronic cash system. *Decentralized business review*, page 21260, 2008.

[8] Primavera De Filippi and Samer Hassan. Blockchain technology as a regulatory technology: From code is law to law is code. *First Monday*, 21(12), 2016.

[9] Melanie Swan. Blockchain: Blueprint for a new economy. *O'Reilly Media, Inc.*, 2015.

[10] David Sharples and Rafael T. Porrata. Blockchain: Future of real estate? *Journal of Property Investment Finance*, 34(3):191–197, 2016.

[11] Merlinda Andoni, Valentin Robu, David Flynn, Simone Abram, Dale Geach, David Jenkins, Peter McCallum, and Andrew Peacock. Blockchain technology in the energy sector: A systematic review of challenges and opportunities. *Renewable and Sustainable Energy Reviews*, 100:143–174, 2019.

[12] Nir Kshetri. Blockchain's roles in strengthening cybersecurity and protecting privacy. *Telecommunications Policy*, 41(10):1027–1038, 2017.

[13] Asaph Azaria, Ariel Ekblaw, Thiago Vieira, and Andrew Lippman. Medrec: Using blockchain for medical data access and permission management. *2016 2nd International Conference on Open and Big Data (OBD)*, pages 25–30, 2016.

[14] David Yermack. Corporate governance and blockchains. *Review of Finance*, 21(1):7–31, 2017.

[15] Divesh Aggarwal, Gavin K Brennen, Troy Lee, Miklos Santha, and Marco Tomamichel. Quantum attacks on bitcoin, and how to protect against them. *arXiv preprint arXiv:1710.10377*, 2017.

[16] Peter W Shor. Algorithms for quantum computation: discrete logarithms and factoring. In *Proceedings 35th Annual Symposium on Foundations of Computer Science*, pages 124–134. IEEE, 1994.

[17] National Institute of Standards and Technology. Secure hash standard (shs). Technical Report FIPS PUB 180-4, National Institute of Standards and Technology, Gaithersburg, MD, USA, 2002.

[18] National Institute of Standards and Technology. Fips pub 202: Sha-3 standard: Permutation-based hash and extendable-output functions. Technical Report FIPS PUB 202, National Institute of Standards and Technology, Gaithersburg, MD, USA, 2015.

[19] Ralph C. Merkle. One way hash functions and des. In Gilles Brassard, editor, *Advances in Cryptology — CRYPTO' 89 Proceedings*, pages 428–446, New York, NY, 1989. Springer.

[20] Ivan Bjerre Damgård. *A Design Principle for Hash Functions.* PhD thesis, Aarhus University, 1989.

[21] Guido Bertoni, Joan Daemen, Michaël Peeters, and Gilles Van Assche. On the indifferentiability of the sponge construction. In Nigel P. Smart, editor, *Advances in Cryptology – EUROCRYPT 2008*, pages 181–197, Berlin, Heidelberg, 2008. Springer.

[22] Bruce Schneier. *Applied Cryptography: Protocols, Algorithms, and Source Code in C.* John Wiley Sons, Inc., New York, NY, USA, 2nd edition, 1996.

[23] Ronald L Rivest, Adi Shamir, and Leonard Adleman. A method for obtaining digital signatures and public-key cryptosystems. In *Communications of the ACM*, volume 21, pages 120–126. Association for Computing Machinery, 1978.

[24] Henri Cohen. *A Course in Computational Algebraic Number Theory.* Springer Science Business Media, 2013.

[25] Cybersecurity and Infrastructure Security Agency. Understanding digital signatures, 2021. Accessed: 2024-05-12.

[26] DocuSign. How digital signatures work, 2024. Accessed: 2024-03-19.

[27] Okta. Digital signatures: What they are & how they work, 2022. Accessed: 2024-04-02.

[28] Cloudflare. What is tls?, 2024. Accessed: 2024-02-09.

[29] Michael Sipser. *Introduction to the Theory of Computation*. Cengage Learning, 3rd edition, 2013.

[30] Stephen A. Cook. The complexity of theorem-proving procedures. *Proceedings of the third annual ACM symposium on Theory of computing*, pages 151–158, 1971.

