Sovereign Sensors:

An investigation of factors pertaining to the governance of informational resources on the decentralized web

Masters of Science Dissertation

Spatial Data Science and Visualisation

UCL Bartlett Centre for Advanced Spatial Analysis

2018 - 2019 Academic Year

John R Hoopes IV

“But unless we recognize the overall failures of our current systems we most probably don’t stand a chance.”

Greta Thunberg

No One Is Too Small To Make A Difference

2019

“... to create a world that better preserves the autonomy of the individual ...”

Vitalik Buterin

2016

## 

## Statement of Ethics

## Statement of Originality

## 

## Acknowledgements

## Introduction

Objects exist in space. These entities - agents - interact over time. We know this because they come into our awarenesses, moment to moment. We[[1]](#footnote-0) are objects that perceive the world, observing it through these moments while we are alive.

Base Assumptions:

Communication is necessary for coordination amongst individuals in a collective.

Knowledge is a form of power ([Foucault](https://en.wikipedia.org/wiki/Power-knowledge)).

Knowledge is derived from the interpretation of information (Floridi).

Information is communicated through transmission of data upon a channel.

How data is communicated has bearing on the power dynamics of the communicators, and the justness of collective coordination.

This dissertation is a reporting back of the current state of my understanding of the topics investigated. My overall interests include the patterns and relationships of our lived experience, including the natures of both matter and meaning, and the relationship between the two. My angle is of the simplest instance of which I am aware of informational entities communicating: connected sensors - computing nodes interpreting, transmitting and receiving binary data. My goal is to investigate the intricacies of these systems of connected sensors in the current moment - 2019.

The relevance is difficult for me to overstate - I am drawn to pursue this research because I feel it might help us to understand the enormous risks and opportunities inherent in the technologies we are just beginning to discover. My hope is that if we can understand these risks we might be able to mitigate them, just as if we do the same for the opportunities they might be maximized toward the promotion of the values we share. I believe that this is a pivotal moment[footnote]Perhaps every present moment is a pivotal one?[/footnote] for us, here on Earth today. It seems we have the potential - the capability, and perhaps the capacity - to finally achieve what so many just people have spent their life in pursuit of: the eradication of violence and the peaceful coordinated thriving of life on the planet. My hope is that this research contributes to bringing about that state[[2]](#footnote-1).

The risks of this endeavor cannot be understated. What I intend to learn will imbue me with great power, the ability to create the services that people and machines may use to live in the 21st century. I’d like to say I understand this, but I cannot - I am only beginning to grasp the ethical implications of this emerging reality. For this reason the work will describe a number of ethical principles, adopted or adapted or arrived at in my studies.

Specifically, I will look at the interaction of networks of connected sensors with blockchain networks. In order to adequately define terms and frame the research, the necessary technologies will be reviewed and clear explanations provided. To better understand their behavior, a middle-range agent based model simulating key features of edge sensors, quasi-Turing-complete smart contract platforms and their interaction was built. Preliminary experiments produced simulation output of parameter sweeps of three independent variables; three dependent variables were measured and analyzed.

In the Discussion I briefly review the model development and results, and raise some of the questions I encountered that were not satisfactorily answered in the current scientific paradigm, as I understand it.

I engage in this work with an ethical mindset, and hope that this writing conveys that I strive to consider deeply not only the technical, but also the ethical, moral, social, economic and political implications of these subjects. I propose the outlines of a few configurations I could imagine might be feasible that utilize the potential of these systems in a just and inclusive way. A distinction needs to be made, between people that are alive and everything else. It seems important that we orient ourselves toward the former. If each one is equal[footnote]In value, yes: 1. In qualities? Never.[/footnote], this must be ingrained in us and in everything we create.

Much as cooling water reaches a point at which its molecules become ordered and aligned, a similar phenomenon might occur as we approach saturation of information transfer between sentient entities. If understanding improves, entities are less likely to act in a way that would disrupt another's intentions; given that agents can adapt their intentions, deconfliction of behavior may be possible.

### Use Cases:

(Examples? Shipping and logistics, autonomous vehicles, surveillance and security)

### Outline

Literature review and enabling technologies

Methodology, ODD protocol (Grimm)

Brief discussion of model results, then an exploration of technical, as well as economic, social, political and ethical factors related to the research.

Conclusion - future research.

## Research question and hypotheses

### General

What is information?

How is information governed?

What if the noosphere is real, and behaves according to rules we can understand?

### Specific

What if trusted sensors could reliably used as oracles for smart contracts? What opportunities would this create? What constraints and limitations?

### Agent-based model

What are the effects of network size and data transfer dynamics on blockchain validation patterns and costs?

## 1 Literature Review

### 1.1 Key Terms

Figure 1: Key Terms

Objects / Agents / Entities

Anything that exists.

They have qualities and an effect on other objects they interact with (encounter)

Physical objects

Inert objects

Inactive objects

Active objects

Is active having agency?

Living objects ----------- Synthetic objects with agency

Sentient objects - perceiving information?

Conscious objects - aware of one’s existence - perceives information

Informational objects - possessing an awareness, perceives / conceives informations

Synthetic objects with agency

Anything that can move or emit energy

Computers

Inputs, CPU, output destinations

CPU performs binary operations on binary data

Table of binary operations

And, nand, or, xor, etc

Data is how the qualities of an object are projected into the world.

Data carries information about its origin on a carrier wave.

Trusted sensors

Oracle

Entity

This research will delve into the ways computers[[3]](#footnote-2) communicate, and investigate the possibility of a new paradigm for solving some of the challenges intrinsic to the governance of informational resources.

### Computers

Humans have relied on physical objects - tools - to help perform logical operations since the origin of writing and mathematics (Huang 2017). A common feature of these tools (tally sticks, abacuses, and so on) is their ability to store symbolic representations of numerical values so the operator did not have to remember the values. With the aid of these tools, human understanding of the patterns of mathematics developed.

Building on the work of Gottfried Wilhelm Leibniz’s work on a mechanical calculator of “the four fundamental operations of arithmetic”[[4]](#footnote-3) (Martin 1992 pp 39), Charles Babbage first proposed an automatic computer in 1822, later outlining an even more sophisticated analytical engine, a “general purpose computer” (CrashCourse 2017). Based on Babbage’s proposed mechanical computers, Ada Lovelace foresaw the potential of computing[[5]](#footnote-4), going so far as to develop software programs for Babbage’s still-unbuilt machine (Lovelace 1843, Fuegi 2003, Essinger 2018). Technology was moving toward the reliable mechanization of logical operations.

In the early 20th century, mathematicians, researchers and engineers worked to incorporate the use of digital electronics in computing systems, based on advancements of understandings in the properties of electricity and conductive materials, as well as Akira Nakashima’s early work on switching theory, based on two-valued Boolean algebra (Wynn-Williams 1931, Stankovic 2008). That same decade Babbage’s vision of a general purpose analytical engine was finally realized by Alan Turing in 1936, in his (now known as) Turing complete “a-machine” (Turing 1937), a mathematical model of computation capable of “simulat[ing] any computer algorithm (Mullins 2012).

\*\*\*Data types?. UTF-8 example table. UTF-8 full table in appendix. Binary operators\*\*\*

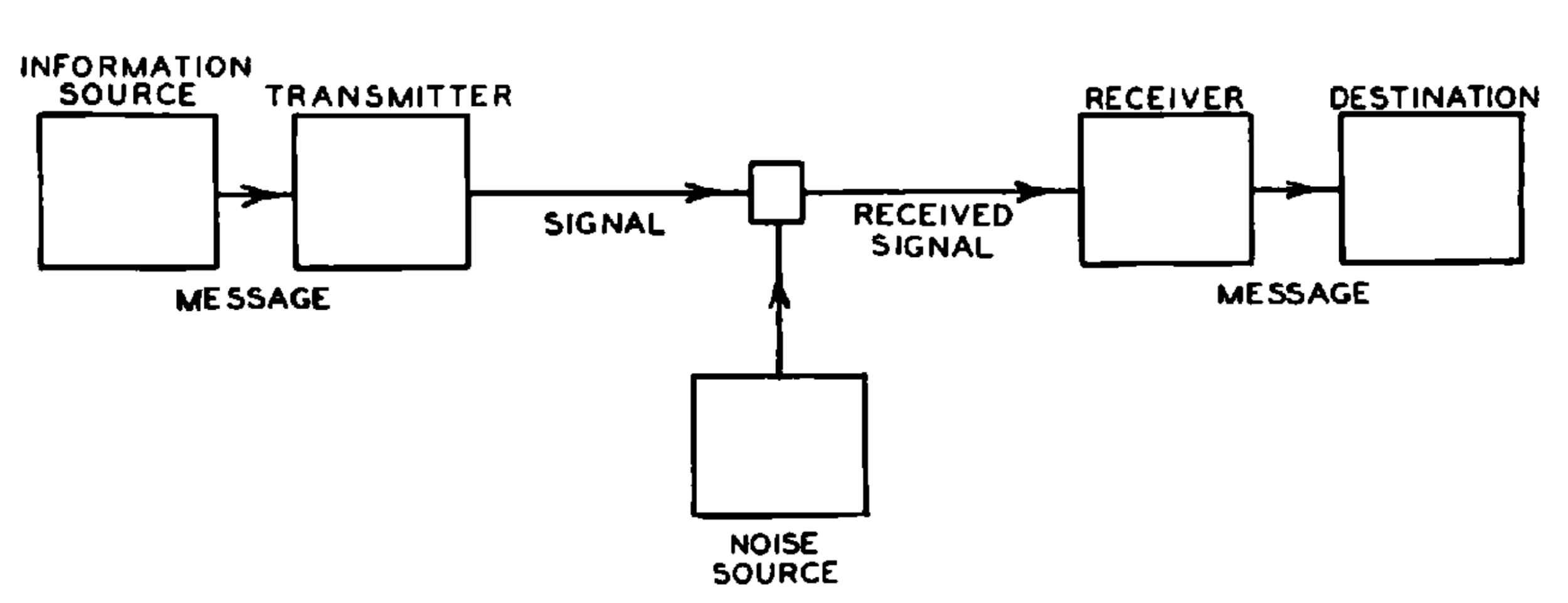
### Communication

As innovations in computing machines were progressing, so were techniques for transmitting information over distances. Signals had been sent on optical and auditory channels for millennia (Gleick 2011); by the early 1800s the semaphore telegraph, by which messages are transmitted via line-of-sight relay stations on a visual channel, was in wide use (Burns 2004). The encoding of symbols into electrical currents had been discussed for some time (Fahie 1884) prior to “the first working electrostatic telegraph” being built in 1816 (Norman 2019). Such a system conferred substantial benefits over the optical telegraphs in use at the time[[6]](#footnote-5).

These telegraphy systems shared the attribute of providing a communication channel upon which messages could travel rapidly and accurately. They necessarily relied on the establishment of some system of encoding information on the channel - without the ability to interpret the symbols encoded, the data would be meaningless to the informee.

This emerging understanding of the behavior of electrical circuits and its implications for data processing, as well as the encoding of information on a serial channel[[7]](#footnote-6), culminated in the publication of Claude Shannon’s master’s thesis[[8]](#footnote-7), *A symbolic analysis of relay and switching circuits* (1938). A decade later, in *A mathematical theory of communication*, Shannon (1948) defined the fundamental principles of the transfer of a message from source (informer) to recipient (informee): “the theory that lies behind any phenomenon involving data encoding and transmission” (Floridi 2010)[[9]](#footnote-8).

Figure 2: Schematic diagram of a general communication system (Shannon 1948)



Shannon’s paper described how a binary message is encoded on a signal[[10]](#footnote-9) by the information source, or informer. The signal is transmitted on a (perhaps noisy) channel, then detected by the receiver and interpreted by the information destination, the informee. Digital information is encoded into a signal by changing some quality of the carrier wave - modulating the amplitude, frequency or phase of fixed-duration pulses of electrical current (Nair 2002 pp 289) or increments of light (Dume 2012, Economist 2017). The information receiver, able to detect the regular modulations of the signal, can decode the binary sequence from the signal upon reception.

Figure x: Digital information encoded on a carrier wave

Shannon’s work provided the necessary framework for the encoding and transmission of binary information on a channel; computers executing instructions encoded in binary were developing in sophistication and usage. From these advancements, networked computing arose.

#### Cryptography

“A message is a discrete unit of communication intended by the source for consumption by some recipient or group of recipients” (Wikipedia 2019a). To maintain privacy on an open network, message senders need to be assured that:

* Only intended recipients can access the contained information - they are “authorized”;
* Intermediary relayers cannot access the contained information - the message is “confidential”;
* The message received is identical to the message sent - it has “integrity”.

In response to these three needs - confidentiality, authorization, and integrity - cryptographic algorithms and protocols have been developed, enabling the establishment of secure channels of communication on an open[[11]](#footnote-10) network. A basic familiarity with their purpose and function is important to understanding subsequent concepts. Note that the algorithms described are applied to binary sequences (messages).