[31] Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein. *Introduction to Algorithms*. MIT Press, 3rd edition, 2009.

[32] Ethan Bernstein and Umesh Vazirani. Quantum complexity theory. *SIAM Journal on Computing*, 26(5):1411–1473, 1997.

[33] Shafi Goldwasser, Silvio Micali, and Charles Rackoff. The knowledge complexity of interactive proof systems. *SIAM Journal on Computing*, 18(1):186–208, 1985.

[34] Oded Goldreich, Silvio Micali, and Avi Wigderson. Proofs that yield nothing but their validity, or all languages in np have zero-knowledge proof systems. *Journal of the ACM*, 38(3):691–729, 1986.

[35] Oded Goldreich. *Foundations of Cryptography: Volume 1, Basic Tools*. Cambridge University Press, 2001.

[36] David Chaum. Blind signatures for untraceable payments. *Advances in Cryptology: Proceedings of Crypto '82*, pages 199–203, 1983.

[37] EarthWeb. Cryptocurrency statistics, 2023. Accessed: 2023-04-13.

[38] Vitalik Buterin and Virgil Griffith. Casper the friendly finality gadget. 2017.

[39] Daniel Larimer. Bitshares 2.0 – financial smart contract platform, 2014.

[40] Miguel Castro and Barbara Liskov. Practical byzantine fault tolerance. *Proceedings of the Third Symposium on Operating Systems Design and Implementation*, pages 173–186, 1999.

[41] Da Hongfei and Erik Zhang. Neo: A distributed network for the smart economy, 2015.

[42] Igor Barinov et al. Poa network: Consensus by decentralized intrinsic assets, 2017.

[43] David Schwartz, Noah Youngs, and Arthur Britto. The ripple protocol consensus algorithm, 2014.

[44] Mohammad Rabiul Islam, Rizal Mohd Nor, Imad Fakhri Al-Shaikhli, and Kabir Sardar Mohammad. Cryptocurrency vs. fiat currency: architecture, algorithm, cashflow & ledger technology on emerging economy: the influential facts of cryptocurrency and fiat currency. In *2018 International Conference on Information and Communication Technology for the Muslim World (ICT4M)*, pages 69–73. IEEE, 2018.

[45] Jonathan Lane. Bitcoin, silk road, and the need for a new approach to virtual currency regulation. *Charleston L. Rev.*, 8:511, 2013.

[46] James Martin. Lost on the silk road: Online drug distribution and the ‘cryptomarket’. *Criminology & Criminal Justice*, 14(3):351–367, 2014.

[47] Lawrence Lessig. *Code: And other laws of cyberspace*. ReadHowYouWant.com, 2009.

[48] Max J. Krause and Thabet Tolaymat. Quantification of energy and carbon costs for mining cryptocurrencies. *Nature Sustainability*, 1:711–718, 2018.

[49] Karl J. O'Dwyer and David Malone. Bitcoin mining and its energy footprint. *25th IET Irish Signals & Systems Conference 2014 and 2014 China-Ireland International Conference on Information and Communications Technologies (ISSC 2014/CIICT 2014). IET*, pages 280–285, 2014.

[50] Eric Masanet, Arman Shehabi, Nuoa Lei, Sarah Smith, and Jonathan Koomey. Bitcoin's growing energy problem: Results from a spatial model of mining operations. *Joule*, 5:1033–1047, 2021.

[51] Richard P. Feynman, Robert B. Leighton, and Matthew Sands. *The Feynman Lectures on Physics, Vol. 3: Quantum Mechanics.* Addison-Wesley, 1965.

[52] Albert Einstein, Boris Podolsky, and Nathan Rosen. Can quantum-mechanical description of physical reality be considered complete? *Physical Review*, 47(10):777–780, 1935.

[53] R. Shankar. *Principles of Quantum Mechanics.* Springer, New York, 2nd edition, 1994.

[54] Werner Heisenberg. The actual content of quantum theoretical kinematics and mechanics. In John A. Wheeler and Wojciech H. Zurek, editors, *Quantum Theory and Measurement*, pages 62–84. Princeton University Press, Princeton, NJ, 1983. Originally published in 1927 as "Über den anschaulichen Inhalt der quantentheoretischen Kinematik und Mechanik".

[55] Michael A. Nielsen and Isaac L. Chuang. *Quantum Computation and Quantum Information.* Cambridge University Press, Cambridge, 2010.

[56] Erwin Schrödinger. The present situation in quantum mechanics. In John A.

Wheeler and Wojciech H. Zurek, editors, *Quantum Theory and Measurement*, pages 152–167. Princeton University Press, Princeton, NJ, 1983. Originally published in 1935 as ”Die gegenwärtige Situation in der Quantenmechanik”.

[57] Charles H. Bennett and David P. DiVincenzo. Quantum information and computation. *Nature*, 404(6775):247–255, 2000.

[58] Jay Gambetta. IBM’s Roadmap for Scaling Quantum Technology. Accessed: Apr. 15, 2021.

[59] Gilles Brassard, Peter Høyer, Michele Mosca, and Alain Tapp. Quantum amplitude amplification and estimation. In *Quantum Computation and Information*, pages 53–74. American Mathematical Society, 2002.

[60] Lov K Grover. Quantum mechanics helps in searching for a needle in a haystack. *Physical review letters*, 79(2):325, 1997.

[61] IBM. Ibm unveils 400 qubit-plus quantum processor and next-generation ibm quantum system two. *IBM Newsroom*, 2022. Accessed: 2024-05-16.

[62] IBM. Ibm quantum computer demonstrates next step towards moving beyond classical supercomputing. *IBM Newsroom*, 2023. Accessed: 2024-05-16.

[63] IBM Quantum. IBM Quantum Experience, 2024. Accessed: 2024-05-16.

[64] Frank Arute, Kunal Arya, Ryan Babbush, Dave Bacon, Joseph C Bardin, Rami Barends, Rupak Biswas, Sergio Boixo, Fernando GSL Brandao, David A Buell, et al. Quantum supremacy using a programmable superconducting processor. *Nature*, 574(7779):505–510, 2019.

[65] Microsoft. Advancing science: Microsoft and quantinuum demonstrate the most reliable logical qubits on record with an error rate 800x better than physical qubits. *The Official Microsoft Blog*, 2024. Accessed: 2024-05-16.

[66] Microsoft. Quantum computing - microsoft research, 2024. Accessed: 2024-05-16.

[67] Microsoft. Microsoft's quantum machine - azure quantum, 2024. Accessed: 2024-05-16.

[68] John Preskill. Quantum computing in the nisq era and beyond. *Quantum*, 2:79, 2018.

[69] Honeywell Quantum Solutions. Honeywell sets another record for quantum computing performance, 2024. Accessed: 2024-05-16.

[70] M. P. da Silva, C. Ryan-Anderson, J. M. Bello-Rivas, A. Chernoguzov, J. M. Dreiling, C. Foltz, J. P. Gaebler, T. M. Gatterman, D. Hayes, N. Hewitt, et al. Demonstration of logical qubits and repeated error correction with better-than-physical error rates. *Microsoft Azure Quantum and Quantinuum*, April 2024.

[71] Simon J. Evered, Dolev Bluvstein, Marcin Kalinowski, Sepehr Ebadi, Tom Manovitz, Hengyun Zhou, Sophie H. Li, Alexandra A. Geim, Tout T. Wang, Nishad Maskara, Harry Levine, Giulia Semeghini, Markus Greiner, Vladan Vuletić, and Mikhail D. Lukin. High-fidelity parallel entangling gates on a neutral-atom quantum computer. *Nature*, 615:682–687, 2023.

[72] Google Quantum AI Team. Our progress toward quantum error correction. *Google Blog*, 2023. Available at `https://blog.google/technology/ai/our-progress-toward-quantum-error-correction`.

[73] Anne J. Manning. Harvard researchers create first logical quantum processor. *Harvard Gazette*, 2023. Available at `https://news.harvard.edu/gazette/story/2023/12/harvard-researchers-create-first-logical-quantum-processor`.