Table x: Cryptography Key Terms

|  |  |
| --- | --- |
| **cipher** | “a secret or disguised way of writing; a code” (Oxford 2019a) |
| **plaintext** | an unencrypted message, in which the information contained is visible |
| **ciphertext** | an encrypted plaintext, in which the information contained is obscured |
| **algorithm** | “a process or set of rules to be followed in calculations or other problem-solving operations” (Oxford 2019b) |
| **key** | “a sequence of bits” (Techopedia 2019) |
| **protocol** | “a set of agreed-upon conventions” (Cerf 1974) |

##### Integrity

###### Hashing Algorithms

Cryptographic hashing algorithms accept a sequence of bits or arbitrary length. From this message, the algorithm calculates a cryptographic hash (or digest), which is a fixed-length sequence calculated from the input parameter. Most often this hash is substantially shorter than the input data[[12]](#footnote-11) - for example, the SHA-256 algorithm returns a 256-bit (32-byte) hash for an input of any length (movable-type 2019).

To be useful, cryptographic hashing algorithms must have a few crucial features. They must be deterministic: the same input sequence will always yield the same output hash. Extending this, changing even a single bit in the input message must result in a different output hash[[13]](#footnote-12). Also, they must be one-way, or trapdoor, functions: it must be extremely difficult to calculate the input data from the message digest[[14]](#footnote-13) (Ker 2014).

Cryptographic hashes serve as a kind of message “fingerprint”: they are effectively[[15]](#footnote-14) unique to the message. This means that, if a message digest is transmitted along with the message, the recipient can confirm that the message’s integrity is uncompromised by hashing the message on their computer and comparing the output hash with the one included in the message[[16]](#footnote-15).

##### Confidentiality and Authentication

Key ciphers enable message senders to obscure the information contained within a message in such a way that it is inaccessible to any recipient without the right key, enabling the message to be decrypted and contained information revealed[[17]](#footnote-16).

###### Symmetric Key Ciphers

A symmetric key cipher allows communicators to maintain confidentiality even if a message is viewed by an unauthorized entity. The message plaintext is encrypted with a secret key by the informer in a private, secure computing environment prior to transmission on the open network. In this process, an arbitrary-length binary plaintext and a secret key are passed into an algorithm, which returns the encrypted message - the ciphertext. To decrypt, the ciphertext and the same key are passed into an algorithm, which returns the plaintext, thereby unobscuring the information represented[[18]](#footnote-17) (Ker 2014).

Figure x: XOR bitwise operation

###### Asymmetric Key Ciphers

While symmetric key ciphers provide communicators confidence in message confidentiality, they are constrained by the need for both parties to have the same secret key. It is infeasible for every pair establishing a secure communication link on a public channel to meet and exchange keys; transmitting a plaintext key on an open network means unauthorized intermediaries could, if aware of the algorithm used[[19]](#footnote-18), decrypt every subsequent ciphertext encrypted with that key.

To resolve this, asymmetric key ciphers and key exchange protocols were developed, based on the work of Ellis, Cocks, Diffie and Hellman, and Rivest, Shamir and Adleman, and Merkle (Ker 2014). The algorithms and protocols developed and defined enabled two parties “to communicate confidentially after transmitting a key which is not confidential” (Ker 2014 pp 71).

The distinguishing feature of asymmetric key ciphers is the creation of two keys: a public and a private key. These numbers are linked based on the mathematical relationship between the two.[[20]](#footnote-19)

Two pairs of algorithms characterize asymmetric key ciphers, based on these public-private keypairs: encrypt-decrypt and sign-verify.

###### Encrypt - Decrypt

Asymmetric key ciphers include algorithms by which messages can be encrypted and decrypted, as in symmetric key ciphers. However, a critical difference exists: encryption is performed by passing the plaintext and the message recipient’s public key into the encryption algorithm. This is a trapdoor function, in that it is “easy to perform and difficult to invert” (Ker 2014). Decryption of the resultant ciphertext is performed by passing it, along with the recipient’s private key, into the decryption algorithm, which returns the plaintext.

```python

```

By disaggregating these two functions, the creators of public key cryptosystems created a way to establish a secure channel on a public network: communicators could simply transmit their public keys to each other in plaintext form, while maintaining the secrecy of their private key. The message sender could then encrypt the message with the recipient’s public key, confident that it could only be decrypted with the linked private key - which only the intended informee had[[21]](#footnote-20).

###### Sign - Verify

Due to the mathematical properties of these public-private keypairs, a second pair of algorithms was possible in the cipher: sign and verify. This pair of algorithms provides message recipients confidence that the message was sent by the holder of the private key. If private keys are properly managed[[22]](#footnote-21), this allows informational entities to prove their identity (the basis of authentication) by proving they possess the private key without ever sharing that key.

The sign algorithm accepts a signer’s private key and a message to be signed; the function returns a signed message.

|  |
| --- |
| sign(data, private\_key) >> returns signed\_data |

The verify algorithm, when provided the signed message and the signer’s public key, enables the message recipient to confirm that the message was signed with the signer’s private key (Ker 2014).

verify(signed\_data, public\_key)

>> returns True if signed\_data was signed with public\_key, False if not[[23]](#footnote-22)

This sign-verify functionality of public key cryptosystems is, in this author’s view, the enabling technology behind the decentralized web, and represents a profound opportunity to improve the systemic justice of the Internet.

##### Cryptographic Protocols

By combining the functionalities associated with cryptographic hashing algorithms and symmetric and asymmetric key ciphers, standardized procedures (i.e. protocols) have been established enabling communicators to verify sender authenticity, ensure message confidentiality[[24]](#footnote-23), and confirm message integrity[[25]](#footnote-24), using technical mechanisms (Ker 2014).

### Networked computers

As general purpose analytical machines, Turing-complete computers processing binary data began to be adopted by the military, industry and academia (CrashCourse 2017b) to efficiently and accurately compute and catalogue (relatively) large volumes of data. The transfer of binary data from one computer to another would enable informational resources to be shared amongst them, thereby enabling the execution of more complex applications, as well as the access of data stored on other computers.

This vision was put forward with striking clarity in the late 1950s and early 1960s by computer scientist J.C.R. Licklider (livinginternet.com 2019): a “galactic network” of interconnected computers (Leiner 1997). Licklider and his contemporaries investigated the concept, including the development of a theory of packet switching, which enabled a channel to be used by multiple traffic sources (Kleinrock 1961). These developments led to the first wide-area network connection being established over a telephone line connected California and Massachusetts in 1965 by Merrill and Roberts (Leiner 1997).

In 1966, the Defense Advanced Research Projects Agency began funding the development of ARPANET, which accelerated the refinement of “the overall structure and specifications” for such a network of computers (Leiner 1997). By the early 1970s, the technical capability for networked computing was established.

However, many of the networks relied on protocols implemented within organizations: because “these protocols have addressed only the problem of communication on the same network” (Cerf 1974), internetworking remained difficult. In *A Protocol for Packet Network Intercommunication*, Cerf and Kahn (1974) established the Transport Communication Protocol, which “provides reliable, ordered, and error-checked delivery of a stream of octets (bytes) between applications running on hosts communicating via an [Internet Protocol] network” (Wikipedia 2019e).

These protocols established a standard format for data packets[[26]](#footnote-25) to be transmitted to a destination address, which included in the message. Critically, Kahn adhered to principles in his design of the protocols, including that “There would be no global control at the operations level” (Leiner 1997). Rather than relying on a centralized authority to organize traffic - which Kahn likely understood to be unscalable - the authority would reside in the rules of the protocol, to which system developers would adhere. This alternative form of authority - in a conceptual framework rather than a decision-maker - will resurface shortly.

#### The Web (1.0)

With the establishment of the Transport Control Protocol and the Internet protocol suite, local area networks using heterogeneous operating systems had a standardized way to structure data to transmit to external networks. While this represented an enormous leap forward in information accessibility, it was only with the creation of the World Wide Web and web browsers - alongside the increase in public access to computers - that these advancements were made more broadly accessible.

The Web 1.0 was characterized by static web pages connected to resources stored elsewhere on the network by hyperlinks: “a single user-interface to large classes of information” “to link and access information of various kinds as a web of nodes in which the user can browse at will” (Berners-Lee 1990).

#### The Web 2.0

Dinucci 1999. Fragmented Future. darcyd.com/fragmented\_future.pdf

As the proportion of people - consumers - connecting to the Internet rose, the commercial opportunity did as well.

“Read-write web” Dale Dougherty 2004

Social web (O’Reilly

AJAX

Broadband

(Choudhury 2014).

##### The Internet of Things

Increasing connectivity and decreasing equipment costs has led to the inclusion of computing and data transfer capabilities on a widening array of devices. Embedding sensors and connected computing nodes on vehicles, infrastructure, appliances, surveillance equipment (on Earth and in orbit) and so on enables device controllers to access data stored on and captured by the device and, possibly, the ability to actuate device operation. Battery-operated, wirelessly connected computing devices capable of transmitting data representing measurements taken from sensors or other on-board data are being deployed on huge scales; it is projected that by 2025, “there will be 41.6 billion connected IoT devices ... generating 79.4 zettabytes (ZB) of data” (Shirer 2019)

#### The decentralized web

As the records underpinning society’s operation have been digitized, the responsibility for maintaining and updating those databases remained where it had historically resided: with governments, banks, medical institutions and other centralized authorities. While providing this service - incurring substantial cost in terms of capital equipment, operating costs, and labor costs[[27]](#footnote-26) - these authorities continued to be trusted to securely and accurately maintain the records, and to act in the interests of the users, the subjects of the data in their custody.

This presented an agency problem[[28]](#footnote-27) for many of these trusted data custodians: what was in their (usually financial) interest sometimes diverged from the interests of their users. Additionally, their centralized data repositories represented a high value target for attackers seeking to gain access to large quantities of valuable data. “Corporations can argue that data are trickier to manage than oil … The hacker only has to be right once to penetrate a system. Defenders have to parry every jab, all the time; one misstep and they lose” (Economist 2019). This value of these data lakes rose as Web 2.0 companies began to aggregate more and more data about their users.

#### 

##### A peer-to-peer electronic cash system

In October 2008 - shortly after Federal Reserve Chairman Alan Greenspan’s perspicacious observation in “shocked disbelief” of a “once in a century credit tsunami” (Quinn 2008) - a pseudonymous entity called Satoshi Nakamoto published *Bitcoin: A Peer-to-Peer Electronic Cash System* (Nakamoto 2008). The paper described key features of a peer-to-peer network of computers maintaining a ledger representing electronic cash, configured to prevent double-spending. The system would not rely on trust between participants.

On 8 January 2009 Nakamoto released a C++ program implementing the functionality described in the white paper, inviting any member of the public[[29]](#footnote-28) to run a node by executing the program, thereby participating in the network[[30]](#footnote-29) (Nakamoto 2009). By creating this ledger (“blockchain”) and the system required to maintain it, Nakamoto presented a solution to the agency problem faced by centralized data custodians, offering an alternative money system fulfilling the three functions of money[[31]](#footnote-30).

While a description of the complexities of the operation of the Bitcoin consensus network are beyond the scope of this paper, [[32]](#footnote-31). It is composed of a network of computers competing to win the right to validate a block of transactions indicating the transfer of funds from holders to recipients and update the database state.

From the money spender’s perspective, their funds are held at “wallet address” that they created[[33]](#footnote-32). This address is a random-seeming number derived from a public key, in turn derived from a private key they securely generated. To transmit funds to another wallet address, the user must generate a valid transaction, which includes a digital signature created with their wallet address’s private key. This transaction is then transmitted to the Bitcoin network,

Figure x: Example Bitcoin Private Key, Public Key and Wallet Address[[34]](#footnote-33)

|  |  |  |
| --- | --- | --- |
| Private key | Hexadecimal | 43c34ee9af7bfaccca6b3bd5d2af0d96bab09732aa5a3dc63a5eaa7015f2a8ce |
| Binary | 100001111000011010011101110100110101111011110111111101011001100110010100110101100111011110101011101001010101111000011011001011010111010101100001001011100110010101010100101101000111101110001100011101001011110101010100111000000010101111100101010100011001110 |
| Public key | Hexadecimal | 04aca6b60b848e3bb6da4fee5b8e8be30a7acef0ed82ef82e63fe3ba5d56525729fddc63723b5a269a5facd9a5316b47da24191757d3c54e9044f29249e65f3fc4 |
| Binary | 10010101100101001101011011000001011100001001000111000111011101101101101101001001111111011100101101110001110100010111110001100001010011110101100111011110000111011011000001011101111100000101110011000111111111000111011101001011101010101100101001001010111001010011111110111011100011000110111001000111011010110100010011010011010010111111010110011011001101001010011000101101011010001111101101000100100000110010001011101010111110100111100010101001110100100000100010011110010100100100100100111100110010111110011111111000100 |
| Wallet address[[35]](#footnote-34) |  | 19ZvdpcrQdS8fPQQjw7UHCvnTGBoAixL4E |

Miners gather together batches of transactions into a block[[36]](#footnote-35) and assembles a “block header” which, crucially, contains a hash of the previous block[[37]](#footnote-36), as well as a hash representing the transactions included[[38]](#footnote-37). The miner then “repeatedly hash[es] the header of the block and a random number [nonce] with the SHA256 cryptographic algorithm” (Antonopoulos 2017). When an output hash less than a certain number[[39]](#footnote-38) is found, the miner broadcasts the solution and block to the network, updating the state[[40]](#footnote-39).

The system hinges on a small feature of the protocol: when a miner finds the solution nonce, entitling it to update the state, with the state update *it puts new bitcoins into a wallet that it controls*. This is how bitcoin are minted: as rewards for participating in the competition to win the right to update the blockchain to its new state. Miners are incentivized to maintain the computing infrastructure required to sustain such a system, users are in full control of their funds and the agency problem is solved. By combining this insight into human nature with the technical implementation to leverage it, Nakamoto’s was “arguably one of the highest-leverage actions in human history” (Ehrsam 2017).

Figure x: Bitcoin Money Supply

Caption: Every 210,000 blocks the block reward is reduced by 50%[[41]](#footnote-40); “in approximately 2140, almost 2,099,999,997,690,000 satoshis, or almost 21 million bitcoin, will be issued. Thereafter, blocks will contain no new bitcoin, and miners will be rewarded solely through the transaction fees.” (Antonopoulos 2014, Ch 10)[[42]](#footnote-41)

##### A quasi-Turing complete world computer

In 2013 Vitalik Buterin recognized that while the Bitcoin protocol provided some capability to execute scripts upon transaction validation - “a weak version of a concept of ‘smart contracts’” - “the scripting language as implemented in Bitcoin [had] several important limitations”, including a “lack of Turing-completeness”.

Buterin’s key insight was that while performing the state update required to validate transactions, arbitrary computations could be performed by the validating computer, and arbitrary data written to the ledger. He proposed Ethereum, “a blockchain with a built-in Turing-complete programming language, allowing anyone to write smart contracts and decentralized applications where they can create their own arbitrary rules for ownership, transaction formats and state transition functions” (Buterin 2013)[[43]](#footnote-42).

With this adaptation Buterin created a system that, in this author’s view, is a technological innovation representing one of the most profound opportunities for improving the dignity of the human condition in history[[44]](#footnote-43). The decentralized web - Web 3.0, or Web3 - is arising from the realization “that entrusting our information to arbitrary entities on the internet [is] fraught with danger” (Wood 2014). This movement toward decentralization has been occurring, in some ways, since the earliest days of networked computing, and even before (Wikipedia 2019f)[[45]](#footnote-44).

###### The Ethereum Protocol

Based on a Proof-of-Work consensus mechanism similar to Bitcoin[[46]](#footnote-45), the Ethereum protocol extended the functionality of the blockchain network to enable transactions to include a field containing compiled bytecode representing an executable computer program. Transactions specifying the transfer of funds from one account (the sender) to the wallet address[[47]](#footnote-46) another (the recipient) were of course possible, as in Bitcoin, but bytecode could also be deployed as a program available for execution in the Ethereum Virtual Machine (EVM)[[48]](#footnote-47), and the program’s “contract address”[[49]](#footnote-48) returned.

Transactions sent to a contract address will result in the EVM attempting “to execute the contract, … [trying] to call the function named in the data payload of your transaction” (Antonopoulos 2018). These smart contracts were capable of computing data, as well as writing data to the blockchain[[50]](#footnote-49).

###### Constraints of Ethereum

As a public, permissionless service, the Ethereum system needed to disincentivize wasteful or malicious behavior. This was achieved by requiring a fee to be paid by any users seeking to transfer funds, or invoke smart contract functions. Each computational operation has a corresponding “gas” cost, representing the work the network would need to do to perform the operation (Wood 2019 Appendix G). As smart contracts run, the gas costs accumulate; once program execution completes, or a limit is reached[[51]](#footnote-50), the total cost of transaction validation[[52]](#footnote-51) is calculated and deducted from the transaction originator’s account[[53]](#footnote-52). This fee is paid in ether, base currency unit maintained by the system[[54]](#footnote-53). Each block has a total limit to the amount of gas that can be consumed, “to keep block propagation and processing time low, thereby allowing for a sufficiently decentralized network” (jnnk 2015, Ethereum 2019). The gas mechanism protects the Ethereum system by making users “pay proportionately for the computational, bandwidth, and storage resources that they consume”; “thereby disincentivizing attackers” (Antonopoulos 2018 Ch 13).

##### Three forms of decentralization

In this research three forms of decentralization that characterize the decentralized web have been identified . These systems realize the benefits of openness and mitigate the risks through game theoretic incentivization of system participants. They seek to eradicate any “single point of failure”, instead valuing antifragility (Taleb 2013), the quality of being self-healing: they tend to strengthen, on balance, in response to stress, shock and volatility.

###### 1 - Open Source

The first form of decentralization is in the authority to propose and make changes to the code and protocols. This - the open-source model of system and software development - opens access to source code to public scrutiny. In doing so, the pool of potential contributors to a project is expanded, and the likelihood of discovering unintentional errors or intentional abuses included rises[[55]](#footnote-54).

Tools such as git (Torvalds 2005) - a “distributed version control system” (git-scm.com 2019) have radically improved the ability of groups of loosely-coordinated developers to work on the same codebase. In theory, an open-source project is a pure meritocracy. Ideas - code update proposals - are judged by the community on their quality; if worthy as deemed by the community, they are accepted and incorporated. Without a central authority directing efforts, participants are free to pursue solutions they think appropriate[[56]](#footnote-55). Openness improves security[[57]](#footnote-56), diversity and adaptability.

###### 2 - Distributed Ledgers

Second is the decentralization of responsibility to maintain and update informational assets: the so-called “distributed ledger”. This form of decentralization is enabled by the creation of a strict protocol enabling nodes to confirm block validity by recomputing each transaction: “each node verifies the results of each transaction ” (Ryan 2017).

###### 3 - Private Key Custodianship

Third, and most importantly, is the decentralization of the responsibility to hold the private keys needed to interact with the systems. This aspect of the decentralized web is enabled by the properties of asymmetric key ciphers described earlier, namely, that a message recipient (in this case, a blockchain miner verifying the validity of a transaction), given a message and a public key, can mathematically confirm that the message was signed by the corresponding private key. This means that all interaction with these systems can be done by a user without ever transmitting the private key over the Internet. The only way to access the funds in a Bitcoin or Ethereum wallet, or to confirm identity as necessary in the invocation of some Ethereum smart contracts, is to present a valid transaction, including a digital signature. This signature is impossible to generate with the private key (Ker 2014, Antonopoulos 2017, Buterin 2013).

This represents a significant break from the centralized database technologies of the prior web, in which the secrets used to authenticate users (passwords) are stored on a centralized server, and must be transmitted[[58]](#footnote-57) to that server for storage. In a simplified model, when a user tries to log in, the server-side software confirms that the password matches the one that it has in its records[[59]](#footnote-58); if so, the user is considered authenticated, and granted appropriate rights.

Based on Wood’s observation on trusting arbitrary entities on the Internet (2014), this model carries risks: the authorities managing a user’s data might mistakenly delete or alter it[[60]](#footnote-59), reveal it to some malicious attacker[[61]](#footnote-60), or may choose to deny access or revoke service provision[[62]](#footnote-61) (McCoy 2015, Haselton 2017, Hopkins 2017, Fernandez 2019, Galperin 2015, Prince 2019)[[63]](#footnote-62).

##### Emergent technologies

###### Decentralized Autonomous Organizations

Smart contracts enable a broad range of decentralized applications to be developed and deployed on a blockchain network. Of particular interest is that of the DAO: “decentralized autonomous organization”, an organizational structure enabled by smart contract technologies. As with many of these concepts, the collective understanding of DAOs is nascent and few are operating successfully at the time of writing, but generally these entities operate according to rules and data encoded on a blockchain. When deployed on public blockchains, total transparency as to the governance[[64]](#footnote-63), data, membership and activity of the DAO is available for public review[[65]](#footnote-64).

Due to the versatility of smart contracts these rules can enforce any policy, but DAOs can be conceived of as providing members an interface through which they can interact with the informational and financial assets necessary to coordinate organizational projects[[66]](#footnote-65). At this early stage DAOs appear to hold enormous promise to improve organizational efficiency, thereby enabling the provision of services to unserved market segments[[67]](#footnote-66). They also hold the potential to enable as of yet unforeseen organizational structures (Aragon 2019, DAOstack 2019, Rea 2019).

###### Private Blockchains

###### Zero-knowledge proofs

###### Content Addressing

In location-addressed systems, resources are referenced by their address, often a URL referencing a location in a web server’s directory structure, or containing parameters for a database query enabling the server to respond with the requested data.

This, however, represents a single point of failure[[68]](#footnote-67): if a file is moved from its location, the URL does not resolve and the web server returns a “404 Not Found” error; this resource is inaccessible. In the serverless Web3 paradigm[[69]](#footnote-68), this problem is solved by addressing data by its content. The technology is built largely on hashing algorithms and their ability to prove the “sameness” of two equal-length binary sequences. Files are identified by a unique and deterministic hash; a user requesting a specific file could receive segments from various peers, re-assembling it on their local machine[[70]](#footnote-69).

Juan Benet (2014) proposed the InterPlanetary File System (IPFS), “a peer-to-peer distributed file system” based on content addressing. The use of content addresses enable short hashes[[71]](#footnote-70) to reference larger files. Due to the cost of writing to a blockchain, and integrity guarantees, IPFS and other content addressing systems are technologies crucial to the decentralized web.

###### Self Sovereign Identities

“ed: The vulnerability that is being exploited in all systems is identity.” (@santisiri 2019b)

In the emerging vision of the decentralized web, digital identities are controlled and owned by the individuals and organizations they represent: “users [are] the rulers of their own identity”. While the community’s understanding of the idea is nascent and developing, Allen defines 10 principles of self-sovereign identity (2016) (Appendix 1). These developments are enabled by the advancements in networked computing and cryptography described - self-sovereign identities rely on users being able to create and manage private keys securely.

\*\*\*Biometric identification as the biological organisms unique deterministic identity?

What is the synthetic correlate?

###### Sovereign Sensors

Humans interact with the Internet through computers, which detect inputs and establish connections with other computing nodes. In the most literal sense, every personal computer is a sensor: they “sense” keyboard or touchscreen inputs, as well as incoming transmissions from other computers, either wirelessly via an antenna or through some wired input. As such, for the purposes of this research the definition of “connected sensors” includes any digital computing node with the ability to receive and transmit data.

However, a distinction should be made by computers that are controlled by direct human interaction[[72]](#footnote-71) and ones that are autonomously controlled by software agents installed by human developers[[73]](#footnote-72). This research was inspired by a peculiar question: what if these autonomous agents had self-sovereign identities of their own? What if computers had private keys that only they had access to? How might such sensors serve society, and how might they threaten its functioning?

###### Trusted Hardware

Such a capability has been an area of active research for some years, primarily led by the Trusted Computing Group, including “AMD, Hewlett-Packard, IBM, Intel and Microsoft” (Merritt 2003). Detailed discussion of the technical aspects of trusted computing is well beyond the scope of this paper[[74]](#footnote-73), but the fundamental premise is of a chip that contains in non-volatile memory a “public and private key pair, ... created randomly on the chip at manufacture time [that] cannot be changed” (Safford 2003) and cannot be accessed by any external entity[[75]](#footnote-74). With this technology, it seems that computers have a right to privacy[[76]](#footnote-75).

With this technology, ethical concerns abound (Anderson 2003), but if configured properly and governed transparently, such a system could offer myriad ways to improve the security in the emerging Internet[[77]](#footnote-76), and the ability for computer networks to monitor environmental, physical and information security.

The notion of computers in sole and secure possession of private keys attains a new level of significance in the context of the decentralized web[[78]](#footnote-77). Edge devices can be used as oracles in smart contracts without any point of human interference, and a computer can hold funds accessible only to it. A vehicle could detect necessary on-board maintenance - and release funds to perform it. A weather buoy could submit data to a smart contract that had to have originated on the buoy. Access to the imagery captured by a satellite with a synthetic aperture radar or a surveillance camera installed in an urban environment or a balloon (Harris 2019)[[79]](#footnote-78) could be managed by a DAO governed by an open and transparent set of rules, subject to identity-based, spatial and temporal conditions.

It is believed that these technologies represent profound risks and opportunities to change the justness of the systems upon which society relies in the 21st century.

### On resource governance

“Governance refers to all processes of social organization and social coordination” (Bevir 2012). In this context - of computers - governance is closely associated with the concept of access control, the processes by which access to and use of the informational resources on a computer system is managed (Rouse 2019).

A thorough exploration of the theory of governance is beyond the scope of this work. Broadly, however, if governance is conceived of as “processes of rule” (Bevir 2012), the principles underlying governance design, as well as the structures manifesting such processes[[80]](#footnote-79) hold great bearing on the justice of the system: its efficacy in equitably governing the subject community.

The governance of the physical commons is fairly well understood, most clearly conveyed in the eight “design principles” of common resource governance identified by Ostrom (1990). However, these principles seem to primarily apply to the governance of physical resources; no clear consensus on the nature and intricacies of informational resource governance appears to exist. On one hand, information is “a good - … an object of economic transactions”, “typically non-rival and sometimes nonexcludable”[[81]](#footnote-80) (Varian 1998). On the other, “numerous English authorities have affirmed that information or data is not property” (Bilbow 2019)[[82]](#footnote-81).

It seems that informational objects exist in a space with different laws to physical space. Given the commitment to investigating the factors pertaining to the governance of informational resources, the differences between these two spaces in which reality manifests is of acute importance to the research agenda. This will be explored further In the Discussion.