[74] Yudong Cao, Jonathan Romero, Jonathan P. Olson, Matthias Degroote, Peter D. Johnson, Maria Kieferová, ..., and Alán Aspuru-Guzik. Quantum chemistry in the age of quantum computing. *Chemical Reviews*, 119(19):10856–10915, 2019.

[75] Ryan Babbush, Jarrod McClean, Dave Wecker, Alán Aspuru-Guzik, and Nathan Wiebe. Chemical basis of trotter-suzuki errors in quantum chemistry simulation. *Physical Review A*, 94(2):022311, 2016.

[76] Crispin H. V. Cooper. Exploring potential applications of quantum computing in transportation modelling. *IEEE Transactions on Intelligent Transportation Systems*, 23(9):14712–14720, 2022.

[77] Christopher D. B. Bentley, Samuel Marsh, André R. R. Carvalho, Philip Kilby, and Michael J. Biercuk. Quantum computing for transport optimization. *arXiv preprint arXiv:2206.07313*, 2022.

[78] Sean J. Weinberg, Fabio Sanches, Takanori Ide, Kazumitzu Kamiya, and Randall Correll. Supply chain logistics with quantum and classical annealing algorithms. *Scientific Reports*, 13:4770, 2023.

[79] Dylan Herman, Cody Googin, Xiaoyuan Liu, Yue Sun, Alexey Galda, Ilya Safro, Marco Pistoia, and Yuri Alexeev. Quantum computing for finance. *arXiv preprint arXiv:2307.11230*, 2023.

[80] Daniel J. Egger, Claudio Gambella, Scott McFaddin, Elena Yndurain, Andrea Simonetto, Stefan Woerner, and Richard Garcia. Quantum computing for finance: State-of-the-art and future prospects. *IEEE Transactions on Quantum Engineering*, 2020.

[81] Adam Bouland, Bill Fefferman, Umesh Vazirani, and Thomas Vasudevan. Portfolio optimization and beyond: Quantum computing for finance. *arXiv preprint arXiv:2107.13418*, 2021.

[82] B. Maurice Benson and R. Keinan. Harnessing the power of quantum computing for drug discovery. *Nature*, 2023.

[83] Chad Edwards and Lucas Siow. The future of drug development with quantum computing. *McKinsey*, 2023.

[84] Innovate UK. Unlocking the potential of quantum: £45 million investment to drive breakthroughs in brain scanners, navigation systems, and quantum computing. 2023.

[85] Various. U.s. government increases investment in quantum computing. 2023.

[86] Inside Quantum Technology. France invests €1.8 billion in quantum technologies. 2023.

[87] Tech Monitor. German government makes €67m quantum computing investment. 2023.

[88] Government of Canada. Government of canada launches national quantum strategy to create jobs and advance quantum technologies. 2023.

[89] Inside Quantum Technology. Japan's quantum computing investments and initiatives. 2023.

[90] Matthew Campagna and Lily Chen. Quantum-safe cryptography and the bitcoin ledger. *National Institute of Standards and Technology*, 9:1–12, 2015.

[91] Marco Tomamichel and Christian Schaffner. Blockchain meets quantum computing. *Quantum Science and Technology*, 3(3):034003, 2018.

[92] Aleksey K Fedorov, Evgeniy O Kiktenko, and Alexei I Lvovsky. Quantum computers put blockchain security at risk. *Nature*, 563(7732):465–467, 2018.

[93] Shigeya Matsuo. Quantum attacks on blockchain cryptography and the need for quantum-safe security. *NTT Technical Review*, 17(4):1–5, 2019.

[94] Hassan Nouri, Ajith Sayakkara, and Mohammad Iqbal Lubis. Blockchain technology and the prospect of quantum attack. In *2019 19th International Conference on Sciences and Techniques of Automatic Control and Computer Engineering (STA)*, pages 27–32. IEEE, 2019.

[95] Yi Yu, Dacheng Qiu, and Liang Qiu. Quantum-resistant blockchain based on quantum key distribution. *International Journal of Quantum Information*, 17(01):1950011, 2019.