## 

## Methodology

### Overview

#### Purpose

An agent-based approach was employed to investigate the dynamics of sensor networks recording data on a public blockchain. As the scale, resolution and connectivity of the Internet of Things grows, so does its potential to improve situational awareness (Shirer 2019). If leveraged properly, enormous gains in the efficiency and prevention of malicious activity might be realized in the networks monitored by providing system operators a more accurate understanding of systemic patterns and dynamics - with substantial social, environmental, military and commercial implications.

As outlined, public blockchains have unique limitations when compared with traditional database management systems, cloud servers and private distributed ledger implementations. This modeling effort was conducted to explore emergent dynamics of connected sensor networks reporting measurements from the edge to a public blockchain. By understanding these dynamics and the trade-offs inherent to using public and permissionless blockchain architectures, these findings might contribute to helping system designers to take a more informed approach to architecting such systems, thereby balancing the costs and benefits of decentralization.

This model is focused on investigating factors related to network costs and transaction processing times. These scaling questions are important in the context of connected sensor networks, which are often characterized by high data capture and transmission volumes, and provide the most value to system operators if the data captured at the edge is made available for analysis rapidly and is available long term (Gutierrez 2015).

#### State variables and scales

Using python’s (van Rossum 1995) Mesa framework (Core Mesa Team 2019), two subclasses inheriting Mesa’s `Agent` class were defined: `Sensor` and `Blockchain`. A `SensorBlockchainNetwork` subclass of Mesa’s `Model` class served to instantiate an individual simulation, configure model schedulers and collect data from model reporters. Parameter sweeps were performed using Mesa’s `BatchRunner` tool in an iPython (Pérez 2007) notebook (\*\*\*file path); collected data was analyzed and visualized within another iPython notebook (\*\*\*file path) using the `numpy` (Oliphant 2007), `pandas` (McKinney 2010), `statsmodels` (Seabold 2010) and `matplotlib` (Hunter 2007) packages.

##### Sensors

Sensor agents represented edge devices capable of recording empirical data about their surrounding environment. Logic designed to simulate on-device data computation and transmission operations was included within the class definition as methods.

Table x: Sensor agent state variables and scales

|  |  |  |
| --- | --- | --- |
| Name | Value | Parameter / Range |
| battery\_life | Energy level |  |
| mortal | Boolean, whether sensor should be removed from scheduler upon battery depletion |  |
| dead | NaN, replaced with block number upon battery depletion (if mortal) |  |
| record\_cost | Energy consumption |  |
| record\_freq | Probability or interval of record-taking |  |
| record\_bytes | Number of bytes captured per recording |  |
| gas\_price | Price (gwei) per unit of gas as needed in transaction (Ryan 2017) |  |
| stochasticity | Arbitrarily introduced variation to mimic reality |  |
| gwei\_spent | Gwei spent that tick |  |
| last\_sync | Block number of previous transaction submission |  |
| nonce | Iterator for valid transaction generation |  |
| db | Block-indexed array with bytes recorded each tick |  |

Caption: “Elementary properties of the model’s entities” (Grimm 2006).

##### Blockchains

Blockchain agents were designed to simulate key features of “quasi-Turing-complete” (Antonopoulos 2018 Chapter 13) public blockchains, with the Ethereum platform serving as the primary inspiration. These information management systems are characterized by nodes executing programs - “clients” (Stackexchange.com 2019) - developed in adherence to a protocol, which strictly defines computational behavior - the so-called “transactional singleton machine with shared-state” (Wood 2019). While deep discussion of the technical aspects of how a network of computing nodes reliably and rapidly establishes consensus on database state exceeds this paper’s scope, the Blockchain class defined for the model here is based on key features of the Ethereum platform (Antonopoulos 2018).

Of note: the `Blockchain` class does not have a `step()` method. This was intentional, to emulate the inherent reactivity of programs executed on blockchain virtual machines. All computations can be traced back to an origin externally-generated transaction invoking a smart contract. Tasks cannot be scheduled on Ethereum (Antonopolous 2018 Chapter 13).

In the `Blockchain` Agent subclass, attributes including the gas limit per block and gas cost per byte written (both in gwei) were defined upon initiation. These integer values were fixed both through a single model run (unlike in Ethereum, in which block gas limits are adjustable (Antonopoulos 2018 Chapter 13) and across model runs. While the dynamic nature of these parameters are critical to the adaptability and scalability of blockchain systems, these models focused on variation in the size and behavior of the networks connecting to the blockchain in each model run.

Class methods defined allowed for the blockchain network to perform necessary actions, including adding submitted transactions to its mempool (`Blockchain.add\_to\_mempool(tx)`) and mining a block (`mine\_block()`).

The latter method included logic to select the highest value and most recent transactions from the mempool, validate them - including invocation of that appropriate sensor’s `confirm\_tx(tx)` method - and update blockchain `chain` dataframe’s state. [footnote]The `confirm\_tx(tx)` method was designed to emulate the response sent from the blockchain to the client upon transaction confirmation. This reality of smart contract behavior is the reason for the asynchronous functionality of the web3.js library, in which JavaScript’s async await syntax is employed to pause execution of client-side code reliant on transaction confirmation and, possibly, returned values (web3.js 2019). If edge devices were running NodeJS programs to execute on-device computations and transmit data, the web3.js library would likely be used to interact with the Ethereum blockchain.(\*\*\*Sample code with async await syntax?) This would be done by establishing a connection to the network (local, test or main net) and instantiating an instance of the \_\_\_\_\_\_\_\_\_\_\_ class, with methods to connect an ABI and invoke methods of the contract instance. This would be done in a local, private environment as one would need to sign the transaction with the private key of the wallet containing the ether necessary to pay for the gas costs. [/footnote]

|  |  |  |
| --- | --- | --- |
| Name | Value | Parameter / Range |
| block\_gas\_limit | Block limit of computation or data write volumes in gas |  |
| gas\_per\_byte | The gas cost to write one byte of data to the chain |  |
| gas\_per\_second | The gas cost to perform one second of computation on-chain |  |
| avg\_block\_time | Time to mine one block, seconds |  |
| chain | A dataframe of boolean values representing whether sensor recordings have been validated on-chain |  |
| tx\_ct | An iterator so each transaction submitted gets a unique id |  |
| mempool | A dataframe containing transactions, including ‘mined’ boolean column |  |

In the middle-range model developed (Gilbert 2008) blockchain complexity is limited to write operations: no on-chain compute operations were simulated. This notable exclusion of a key feature of smart contract platforms was deemed necessary to reduce model complexity[[83]](#footnote-82).

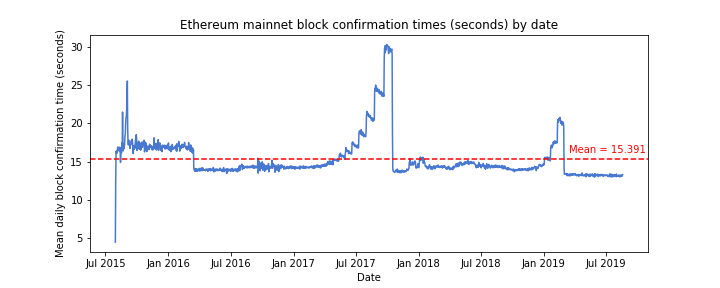
##### Higher-level entities

No higher-level groupings of agents exist, though much opportunity to model heterogeneous networks exists within the modeling framework developed.

##### Scales

The physical location of model agents is not specified. A single tick in the model represents a block confirmation. For context, Figure x depicts block times on the Ethereum mainnet since network launch: 15.391 seconds, on average.

Figure x: Ethereum mainnet block confirmation times, 30 July 2015 - 19 August 2019



Model runs were executed for 300 ticks. Scales were selected based on a series of initial model runs. Visual inspection of the results of these model runs revealed that network block confirmation times and informational currency measures stabilized around the optimal values of 1 block and 1.0 respectively. Increasing network loads through adjustment of record volume, record frequency and network size parameters resulted in these measurements degrading rapidly, representing the network reaching some operational performance threshold. Fixed parameters, and the ranges of swept parameters, were chosen to enable investigation into network dynamics around this threshold.

It is important to acknowledge the middle-range nature of this model. Simulating 300 block confirmations substantially abstracts a blockchain network’s complexity: the value of such a platform is largely found in its persistent operation (cite?). Sophisticated mechanisms enabling the network to adapt to dynamic demands are built into successful blockchain protocols (Nakamoto 2009, Buterin 2013).

However, development of a facsimile model exactly simulating each of these aspects was well beyond the scope possible given time, computational and analytical constraints. While such a model would be difficult to build, it could provide researchers the opportunity to simulate unanticipated stresses on blockchain networks, as well as the emergent complexities of on-chain interactions such as DAO behavior (DAOStack, Aragon), voting and governance mechanism performance, high-demand periods such as token listings (Peaster 2018), dApp launches (reddit 2018, BBC 2017) asset ownership transfers (Mattereum), and so on.

#### Process overview and scheduling

To model the operation of a network of sensors connected to a blockchain, key steps in the process were defined for simulation, while others were lost in abstraction. Judging the relevance of features of a complex system is a primary point where bias can be injected into such an enquiry; good faith efforts were made to capture important aspects.

Actions can be conceived of as taking place in two environments: on board the edge sensor and on the computing nodes forming the blockchain’s consensus network, where transactions are validated and the state is updated, including execution of any smart contract code. The data follows a serial process:

The edge device takes an empirical recording from the edge and stores it in on board memory, such as an SD card.

When ready to transmit data to the blockchain, the edge device gathers all un-submitted data (i.e. data recorded since the prior transaction submission), bundles it into a transaction, signs it and transmits the data to the blockchain network.

Upon reception of the transaction data the blockchain network (or, more specifically, miner) adds the transaction to the mempool, a local data store where transactions await validation.

Each block (tick), the network mines the next block by selecting a set of transactions requiring less gas to execute than the block gas limit. Upon validation, the blockchain updates its state to reflect the inclusion of the data (edge recordings) contained in those transactions.

A message is transmitted back to each edge sensor confirming transaction validation. This enables sensors to keep a record of the financial expenditures required to perform write operations.[footnote]Note: This is not an accurate representation of reality: the accounts and balances are stored on the blockchain itself, and when an externally-owned account’s transaction is validated the transaction costs are deducted from that account’s balance. No communication is required with the submitting edge node to complete transaction validation. However, this structure was used for conceptual simplicity, meaning the Sensor objects kept track of expenditures - relevant for reporting procedures. This meant that costs were measured by adding costs incurred to the Sensor object’s `gwei\_spent` value, rather than seeding sensor-controlled accounts with currency and subtracting from it each transaction validation.[/footnote]

To simulate the lack of coordination in reality amongst edge nodes, Sensor agents were activated in random order. The tick was completed upon block confirmation, at which point the next transactions in the mempool move up the queue; for this reason the final action taken in each model step was the invocation of the `Blockchain.mine\_block()` method.

### Design concepts

##### Sensing

As informational entities, the agents as defined and simulated sense their environment in a few ways. Edge nodes convert the qualities of the analogue data contacting their sensory interface (i.e. moisture, vibration, motion, pressure, temperature, tilt, or image sensor) into some quantitative representation of that quality, represented as a binary data structure (a number, array, array of arrays, and so on). This moment will be explored more thoroughly in the Discussion, as understanding it is critical, in this author’s view, to understanding informational reality and thus the governance of informational resources. However, this is largely abstracted in this model, with only the “volume” of data captured in bytes being explicitly referenced in model execution. Once this analogue-digital conversion is performed, the model focuses on the dynamics of digital data within the information systems studied.

All other agent “sensing” pertains to the detection of signals encoded with binary data, carried on some channel. Specifically, these include the detection of a transaction submitted to the blockchain network (via invocation of `Blockchain.add\_to\_mempool(tx)` within the `Sensor.transmit()` method body), and the subsequent detection of the message from Blockchain to Sensor confirming transaction validation (`Sensor.confirm\_tx(tx)`, called within the `Blockchain.mine\_block()` method. Again, the specifics of this process are almost entirely abstracted. This was necessary to conduct this study within its constraints, but the dynamics of information transfer and dissemination through these networks of computing nodes is very far from irrelevant.[footnote]In fact, resolving issues related to these realities is one of the primary innovations of blockchain consensus networks: to enable nodes separated by network distance to trust that they are each holding the same version of a database. This rather simple-sounding innovation carries enormous implications for the justice of the informational systems that underpin societal coordination, especially regarding the transfer of value between members of society.[/footnote]

##### Interaction

Agents interact by transmitting data to a recipient - Sensor agents submitting transactions to Blockchain agents, and Blockchain agents sending data regarding transaction confirmation to Sensor agents. Exploration of peer-to-peer data transfer systems - in which interactions occur between Sensor agents and between different Blockchains - holds potential to improve system scalability, but was beyond the scope of this research.

##### Stochasticity

A degree of stochasticity was included at some points in the model design to simulate spontaneous or deliberate variations arising in system functioning. Within Sensor agents the stochasticity measure was used to inject variability in the number of bytes captured when a new record was taken. Additionally, if Sensor instance variables `record\_freq` or `transmit\_freq` were assigned float values between 0 and 1.0 exclusive (rather than unsigned integers), a probabilistic conditional test was applied before procession with the `.record()` or `.transmit()` operation. This was meant to simulate variability in the frequency of these actions taking place, either due to some on-device logic, or intermittent Internet connectivity. The degree of randomness is quite system-dependent; if a more accurate model were being built consideration of such randomness would be necessary.

##### Collectives

No higher-level groupings of agents was simulated in this version of the agent-based model. However, much about system performance could relate to these collectives and their interaction. Indicating the owners or manufacturers of devices might enable simulation of changes affecting subsets of the network such as a corporate decision-maker - or a hacker exploiting a software vulnerability - removing a number of devices from participation in the network, or changing their behavior somehow. This is of special concern if edge devices are in custody of private keys controlling financial assets, as suggested in the Literature Review. A malicious actor co-opting a botnet of such sovereign devices could carry much more significant consequences (Sabanal 2016).

##### Observation

Upon completion of all agent activation, on each tick agent-level data was collected for further analysis, including gwei spent, battery life, total data collected and that agent’s current informational currency based on a window of the most recent 30 blocks mined. Additionally, summary statistics describing transaction mining times was collected upon termination of the model run.

### Details

#### Initialization

Upon initialization, a Blockchain object is instantiated with an empty mempool - akin to the genesis block. A specified number of Sensor agents are also instantiated with instance variables specified, including reference to the Blockchain instance just created. Sensors are added to the model scheduler. Fixed values and variable parameter ranges are shown in Table x.

Table x: Initialization variables - fixed parameters

|  |  |  |
| --- | --- | --- |
| Object class | Variable | Parameter value or range |
| SensorBlockchainNetwork  (Model subclass) | stochasticity | 0.05 |
| num\_sensors | {s: s = n \* 50, n ∈ [ 1, 2, … 10 ] } |
| info\_currency\_window | 30 |
| Blockchain  (Agent subclass) | block\_gas\_limit | 8000000 |
| gas\_per\_byte | 625 (Ryan 2017) |
| Sensor  (Agent subclass) | battery\_life | 10000 |
| mortal | False |
| record\_freq | [0.01, 0.1, 0.3, 0.5, 0.7, 0.9, 1] |
| record\_bytes | {r: r = 1 + n \* 20, n ∈ [ 0, 1, 2, … 25 ] } |
| sensor\_gas\_price | 20 |

#### Input

No environmental inputs were introduced during model execution; all activity was in response to pre-programmed agent behaviors and state variables as initialized.

To move toward a facsimile model of connected sensor networks, the programming of responsiveness of sensors to environmental conditions represents an interesting line of further research. For example, some sensor monitoring coral reef health could transmit a transaction in the event of reef damage, triggering the release of funds to some local repair crew [footnote]This type of parametric insurance is being considered by the insurance industry; Willis Towers Watson recently launched reef insurance for sites in central America (Vincent 2018). It is unknown, however, if such parametric policies are connected to smart contracts holding funds for disbursement, or how parametric triggers are monitored and policies paid in the event of conditional thresholds being exceeded. A recent Lloyd’s of London report suggests it is unlikely that such a system has been connected to smart contracts (Lloyd’s 2019)[/footnote], or submit a transaction containing relevant information if an autonomous ship strays from a buffer around its intended route.

#### Submodels

On model instantiation a single instance of the Blockchain class is created. The specified number of Sensor objects are created and the model Blockchain object is added as an instance variable. [footnote]For these model runs the blockchain object could have been assigned the Sensor objects as a class variable. However, this design pattern was chosen to leave room for a model design that includes the simulation of multiple blockchain networks, with different sensors connecting to different chains - or even the same sensor submitting transactions to different chains based on some logic executed on the device.[/footnote] In this way, each sensor attained the ability to invoke Blockchain object methods. Sensors were then added to the model scheduler - but the Blockchain object was not.

Each tick a number of operations were executed.

In an order randomized each step, Sensor agents were activated by invoking their `.step()` methods. If the Sensor was active[footnote]In order to enable modeling of edge sensors constrained by limited energy supplies, a `mortal` boolean was included. If the `battery\_life` instance variable was depleted below zero, the Sensor instance would not perform on-device actions such as recording and data transmission. These sensors were not simply removed from the scheduler because they might still have unvalidated transactions pending in the mempool, and needed to be accessed via model scheduler for the invocation of `confirm\_tx()` method by the blockchain upon confirmation. This would ensure complete recording of Sensor costs. For the simulations analyzed here, however, edge devices were not mortal - i.e. they were connected to a reliable power source, rather than a limited battery.[/footnote], this process began with invocation of the Sensor’s `.record()` method. If the sensor was determined to attempt a transmission that tick (based on comparison of its `transmit\_freq` variable with either a randomly-generated float value (probabilistic) or the current block number (deterministic, interval-based)), then a `.transmit()` method simulated the preparation and signing of a transaction, transmitted to the blockchain network. [footnote] A `compute()` method to simulate data reduction by analyzing data on the edge device was defined, but not utilized in the batch runs performed. This code would enable modeling of energy usage at the edge, and would have implications for each of the dependent variables measured in model runs: mining times, financial costs and informational currency. [/footnote]

Within the `.transmit()` method execution the sensor invoked the instance’s `blockchain.add\_to\_mempool(tx)` method, simulating the submission of the valid, signed transaction to the blockchain miners to add to the queue of unvalidated transactions.

Within the model `.step()` method, after randomly activating each Sensor agent, the Blockchain instance’s `.mine\_block()` method was called[footnote]This aspect of model design would need adaptation to simulate the operation of multiple blockchain networks. If the activation of all Sensor agents prior to the Blockchain agents mining action is as important as assumed, the `RandomActivationByBreed` subclass defined in Mesa’s wolf\_sheep/schedule.py example may achieve this. (Mesa 2018) [/footnote], in which a subset of unvalidated transactions was selected from the mempool. [footnote]Transactions were selected for inclusion in a block based on transaction value, to simulate the miners’ incentive to mine highest-value transactions first, then block submission time, to prevent transactions arbitrarily remaining unvalidated (getting “stuck”). It is worth noting that transactions can get stuck in the Ethereum mainnet mempool, remaining unvalidated (usually because the gas price is so low that miners do not select it for validation) (McDonald 2017). It was decided to remove this feature to simplify the modeling of network informational currency; it is unknown how its inclusion might affect model behavior. Furthermore, although gas prices were fixed in this research, the modeling of variable gas prices could shed light on the costs of increased informational currency in heterogeneous agent networks. This transaction selection approach should capably handle such an adaptation.[/footnote] Upon transaction validation, the `chain` dataframe was updated such that the appropriate Sensor’s column reflected the current state of on-chain data availability. Additionally, the Sensor agent’s `.confirm\_tx()` method was invoked, simulating the deduction of the gwei spent for transaction validation from the agent’s account.

### ODD

### Model

#### Instantiation

#### Scheduling

For each model run, a single Blockchain agent was instantiated.

#### Data Collection

Gwei spent: In this middle-range model “gwei” is simply borrowed from the Ethereum lexicon; it represents a theoretical cost, and should not be thought of in terms of value in fiat currency. Rather than attempt to estimate true costs, this dimension was modeled in an effort to understand trends in total transaction costs per sensor across the parameters swept.

Informational currency: Much of the value of awareness-enhancing networks is gained by its provision of timely information to system users. Calculation of mean informational currency for each iteration excluded the warm-up period of 30 ticks. The metric quantified the proportion of data captured at the edge reflected on chain, having completed the full process of recording the data, transmitting a transaction, and that transaction being validated through mining. To

#### Execution

Each tick represented a block validation by the consensus network maintaining the blockchain, during which the blockchain’s state was updated to incorporate data included in each transaction mined.

Number of ticks

#### Parameter Sweeps

* Fixed parameters
* Each sweep
* Number of iterations at each sweet - computational costs

## Results

Analysis of data captured during model runs revealed interesting system dynamics, though questions remain regarding whether the patterns observed reflect system operation in reality.

This investigation sought insight into the effects of three independent variables on three independent variables. Each IV-DV pair will be reported, with extra attention paid to interesting and unexpected behavior.

### Network size

The network size was defined as the number of sensors submitting transactions to a blockchain. Model simulations were executed ranging from 50 to 500 sensors, on intervals of 50.

##### Mining dynamics

A positive relationship between network size and mean transaction mining times per transaction was observed: larger networks tended to validate transactions more slowly. For each model run, a mean mining time was calculated by averaging the difference between block submitted and block mined for all transactions.

Table : Summary statistics, Mean mining times per transaction across network size parameter sweep

|  |  |
| --- | --- |
| n | 30 |
| Mean | 47.34600 |
| σ | 33.75534 |
| Minimum | 1.00404 |
| Median | 46.59344 |
| Maximum | 95.18947 |

The observed mean mining times exhibited a positive correlation (Pearson’s correlation coefficient r\_xy = 0.971728). It stands to reason that this correlation is due to the block gas limit, which fixes a limit on the amount of data the network can write each tick. Transactions each specified the same gas price, and data capture volumes and transmit frequencies were homogeneous across sensors, so network size directly correlated with the volume of transactions being submitted to the virtual machine for validation. Because blocks have a fixed limit to the gas that could be consumed - here, the amount of data to be written - in larger networks generating more data, transactions will take longer to be validated.

An ordinary least squares simple linear regression analysis was conducted to produce a line of best fit modeling the relationship between network size and mean mining times.

Figure : The effect of network size (number of sensors) on mean mining time per transaction (in blocks)

[insert figure here]

Caption:

While the correlation coefficient indicates a strong positive correlation between the variables considered, the residual errors do not appear to be normally distributed, calling into question the validity of the line of best fit calculated using OLS simple linear regression.

Curiously, the observed values at a network size of 200 breaks with the trend established in smaller network sizes; mean mining times were, on average, less than half of the values observed in sensor networks of 150 sensors. It is unclear if this break in the otherwise consistent positive relationship between the two variables is due to some quirk of the model or true emergent behavior of these systems interacting in reality; the former seems likelier, but without a dataset to validate observations from the simulation this is difficult to assess.

Visual inspection of the scatterplot suggests the possible subsequent establishment of a new, convex positive non-linear relationship for networks of 200 or greater sensors. Further investigation of the nonlinear effects of transaction volumes on mining dynamics is warranted, but beyond the scope of this investigation.

##### Gwei spent

The mean gwei spent per sensor was calculated for each model run to gain insight into the effects of network size on financial costs to system participants. Transaction costs disincentivize malicious or wasteful behavior; the higher marginal costs represent a primary way public blockchains differ from traditional data storage and cloud computing resources.

Table : Summary statistics, Mean total gwei cost per sensor across network size parameter sweep

|  |  |
| --- | --- |
| n | 30 |
| Mean | 10.44038 |
| σ | 0.64924 |
| Minimum | 9.67701 |
| Median | 10.27899 |
| Maximum | 11.73449 |

Figure xxx: Network size (number of sensors) against mean gwei spent per sensor per block

Figure xxx depicts mean gwei spent per transaction across the network sizes modeled; a concave negative non-linear relationship is clearly seen.

Plotting the log of network size against gwei spent per sensor per block yielded an apparent negative linear relationship. An OLS simple linear regression analysis produced a model with an R-squared value of 0.9938, indicating that the network size strongly predicts mean gwei spent per sensor. Figure xxx visualizes the scatter plot of the transformation and the calculated line of best fit.

Figure xxx: Log of network size (number of sensors) against mean gwei spent per sensor per block

Figure xxx: Residual errors vs predicted gwei spent per sensor per block

Visual inspection of the residual errors plotted against predicted values based on the model, however, shows that the errors are not normally distributed, invalidating the OLS simple linear regression as a method for assessing the significance of the relationship between the variables. Still, the negative relationship between the variables is irrefutable.

As with mining dynamics, the decrease in gwei spent per sensor per block observed with increasing network sizes is caused by the static block gas limit. As the number of sensors attempting to submit data to be stored on chain increases, the likelihood of each sensor's transactions being validated, contained data being written, and gas costs incurred, decreases. Costs in gwei are only deducted from the sensor's externally owned account upon transaction validation.

Oddly, in model runs with networks larger than 150 sensors, mean gwei costs per sensor were the same across the three iterations executed at each network size simulated, while some, albeit minor, variation was observed in smaller networks (Table xxx). Due to the inclusion of the stochasticity metric, as well as a probabilistic record frequency value, this was not expected. Regardless, it almost certainly is a quirk of model design and not an indicator of actual system behavior. (\*\*\*interpretation?\*\*\*)

Table xxx: Standard deviation of mean gwei spent per sensor per block across identical model runs

|  |  |
| --- | --- |
| Network size (sensors) | σ - mean gwei / sensor / block spent across iterations |
| 50 | 0.014425 |
| 100 | 0.000423 |
| 150 | 0.000339 |
| 200 | 0.0000 |
| 250 | 0.0000 |
| 300 | 0.0000 |
| 350 | 0.0000 |
| 400 | 0.0000 |
| 450 | 0.0000 |
| 500 | 0.0000 |

##### Informational currency

Interesting and unexpected patterns of network informational currency were observed across the parameter space - and across the time series of individual model runs. A linear negative correlation between network size and mean informational currency values was expected, as was stationarity within model run time series plots. While the pattern expected was observed across model runs, this was not the case within individual iterations.