[96] Kaito Ikeda, Kazumasa Omote, and Atsuko Miyaji. Quantum blockchain: A decentralized, encrypted and distributed database based on quantum mechanics. *Future Internet*, 10(11):105, 2018.

[97] Marcos Allende, Diego López León, Sergio Cerón, Adrián Pareja, Erick Pacheco, Antonio Leal, Marcelo Da Silva, Alejandro Pardo, Duncan Jones, David J Worrall, et al. Quantum-resistance in blockchain networks. *Scientific Reports*, 13(1):5664, 2023.

[98] QRL Team. The quantum resistant ledger (qrl), 2020. Available at `https://theqrl.org`.

[99] David McGrew, David A. Mcgrew, Michael Curcio, and Scott Fluhrer. Xmss - a practical forward secure signature scheme based on minimal security assumptions. In *2011 IEEE Symposium on Security and Privacy*, pages 44–59. IEEE, 2011.

[100] National Institute of Standards and Technology (NIST). Post-quantum cryptography standardization. Technical report, National Institute of Standards and Technology, 2022. Available at `https://csrc.nist.gov/projects/post-quantum-cryptography/post-quantum-cryptography-standardization`.

[101] Vitalik Buterin. How to hard fork to save most users' funds in a quantum emergency, 2024. Available at https://ethresear.ch/t/how-to-hard-fork-to-save-most-users-funds-in-a-quantum-emergency/18901.

[102] *Cryptocurrency Prices, Charts and Market Capitalizations.* Accessed: Apr. 15, 2021.

[103] Bitcoin developers. Transactions - bitcoin developer documentation. https://developer.bitcoin.org/reference/transactions.html, 2021. Accessed: 2023-04-13.

[104] Andreas M. Antonopoulos. *Mastering Bitcoin: Unlocking Digital Cryptocurrencies.* O'Reilly Media, Inc., Sebastopol, CA, 2014.

[105] Adam Back. Hashcash - a denial of service counter-measure. Technical report, 2002.

[106] Vitalik Buterin et al. A next-generation smart contract and decentralized application platform. *white paper*, 3(37):2–1, 2014.

[107] Gavin Wood et al. Ethereum: A secure decentralised generalised transaction ledger. *Ethereum project yellow paper*, 151(2014):1–32, 2014.

[108] Ethereum Foundation. Ethash: Ethereum proof of work. https://github.com/ethereum/wiki/wiki/Ethash, 2015.

[109] Vitalik Buterin and Danny Ryan. Ethereum 2.0 specifications. Ethereum Foundation, 2020.

[110] Ethereum Foundation. Randao: A dao working as rng of ethereum. Ethereum Foundation.

[111] Solidity Developers. *Solidity —Contract-Oriented Smart-Contract Language*. Ethereum Foundation, 2024. Version 0.8.25, accessed 8 June 2025.

[112] Ralph C. Merkle. A digital signature based on a conventional encryption function. In *Advances in Cryptology - CRYPTO '87, A Conference on the Theory and Applications of Cryptographic Techniques*, pages 369–378, 1987.

[113] Gav Would. Modified merkle patricia tree. 2015.

[114] Charlie Lee. Litecoin. Cryptocurrency, 2011.

[115] Colin Percival. Stronger key derivation via sequential memory-hard functions. In *Proceedings of the 2009 BSDCan Conference*, 2009.

[116] SerHack and Monero Community. *Mastering Monero: The future of private transactions*. CreateSpace Independent Publishing Platform, 2018.

[117] Nicolas van Saberhagen. Cryptonote v 2.0, 2013.

[118] Masarah Paquet-Clouston, Petra Kijewski, and Rainer Decker. Ransomware payments in the bitcoin ecosystem. *The Journal of Cybersecurity*, 4(1), 2018.

[119] Tevador and Monero Community. Randomx: A proof-of-work algorithm for general purpose cpus. `https://github.com/tevador/RandomX`, 2019. Accessed: 2023-04-14.