###### Time Series

Figure xxx depicts time series plots of mean network informational currency by tick for each of the model iterations recorded, colored by network size. In the smallest networks (50 sensors), informational currency rapidly rose in early ticks and - prior to tick 30, at which point older blocks were excluded from the measure calculations - reached a maximum value of > 0.9.

Table xxx: Summary statistics, Mean informational currency of 50-sensor networks (t > 30)

|  |  |
| --- | --- |
| n | 270 |
| Mean | 0.911018 |
| σ | 0.011579 |
| Minimum | 0.87778 |
| Median | 0.91111 |
| Maximum | 0.93978 |

Caption: Summary statistics for mean network informational currency measured at each tick.

After the warm-up period for these 50-sensor networks, mean network informational currency was a stationary process, as evidenced by Augmented Dickey-Fuller test results (Table xxx). Due to p-values of < 0.05, the null hypothesis is rejected: these time series do not have a unit root, and are stationary.

Table xxx: Augmented Dickey-Fuller test results, mean informational currency over time for 50-sensor networks

|  |  |  |
| --- | --- | --- |
| Iteration | ADF Statistic | p-value |
| 1 | -7.2910 | < 0.05 |
| 2 | -5.3041 | < 0.05 |
| 2 | -6.3199 | < 0.05 |

Networks larger than 50 sensors exhibited unusual behavior: after achieving a maximum mean informational currency after warm-up, the measures decayed over time until a threshold was reached, at which point the process became stationary. The greater the maximum measured value, the slower the decay rates to the floor threshold. Specifically, ~0.5 appeared to serve as a threshold support level; any networks that exceeded this mean currency receded to this point over time, then achieved stationarity there. Networks of 100 and 150 sensors exhibited this behavior. In networks of 250 or more sensors, which did not achieve a mean informational currency of >= 0.5, after the initial 30-tick warm-up period the measure diminished, ultimately receding to 0. The closer these larger networks got to this threshold level, the longer it took for them to recede to 0.

The eventual establishment of stationary processes around these threshold mean informational currency levels was unexpected; it is again unclear if these behavior patterns are due to a quirk of model design or represent a valid emergent dynamic of these complex systems. The decay in the measurements is likely due to the blockchain block gas limit: if more data is being generated each time step than can be recorded on the ledger, and because mining prioritizes earlier transactions over more recent ones (given equal gas costs) (cite line in python script), transaction mining times will increase as the number of unvalidated transactions in the mempool grows. Higher mining times and lower informational currency measures can be attributed to the common cause of block gas limits.

However, the threshold mean informational currency of 0.5 observed in smaller networks remains unexplained. Despite thorough review of model source code, no obvious point causing such behavior was found. Further investigation into this unexpected emergent dynamic would help to identify its cause, and to understand if it represents a result representative of actual system behavior or simply a quirk of model design.

\*\*\* Could add analysis here of slope of lines decaying to threshold / 0 based on network size … if we have time and space \*\*\*

* Identify transition from stationary to non-stationary processes
* Verify with ADF tests of time series subsets
* Perform linear regression on non-stationary segments
* Compare regression coefficients across network sizes; comment on effects of network size on slopes

### Recorded Data Volumes

Empirical sensors operating at the edge record data representing some quality of their environment; the information contained in these data has value to other informational entities seeking insight into conditions in the vicinity of that sensor. Given the high costs of storing data on a public blockchain, a sweep of the quantity of data in bytes captured in each edge recording was performed. Here the effects of changes in data volumes recorded in each observation on the dependent variables of mean transaction mining times (in blocks), financial costs (in gwei) to each sensor and network informational currency are analyzed.

#### Mining dynamics

A positive relationship between data volumes recorded per observation and mean transaction mining times is visible in Figure xxx: as sensors capture more data per observation, transactions tend to take longer to mine (Pearson’s correlation coefficient = 0.74258). Excluding two outliers, for which mining times exceeded the Tukey fence of 18.567, a Pearson’s correlation coefficient of 0.78448 was calculated. Summary statistics for the sample excluding outliers

Summary statistics: Mean transaction mining times across edge record volumes parameter sweep

|  |  |
| --- | --- |
| Statistic | Value |
| n | 76 |
| Mean | 7.8928 |
| σ | 3.9433 |
| Minimum | 1.0010 |
| Quartile 1 | 5.9778 |
| Median | 7.8253 |
| Quartile 3 | 10.712 |
| Maximum | 16.535 |

(REMOVE? - Visual inspection of the plot of record volumes against mean transaction mining times reveals a heteroscedastic distribution: as record volumes increase, variation in observed mining times increases as well. This “cone” shaped distribution means that residual errors from an OLS line of best fit will not be normally distributed. However, rather than testing the significance of the relationship, an OLS simple linear regression was performed to calculate the slope of the trendline.)

Figure xxx: Sensor data capture volumes (bytes per recording) against transaction mean mining time (blocks)

Caption:

Figure xxx: Mean mining time residual errors versus values predicted from OLS simple linear regression

Figure xxx: Distribution of residual errors of mean mining times

Figures xxx depict the plot of record volumes against mean transaction mining times with the line of best fit. The calculated R^2 value of 0.61541 indicates that the independent variable accounts for ~61.5% of the variance observed in the dataset. Visual inspection of Figures xxx and xxx suggest that residual error is normally distributed; because the p-value calculated is less than the significance level of 0.05, we fail to reject the null hypothesis. The mean mining time increased by 0.02051 blocks for each additional byte of data recorded per sensor observation.

#### Gwei spent

As Figure xxx shows, record volumes had an unusual effect on gwei spent per sensor. At smaller record volumes, the financial expenditures rose rapidly with the independent variable, quickly leveling off between 350000 - 400000 gwei spent per sensor. These mean recordings appeared to begin to decrease as record volumes approached 241 bytes, at which point two highly uniform segments of data are seen.

First, in model runs simulating edge recording volumes of 241 - 321 bytes, a very strong positive linear correlation is observed (Pearson correlation coefficient: 0.99997). Figure xxx depicts this segment of the sample, including the OLS line of best fit - note the R^2 value of x, indicating that x% of variance is explained by the independent variable.

Figure xxx: Data capture volumes (241 <= bytes <= 321 ) against mean gwei spent per sensor

Between 321 and 341 bytes per record, average gwei expenditures drop dramatically and begin another, less steep upward trend, also a positive linear correlation (Pearson’s correlation coefficient = 0.99999) which continues to the end of the parameter sweep. This segment of the sample is shown in Figure xxx; an R^2 value of xx indicates that this positive linear relationship is very strong.

Figure xxx: Data capture volumes (341 <= bytes <= 501 ) against mean gwei spent per sensor

While these results are interesting, it seems unlikely that these near-perfectly correlated relationships are due to something other than an unintended aspect of model design. As such, these results are determined to be not useful to this investigation into the dynamics of a scaling blockchain network.

(If time - look at Segment 0 - prior to perfect linear segments)...

#### Informational currency

As record volumes increased, informational currency was expected to decrease, due to limitations in the amount of data that could be written to the blockchain each tick - the block gas limit. As shown in Figure xxx, this was observed: after the warm-up period, in each model iteration a mean level of informational currency was established. Variations around this mean are explained by introduced stochasticity and the probabilistic transaction transmission frequency.

Figure xxx: Informational currency over time across a range of data capture volumes

Calculating mean informational currency values for each model run (excluding the warm-up period) enabled the visualization of the effects of data capture volumes on the stable informational currency levels depicted in Figure xxx (time series). A non-linear negative relationship is visible - perhaps a sigmoid curve. Interestingly, as the curve approaches its lower horizontal asymptote at IC ~= 0.2, a point is reached between 321 and 341 bytes per record where the dependent variable drops to a new stable level of ~ 0.09. The cause of the measured values dropping at this threshold is unclear.

(\*\*\*If time and space: investigate temporal autocorrelation of informational currency\*\*\*)

#### 

### Sensor observation frequency

Sweeping the frequency with which edge sensors recorded empirical observations about their environment was intended to yield further insight into the effects of network activity on blockchain behavior.

Initial examination of results obtained from sweeping record frequencies using fixed parameters as in other independent variable sweeps, it was evident that the model behaved predictably up until near the limit of the sweep. Notably, with fixed network sizes of 20 sensors, the maximum gwei spent per sensor per block of 390000 meant that the block gas limit of 9000000 gwei was never reached.

Since these models were intended to simulate the challenges blockchains might encounter while scaling, and block gas limits are one of the primary constraints to network scalability, the parameter sweep was performed after doubling the number of sensors simulated in each iteration. In these sweeps, block gas limits were reached in the median swept values, enabling the analysis of model behavior as the network transitioned from underutilized to oversubscribed. This is noteworthy because for the analysis below, fixed parameters were identical to investigations into the independent variables’ effects conducted above, with the exception of the number of sensors in each network iteration.

#### Mining dynamics

Initial examination of the scatter plot of mean mining times across record frequencies indicated that for record frequencies below 0.2, transactions tended to be mined immediately after submission. For simulations in which sensors took more frequent recordings, an exponential positive relationship was visible; this segment of the sample was analyzed. A OLS linear regression of record frequencies and the square root of mean mining times yielded a line of best fit shown in Figure xxx, with an R^2 value of 0.99243.

Figure xxx: Record frequency (probability per block, > 0.2) vs square root of mean transaction mining time (blocks)

Figure xxx: Residuals vs fits - record frequency vs square root of mean transaction mining time

Analysis of this plot, and of the residual errors (Figure xxx) reveals that the square root transformation did not fully straighten the positive nonlinear relationship observed in the untransformed data. Performing an additional square root transformation yielded a distribution of residual errors nearer to normal, though uncertainty remains as to the true nature of the nonlinear relationship observed in the untransformed dataset (Figures xxx and xxx). Deeper investigation of these nonlinearities is beyond the scope of this analysis.

Figure xxx: Record frequency (probability per block, > 0.2) vs double square root of mean transaction mining time (blocks)

Figure xxx: Residuals vs fits - record frequency vs double square root of mean transaction mining time

It does seem clear, however, that once a certain threshold level of transaction activity per block is reached, mining times per transaction rise in a nonlinear fashion. This result is important, and deserves further investigation, as it indicates that blockchain update performance would diminish rapidly as demand increases. Substantial research into mechanisms for managing these scaling challenges due to the inevitable fluctuations in network demand is ongoing (cite - sharding?); a solution to seems necessary if blockchains are to achieve their potential to form the critical informational infrastructure of the web.

#### Gwei spent

Figure xxx shows that mean gwei spent per sensor per block exhibited a positive linear correlation with record frequencies until a limit was reached, at an approximate record frequency of 0.5, after which the level remained roughly constant. This limit occurred at just below 200000 gwei per sensor per block. (Note the small difference between the median, third quartile and maximum values in Table x.) In a 40 sensor network, this is almost certainly caused by the block gas limit: higher gwei expenditures would have exceeded the block gas limit, an impossibility according to the blockchain protocol. (Rather than exceed this limit, unvalidated transactions are simply left in the mempool to be validated in a future block.)

Table x: Summary statistics, Mean gwei spent across record frequency parameter sweep

|  |  |
| --- | --- |
| Statistic | Value |
| n | 21 |
| Mean | 135006.0 |
| σ | 77359.1 |
| Minimum | 3818.8 |
| Quartile 1 | 42969.8 |
| Median | 192168.8 |
| Quartile 3 | 194968.8 |
| Maximum | 195385.4 |

Figure xxx:

An OLS simple linear regression was performed to model the linear relationship between the variables at record frequency values less than 0.5; this plot, including the line of best fit, are depicted in Figure xxx. An R-squared value of 0.99924 corroborates the visual assessment of a near-perfect linear relationship.

Figure xxx: Record frequency (< 0.5) vs mean gwei spent per sensor per block with OLS line of best fit

(deprecated vvvv

A strong positive correlation (Pearson correlation coefficient: 0.99897) was observed between the recording frequency and the gwei spent per sensor, in line with expectations. Table x contains summary statistics. Of note: with fixed network sizes of 20 sensors, the maximum gwei spent per sensor per block of 390000 meant that the block gas limit of 9000000 gwei was never reached. This limited the potential insight into the behavior of the network as it scaled. Due to this, the parameter sweep was extended below.

First, an OLS simple linear regression analysis was performed to model the relationship between record frequency and gwei expenditures. Figure xxx depicts a scatter plot of the two variables, along with the line of best fit.

Figure xxx: Sensor data capture frequencies against gwei spent per sensor per block

The distribution of residual errors (Figure x), however, is not normal - the error observed at record\_freq = 1.0 is outside the Tukey fences of the sample (k=1.5), and constitutes an outlier. Once again, this does not negate the positive correlation observed, it simply mean statistical significance cannot be placed on the Ordinary Least Squares regression model calculated.

##### Extending parameter sweeps

Because the block gas limit was not exceeded in the record frequency parameter sweep, an additional sweep was performed with 40 sensors per network.

)

#### Informational currency

Measures of informational currency collected from model runs across the parameter sweep revealed that a recording frequency of ~0.5 yielded resulted in the highest values. Less frequent updates meant the blockchain was not updated often enough to result in high measures of the metric; as record frequency (and therefore transaction volumes) increased above this optimal level increasing mining times adversely affected network informational currency.

Figure xxx: Informational currency over time across recording frequencies

Figure xxx: Record frequency vs mean model run informational currencies (excluding warm-up period)

Interestingly, the model results depicted in Figure xxx show that at higher record frequencies, from a time series perspective within model runs informational currency measures tended to achieve their maximum toward the end of the warm-up period (equal to the window used for measuring the metric), then decayed to a stable level, at which point a stationary process was established. This pattern only occurred with record frequencies greater than 0.5. It seems that stable states may exist at different transaction volumes. As often, distinguishing between authentic system behavior and unintended behavior resulting from model design is difficult. Nonetheless, the result is interesting, and warrants further research.

### Further Investigation

* The thresholds in mean informational currency observed across network sizes
* Heterogeneity in sensors
* Learning element: sensors adapting transmit frequency, on-board compute (data reduction), gas price in response to transaction confirmation times.

## 

## Discussion

## This research has explored the technologies constituent to a system of trusted sensors acting as oracles to smart contracts deployed on blockchain networks. A middle range agent-based model of such a system was developed; initial results were analyzed. The insights gleaned from these simulations and analyses, along with a broader conversation of the implications of this system, is presented here.

### IoT + Blockchain

Public blockchains can only function if the costs associated with utilization of the public resource deter excessive, lazy or malicious use. Sensors embedded on devices generate large quantities of data. On the surface, blockchains are not a sensible informational architecture to store or compute IoT data. This intuition was validated by the observation of diminishing informational currency measures and increasing costs associated with increasing network loads (\*\*\*reference section).

However, solutions such as edge computing and content addressing offer the opportunity to dramatically reduce data loads without necessarily reducing the user’s access to data. Designers of IoT systems connecting to blockchain networks will need to balance the trade-offs between edge resource consumption, financial costs and informational availability. In the near future, focus should be placed the highest-value use cases for connecting edge devices to smart contracts, where the cost could be justified. Parametric insurance products, which trigger a payout if some threshold condition[footnote]Flood waters reach a certain height in a building, or a flight is delayed by a number of minutes.[/footnote] is reached, seem an optimal initial use case (Lloyd’s 2019). While this technology is still in its infancy experiments should be conducted such that the harm caused by failure would be inconsequential.

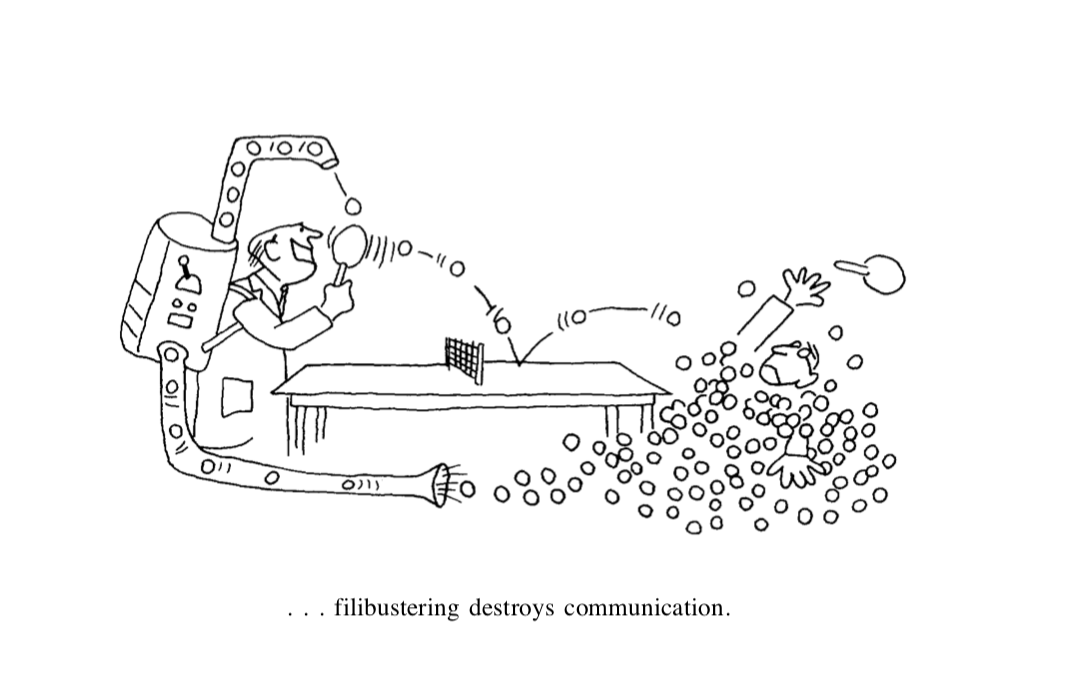
#### Proposing Ricardian treaties

Once matured, trusted sensors connected to blockchain networks might provide a transparent way for uncoordinated entities[footnote]i.e. ones with no trusted central authority[/footnote] to maintain situational awareness. This author is especially interested in applications of the technologies by sovereign nations holding one another to account. Arms control represents a particularly high impact application of this system of oversight. As an example, if secured in smart containers embedded with trusted sensors (Becha 2019), a protocol of intermittent data transmissions including location and local environmental characteristics could provide governments confidence that their adversaries are behaving in compliance with mutually-agreed rules. Authority could be designated to such smart contracts in the terms of treaties and other interstate agreements; technical mechanisms incentivizing proper behavior[footnote]Such as staking of funds, forfeited upon treaty violation, or the automated enforcement of sanctions on participating states that fail to adhere to agreement terms[/footnote] could supplement legal ones, which can be slow and difficult to implement. Of course, privacy and security concerns about the entities authorized to access this sensitive information are valid; it seems like that a private, permissioned blockchain implementation[footnote]Likely with nodes hosted by adversarial governments, as well as entities with a range of incentives.[/footnote] would be more appropriate and palatable to sovereign nations considering such an arrangement[footnote]This may be another situation in which homomorphic cryptographic techniques and zero-knowledge proofs could provide assurance to adversarial states without revealing additional strategic information (Glaser 2014).[/footnote]. Of course, many applications beyond arms control are conceivable, including in customs and immigration, international finance, military intelligence sharing, environmental monitoring, disaster response, conflict mitigation, etc. Such Ricardian treaties could bring many of the benefits of decentralization to international affairs, and improve sovereign nations’ ability to leverage the improving capacity to maintain situational awareness around the world.

#### A DAO for access control

This work was inspired in part by the author’s recognition of an enormous opportunity being missed. Legions of sensors are being deployed and connected to the Internet - in theory, able to provide information feeds to people who could derive value from that information. However, due to the current configuration, based on the client-server web paradigm and an approach that is often required to place private interests over public ones[footnote]Many sensors would not be installed if their owners could not monetize them.[/footnote], much of the value to humanity being created is lost as the relevance of the data fades, it left siloed on its device or in a proprietary relational database.

This is, in this author’s view, a tragedy of the informational commons[footnote]One of three identified. The second is when a channel’s openness is abused, rendering it unusable for others - Distributed Denial of Service attacks are an example of this (Beal 2019). (Licklider 1968 pp 35) (DDoS attacks)



A possible third: not admitting to a security breach from embarrassment or due to reputational cost: “many firms treat hacks like gonorrhoea, an embarrassing affliction no one wants to admit even if speaking about it would stop its spread. Some call it a tragedy of the cyber-commons.” (The Economist 2019)

[/footnote]: the failure to share information with someone who would benefit from it. There are certainly valid instances where sharing such information would provide a competitive advantage to the informee, but in many cases this is not so.

As a solution to this, a peer-to-peer data access protocol is envisioned. Edge nodes would need to be visible to a central administrator - in this conception, a DAO hosted on a blockchain. Data consumers could - for a fee, or under certain conditions, or possibly if assessed as deserving by some decentralized review mechanism - connect to edge networks and retrieve data for analysis, even in real-time. This could provide sensor owners an additional source of revenue - and they could price the data according to their perception of its value - and it would provide the public an opportunity to extract value from the unprecedented capability for situational awareness emerging.

A thorough exploration and feasibility study of this system is beyond this paper’s scope, but it seems that work related to decentralized public key infrastructures (Allen 2015), content addressed storage (Benet 2014), and proxy re-encryption techniques (Nuñez 2018) might resolve many of the most difficult challenges to its implementation.

### Toward a theory of conceptual reality: expanding on “objects”

This research has required the development of an accurate understanding of information and communication technologies. To that end, as much knowledge regarding the physical behavior of computers was attained[footnote]with no formal training in physics, electrical engineering or materials science.[/footnote]. Clearly the inventors of these technologies understand the physical and conceptual properties of the objects and materials that forms the basis of their inventions. Still, some questions regarding the nature of information remain unanswered. These questions point to an incomplete understanding of the nature of information. Here this author’s current understanding of the nature of conceptual reality is outlined.

Based on the insight that physical resources fundamentally differ from informational ones, it appears that a distinction must be made between physical, tangible[footnote]discrete? As in, contained within a continuous surface. Is someone’s liver an object if it is inside their body?[/footnote] objects and conceptual or informational objects. Here the char

#### The Conceptual Space

Physical objects exist in the physical space, and adhere to the physical laws. Conceptual objects, however, do not appear to exist in a physical space[footnote]Or at least do not primarily manifest there. Initial enquiry suggests that each conceptual object is necessarily grounded in some physical manifestation.[/footnote], but rather in conceptual space: within the awareness of a perceiving, or informational, entity. An informational entity should be understood as a physical object[footnote]Could informational entities exist in the conceptual space? As all conceptual objects seem to arise from a physical manifestation, maybe not.[/footnote] that is capable of perceiving its environment in some meaningful way, i.e. it can adapt its behavior based on the stimulus.[/footnote] capable of perceiving its environment in some meaningful way, such that it can adapt its behavior based on the stimulus[footnote]By this definition many “active” objects (Figure 1) are informational entities: amoeba can perceive their environment, discern meaning, and react, as can a computer, and possibly even a tractor (in which the gas pedal and steering wheel would be the sensory organs?).[/footnote]

#### Characteristics of conceptual reality and the process of perception

The unmanifest conceptual space is ever present and global[footnote]i.e. universal[/footnote] - all places at once[footnote]The meaning contained within the universe is infinite at every point[/footnote].

Without sentience conceptual reality does not manifest. Within a sentient entity’s private awareness, it does - to the depth that the awareness perceives.

Manifestations within conceptual reality originate in some physical data, and do not manifest without a physical cause[footnote]Necessary but not sufficient.[/footnote].

Data is physical; information is conceptual. It seems as though data is very closely related to - or perhaps just is - energy. It is emitted or reflected by a source and travels through the “‘ether’ the hypothetical invisible medium that permeates the universe” (Buterin 2014b)[footnote]This needs to include air, as the auditory channel carries data, as well as conductive materials, which can carry data on an electrical signal. Data can also be conveyed through two objects physically contacting each other.[/footnote], carrying information about the qualities of its source[footnote]Note: here, “data” refers to analogue data. Analogue data exists on a continuous spectrum; digital information exists on a discrete spectrum. It is only within conception that discrete entities arise; digital information only exists within an informational entity’s awareness. By this reasoning, it is possible that “digital data” is a misnomer. This digital information is, as described in the Literature Review, encoded onto an analogue signal. Investigation of the differences between analogue and digital holds promise for explaining the nature of conceptual reality. Taking light as an example, a key distinction between analogue and digital data is that meaning is typically conveyed by the two dimensional spatial configuration of individual photons changing over time. Our retinas detect these relative variations - dark and light, or colors - and symbols are discerned. This is what was meant by “parallel channel”, Footnote 6. Digital information can be carried on a one-dimensional analogue channel: a stream of single photons. [/footnote].

Data must come into contact with the sensory organ[footnote]In biological organisms: eye, ear, tongue, skin, cell membrane; in synthetic informational organisms, antenna or conductive connection with the signal origin.[/footnote] of a perceiving entity in order to be sensed[footnote]A sleeping person is still an informational entity: it can detect a signal - say, a stick poking them - and awaken.[/footnote].

If sensed, by focusing its attention[footnote]Another key to all of this ...[/footnote] on the incoming data, the information contained can be interpreted by the informee[footnote]It appears that information only exists within the awareness of a perceiving entity. Physical data holds the *potential* for meaning (information), but it is only upon sensing and perception that this information comes into existence. This trait - that information only exists within the manifest conceptual space (i.e. the awareness) of informational entities - means that it is a necessarily subjective phenomenon.[/footnote].

Within the local conceptual space within the informational entity’s awareness, if its attention is placed upon the signal carrying the data into its organism[footnote]Here, focused on perception of external events. It seems likely that the conception of internal events must arise from the detection of internal data entering awareness. Generally the process of data entering a sentient entity’s awareness is termed “qualia”. In this author’s view, this is empirical observation in its strictest sense. Because the physical universe is perfectly unique, each qualia is unique, an instant in your lived experience where the universe pours into you. It is by conceptualizing these qualia that we ascribe meaning to our perceived reality.[/footnote], perception can occur.

Once perceived, information[footnote]A conceptual object.[/footnote] can be interpreted and acted upon.

Every object has a subject and every subject has an object[footnote]This applies to both physical and conceptual objects.[/footnote]. Without either the object would not exist.