[120] Samuel Neves, Zooko Wilcox-O'Hearn, Christian Winnerlein, and Jean-Philippe Aumasson. Blake2: simpler, smaller, fast as md5. `https://blake2.net/`, 2013.

[121] Shen Noether, Sarang Noether, and Adam Mackenzie. Ring signature confidential transactions for monero, 2015.

[122] Ronald L Rivest, Adi Shamir, and Leonard Adleman. A method for obtaining digital signatures and public-key cryptosystems. *Communications of the ACM*, 21(2):120–126, 1978.

[123] Daniel J. Bernstein, Peter Birkner, Marc Joye, Tanja Lange, and Christiane Peters. Twisted edwards curves. Cryptology ePrint Archive, Paper 2008/013, 2008. `https://eprint.iacr.org/2008/013`.

[124] Daniel J Bernstein, Niels Duif, Tanja Lange, Peter Schwabe, and Bo-Yin Yang. High-speed high-security signatures. *Journal of Cryptographic Engineering*, 2(2):77–89, 2012.

[125] Benedikt Bünz, Jonathan Bootle, Dan Boneh, Andrew Poelstra, Pieter Wuille, and Greg Maxwell. Bulletproofs: Short proofs for confidential transactions and more. *2018 IEEE Symposium on Security and Privacy (SP)*, pages 315–334, 2018.

[126] Torben Pryds Pedersen. Non-interactive and information-theoretic secure verifiable secret sharing. In *Annual International Cryptology Conference*, pages 129–140. Springer, 1991.

[127] Eli Ben-Sasson, Alessandro Chiesa, Christina Garman, Matthew Green, Ian Miers, Eran Tromer, and Madars Virza. Zerocash: Decentralized anonymous payments from bitcoin. 2014. Available at `http://zerocash-project.org/media/pdf/zerocash-extended-20140518.pdf`.

[128] Gaurav Roy. Monero, dash, and zcash - will privacy coins survive the oncoming regulatory onslaught?, 2023. Accessed: 2023-09-02.

[129] Steve Kaaru. Huobi delists monero, zcash and other privacy coins amid regulatory pressure, September 2022. Accessed: 2023-08-12.

[130] Eli Ben-Sasson, Alessandro Chiesa, Daniel Genkin, Eran Tromer, and Madars Virza. Succinct non-interactive zero knowledge for a von neumann architecture. 2013. Available at `https://eprint.iacr.org/2013/879.pdf`.

[131] Alex Biryukov and Dmitry Khovratovich. Equihash: Asymmetric proof-of-work based on the generalized birthday problem. In *NDSS Workshop on Blockchain, Cryptocurrencies and Contracts*. The Internet Society, 2016. Available at `https://www.researchgate.net/publication/316971847_Equihash_Asymmetric_Proof-of-Work_Based_on_the_Generalized_Birthday_Problem`.

[132] Leif Both and Alexander May. The approximate k-list problem. *IACR Transactions on Symmetric Cryptology*, 2017(1):380–397, 2017. Accessed: 2024-04-30.

[133] Grin Community. Grin, 2023. Accessed: [13/08/2023].

[134] Tom Elvis Jedusor. Mimblewimble. Online text file, 2016.

[135] Privacy coin grin is victim of 51% attack, 11 2020. Accessed: 2023-06-18.

[136] John Tromp. Cuckoo cycle: A memory bound graph-theoretic proof-of-work. In *International Conference on Financial Cryptography and Data Security*, pages 49–62. Springer, 2015.

[137] Menghan Zheng, Shan Wang, Yushun Fan, Tao Yue, and Shaukat Ali. A systematic literature review on blockchain governance, 2021. Accessed: 2024-03-02.

[138] CryptoSlate. Ethereum tokens, 2024. Accessed: 2024-04-09.

[139] Tether Operations Limited. Tether: Fiat currencies on the bitcoin blockchain, 2018. Accessed: 2023-12-09.

[140] Steve Ellis, Ari Juels, and Sergey Nazarov. Chainlink: A decentralized oracle network, 2017. Accessed: 2024-02-09.

[141] Ari Juels, Lorenz Breidenbach, Andrew Miller, and Sergey Nazarov. Chainlink 2.0: Next steps in the evolution of decentralized oracle networks, 2021. Accessed: 2024-03-12.

[142] Lorenzo Grassi, Andrey Bogdanov, Thomas Kranz, and Gregor Leander. Quantum algorithms for the k-xor problem. *Designs, Codes and Cryptography*, 86(1):1–26, 2018.

[143] CoinWarz. Bitcoincash hashrate chart - bch hashrate, 2024. Accessed: 2024-04-10.

[144] Leslie Lamport, Robert Shostak, and Marshall Pease. The byzantine generals problem. *ACM Transactions on Programming Languages and Systems*, 4(3):382–401, 1982.

[145] Gordon E Moore. Cramming more components onto integrated circuits, 1965.

[146] *Hash-Rate in Bitcoin*. Accessed: Apr. 15, 2021.

[147] Artur Meynkhard. Fair market value of bitcoin: Halving effect. *Invest. Manag. Financ. Innov*, 16:72–85, 2019.

[148] *Controlled supply*. Accessed: Apr. 15, 2021.

[149] Davide Castelvecchi. Ibm's quantum cloud computer goes commercial. *Nature News*, 543(7644):159, 2017.

[150] Simon J Devitt. Performing quantum computing experiments in the cloud. *Physical Review A*, 94(3):032329, 2016.

[151] L. Ismail and H. Materwala. Evolution of blockchain consensus algorithms: a review on the latest milestones of blockchain consensus algorithms. *Cybersecurity*, 3(1):1–19, 2020.

[152] Comparisons, 2022.

[153] Sean Steinsmith. Crypto power usage is helping to spur renewable energy investments, June 2022.

[154] Bitmain. Antminer s19j pro (100 th/s), 2023. Accessed: 2023-06-05.

[155] Rolf Landauer. Irreversibility and heat generation in the computing process. *IBM Journal of Research and Development*, 5(3):183–191, 1961.

[156] Peter J. Mohr, David B. Newell, and Barry N. Taylor. *CODATA Recommended Values of the Fundamental Physical Constants: 2018*. National Institute of Standards and Technology, Gaithersburg, MD, 2018.

[157] Hugo Paquet, Jean-Pierre David, and Claude Tanguay. Foundations of reversible computation. In Robert Wille and Rolf Drechsler, editors, *Reversible Computation*, volume 11497 of *Lecture Notes in Computer Science*, pages 3–25. Springer International Publishing, 2019.

[158] Daniel Eric Gottesman. *Stabilizer Codes and Quantum Error Correction*. Ph.d. thesis, California Institute of Technology, Pasadena, California, 1997.

[159] Peter W. Shor. Scheme for reducing decoherence in quantum computer memory. *Physical Review A*, 52(4):R2493–R2496, 1995.

[160] Mooseop Kim, Jaecheol Ryou, and Sungik Jun. Efficient hardware architecture of sha-256 algorithm for trusted mobile computing. In *International Conference on Information Security and Cryptology*, pages 240–252. Springer, 2008.

[161] Akito Noiri, Kenta Takeda, Takashi Nakajima, Takashi Kobayashi, Amir Sammak, Giordano Scappucci, and Seigo Tarucha. Fast universal quantum gate above the fault-tolerance threshold in silicon. *Nature*, 601(7893):338–342, 2022.

[162] Colin D Bruzewicz, John Chiaverini, Robert McConnell, and Jeremy M Sage. Trapped-ion quantum computing: Progress and challenges. *Applied Physics Reviews*, 6(2), 2019.

[163] International Energy Agency. Sweden: Countries: IEA, 2023. Accessed: 2023-06-05.

[164] Elizabeth Napolitano. U.s. bill proposes outlawing government use of china-made blockchains and tether's usdt, 2023. Accessed: 2023-07-09.

[165] Chambers and Partners. Global legal insights - blockchain cryptocurrency laws and regulations 2023, 2023. Accessed: 2023-07-09.

[166] Global Legal Insights. Blockchain cryptocurrency laws and regulations, 2023. Accessed: 2023-07-09.