Object classes only exist in the conceptual space; sameness only exists within the conceptual space[footnote]This observation is derived from the recognition that physical reality is entirely heterogeneous: every point in the universe, at every instant, is perfectly unique.[/footnote].

initial ideas of a theory of conceptual reality will be outlined[[84]](#footnote-83)

Bilbow, Angela. 2019. *High Court tackles Bitcoin ‘property’ first*. [Accessed at <https://www.cdr-news.com/categories/litigation/10003-high-court-tackles-bitcoin-property-first>]

Varian, H. R. 1998. *Markets for Information Goods*. University of California, Berkeley. [Accessed at <http://people.ischool.berkeley.edu/~hal/Papers/japan/>]

Vu, T. M. 2015. *Are the differences between Public Good and Common Pool Resource too blurred?* [Accessed at <https://www.researchgate.net/post/Are_the_differences_between_Public_Good_and_Common_Pool_Resource_too_blurred>]

Broader discussion of the governance of informational resources

The informational commons

“We can possess nothing—neither thing nor thought—absent a border between self and State. Each individual right is derived from this line, which we call privacy.” @Snowden, 24 August 2019. <https://twitter.com/Snowden/status/1165096076799619073>

## References

Core Mesa Team. 2019. *Mesa: Agent-based modeling in Python 3+*. [Accessed at <https://github.com/projectmesa/mesa>]

Wood 2019. Ethereum Yellow Paper.

Buterin 2014. Ethereum White Paper.

Ryan 2017. <https://hackernoon.com/ether-purchase-power-df40a38c5a2f>

Stackexchange.com 2019. <https://ethereum.stackexchange.com/questions/269/what-exactly-is-an-ethereum-client-and-what-clients-are-there>

Antonopolous 2018. Mastering Ethereum. <https://github.com/ethereumbook/ethereumbook>

Antonopoulos 2017. Mastering Bitcoin. <https://github.com/bitcoinbook/bitcoinbook>

BusinessDictionary 2019. <http://www.businessdictionary.com/definition/utility.html>

Gilbert, Nigel. 2008.  *Agent-based models.* Sage Publications.

Web3.js 2019. <https://web3js.readthedocs.io/en/v1.2.1/callbacks-promises-events.html>

Peaster 2018. <https://bitsonline.com/fcoin-gpm-listing-cloggage/>

BBC 2017. <https://www.bbc.com/news/technology-42237162>

McDonald 2017. <https://medium.com/@jgm.orinoco/releasing-stuck-ethereum-transactions-1390149f297d>

Sabanal 2016. <https://securityintelligence.com/thingbots-the-future-of-botnets-in-the-internet-of-things/>

Huang 2017. <https://medium.com/@Ashley_Huang/the-origins-of-computing-the-sumerians-the-abacus-binary-code-15289c3ced16>

CrashCourse 2017. <https://www.youtube.com/watch?v=O5nskjZ_GoI>

Babbage 1822. A Note Respecting the Application of Machinery to the Calculation of Astronomical Tables. <http://cyn.io/charles-babbage-a-note-respecting-the-application-of-machinery-to-the-calculation-of-astronomical-tables/>

Lovelace, A., 1843. ‘Notes on L. Menabrea’s ‘Sketch of the Analytical Engine Invented by Charles Babbage, Esq.’’’. *Taylor’s Scientific Memoirs*, *3*(1843), p.1843.

Engel 2019. <https://magenta.tensorflow.org/gansynth>

Toole, Betty A., 1998. *Ada, The Enchantress Of Numbers: Prophet Of The Computer Age*. Strawberry Press. Accessed at [<http://www.cs.yale.edu/homes/tap/Files/ada-lovelace-notes.html>]

Essinger 2018. <https://www.historyextra.com/period/modern/ada-lovelace-stem-women-computing-science-facts-life/>

Wynn-Williams 1931. The use of thyratrons for high speed automatic counting physical phenomena. <https://doi.org/10.1098/rspa.1931.0102>.

Stankovic 2008. Reprints from the Early Days of Information Sciences: On the Contributions of Akira Nakashima to Switching Theory.

Shannon, C.E., 1938. A symbolic analysis of relay and switching circuits. Trans. Am. Inst. Electr. Eng. 57, 713–723.<https://doi.org/10.1109/T-AIEE.1938.5057767>

Turing, A.M., 1937. On Computable Numbers, with an Application to the Entscheidungsproblem. Proceedings of the London Mathematical Society s2-42, 230–265.<https://doi.org/10.1112/plms/s2-42.1.230>

Mullins 2012. <https://www.cl.cam.ac.uk/projects/raspberrypi/tutorials/turing-machine/one.html>

Gleick 2011. The Information: A History, a Theory, a Flood

Fahie 1884. A History of Electric Telegraphy to the Year 1837.

Norman 2019. <http://historyofinformation.com/detail.php?entryid=519>

Burns, Russell. 2004. *Communications: An International History of the Formative Years.* The Institution of Electrical Engineers. [Accessed at <https://books.google.co.uk/books?id=7eUUy8-VvwoC&pg=PA29&redir_esc=y#v=onepage&q&f=false>]

Nair, B. Somanathan. 2002. *Digital Electronics and Logic Design*. Prentice Hall. [Accessed at <https://books.google.co.uk/books?id=WK45wLHL-ycC&pg=PA289&dq=%22digital+signals+are%22+%22digital+electronics%22&hl=sv&sa=X&redir_esc=y#v=onepage&q=%22digital%20signals%20are%22%20%22digital%20electronics%22&f=false>]

Dume 2012. <https://physicsworld.com/a/photon-shape-could-be-used-to-encode-quantum-information/>

Economist 2017. A clever way to transmit data on the cheap. <https://www.economist.com/science-and-technology/2017/09/16/a-clever-way-to-transmit-data-on-the-cheap>

Martin, Ernst. 1992. *The Calculating Machines: Their History and Development*. The MIT Press. [Accessed at [www.rechenmaschinen-illustrated.com/Martins\_book/Ernst%20Martin%20-%20Rechen%20Machinen%20OCR%204.pdf](http://www.rechenmaschinen-illustrated.com/Martins_book/Ernst%20Martin%20-%20Rechen%20Machinen%20OCR%204.pdf)]

CrashCourse 2017b. <https://www.youtube.com/watch?v=LN0ucKNX0hc&list=PLME-KWdxI8dcaHSzzRsNuOLXtM2Ep_C7a&index=4&t=189s>

Wikipedia 2019a. <https://en.wikipedia.org/wiki/Message>

Oxford 2019a. <https://www.lexico.com/en/definition/cipher>

Oxford 2019b. <https://www.lexico.com/en/definition/algorithm>

Oxford 2019c. <https://www.lexico.com/en/definition/protocol>

Techopedia 2019. <https://www.techopedia.com/definition/24749/cryptographic-key>

Phant0m 2013. <https://stackoverflow.com/questions/14526584/what-does-the-xor-operator-do>

Daemen, J., n.d. 1999. The Rijndael Block Cipher 47. [Accessed at csrc.nist.gov/archive/aes/rijndael/Rijndael-ammended.pdf#page=1]

National Institute of Standards and Technology. 2001. FIPS 197, Advanced Encryption Standard (AES), n.d. 51. [Accessed at nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.197.pdf]

Kerckhoffs, A. 1883. La Cryptographie Militaire (première partie) 37. [Accessed at <https://www.petitcolas.net/kerckhoffs/crypto_militaire_1_b.pdf>; translation at <https://www.petitcolas.net/kerckhoffs/index.html>]

Wikipedia 2019c. <https://en.wikipedia.org/wiki/Public-key_cryptography#History>

Wikipedia 2019d. <https://en.wikipedia.org/wiki/Checksum>

Wikipedia 2019e. <https://en.wikipedia.org/wiki/Transmission_Control_Protocol>

Wikipedia 2019f. <https://en.wikipedia.org/wiki/History_of_free_and_open-source_software>

Ker, Andrew D. 2014. *Computer Security*. Oxford University Department of Computer Science. [Available at ]

Movable-type.. 2019. <https://www.movable-type.co.uk/scripts/sha256.html>

Cerf, V.G., Kahn, R.E., 1974. A Protocol for Packet Network Intercommunication 13. [Accessed at https://www.cs.princeton.edu/courses/archive/fall06/cos561/papers/cerf74.pdf]

livinginternet.com. 2019. <https://www.livinginternet.com/i/ii_licklider.htm>

Licklider. 1968. *The Computer as a Communication Device*. Science and Technology. [Accessed at memex.org/licklider.pdf]

The Economist. 2019. Schumpeter: The Exxon-Valdez of Cyberspace. [Accessed at <https://www.economist.com/business/2019/08/08/the-exxon-valdez-of-cyberspace>]

Leiner 1997. Brief History of the Internet. Internet Society. [Accessed at <https://www.internetsociety.org/wp-content/uploads/2017/09/ISOC-History-of-the-Internet_1997.pdf>]

Kleinrock, Leonard. 1961. Information Flow in Large Communication Nets. Massachusetts Institute of Technology Research Laboratory of Electronics. [Accessed at <https://www.google.com/url?sa=t&rct=j&q=&esrc=s&source=web&cd=1&ved=2ahUKEwiE3LaF4JbkAhWoTxUIHTGDDrwQFjAAegQIAhAC&url=https%3A%2F%2Fwww.lk.cs.ucla.edu%2Fdata%2Ffiles%2FKleinrock%2FInformation%2520Flow%2520in%2520Large%2520Communication%2520Nets.pdf&usg=AOvVaw11sqIM4l6SD1VAaJbM3ODH>]

Berners-Lee, Tim. 1990. *WorldWideWeb: Proposal for a HyperText Project*. [Accessed at <https://www.w3.org/Proposal.html>]

Quinn 2008. <https://www.telegraph.co.uk/finance/financialcrisis/3248774/Greenspan-admits-mistakes-in-once-in-a-century-credit-tsunami.html>

Nakamoto 2008.

Nakamoto 2009. Bitcoin v0.1 released. <https://www.metzdowd.com/pipermail/cryptography/2009-January/014994.html>

Long 2019. <https://www.forbes.com/sites/caitlinlong/2019/06/29/bitcoin-the-dollar-and-facebooks-cryptocurrency-price-volatility-versus-systemic-volatility/#1aa3f14d88b8>

Vincent 2019. <https://www.theverge.com/2019/7/4/20682109/bitcoin-energy-consumption-annual-calculation-cambridge-index-cbeci-country-comparison>

Blockchain.com 2019. <https://www.blockchain.com/en/charts/difficulty?timespan=all>

Kim 2019. <https://www.coindesk.com/code-for-ethereums-proof-of-stake-blockchain-to-be-finalized-next-month>

Buterin 2016. *A Proof of Stake Design Philosophy*. <https://medium.com/@VitalikButerin/a-proof-of-stake-design-philosophy-506585978d51>

Szabo 1994. <http://www.fon.hum.uva.nl/rob/Courses/InformationInSpeech/CDROM/Literature/LOTwinterschool2006/szabo.best.vwh.net/smart.contracts.html>

@VitalikButerin. 2018. <https://twitter.com/VitalikButerin/status/1051160932699770882>

Wood 2014. <http://gavwood.com/dappsweb3.html>

Raymond 1999. The Cathedral and the Bazaar. <http://www.catb.org/~esr/writings/cathedral-bazaar/cathedral-bazaar/ar01s04.html>

Torvalds 2005. Initial revision of “git”, the information manager from hell. <https://github.com/git/git/commit/e83c5163316f89bfbde7d9ab23ca2e25604af290>

git-scm.com 2019. <https://git-scm.com/>

Taleb, N. N., 2013. *Antifragile: Things That Gain From Disorder*. Penguin.

Vyas, Tanvi. 2016. *No More Passwords over HTTP, Please!* Mozilla. [Accessed at <https://blog.mozilla.org/tanvi/2016/01/28/no-more-passwords-over-http-please/>]

Krebs, Brian. 2019. *Facebook Stored Hundreds of Millions of User Passwords in Plain Text for Years.* <https://krebsonsecurity.com/2019/03/facebook-stored-hundreds-of-millions-of-user-passwords-in-plain-text-for-years/>

Manadhata, P., Wing, J.M., 2004. *Measuring a System’s Attack Surface*. Defense Technical Information Center, Fort Belvoir, VA.<https://doi.org/10.21236/ADA458115>

McCoy 2015. <https://www.r1soft.com/blog/6-notorious-cases-of-data-loss-all-hosting-providers-can-learn-from>

Haselton 2017. <https://www.cnbc.com/2017/09/07/credit-reporting-firm-equifax-says-cybersecurity-incident-could-potentially-affect-143-million-us-consumers.html>

Hopkins 2017. <https://www.theguardian.com/business/2017/sep/25/deloitte-hit-by-cyber-attack-revealing-clients-secret-emails>

Fernandez 2019. <https://www.nytimes.com/2019/08/22/us/ransomware-attacks-hacking.html>

Prince 2019. <https://blog.cloudflare.com/terminating-service-for-8chan/>

Galperin 2015. <https://www.eff.org/deeplinks/2015/12/changes-facebooks-real-names-policy-still-dont-fix-problem>

Allen, Christopher. 2016. *The Path to Self-Sovereign Identity*. <http://www.lifewithalacrity.com/2016/04/the-path-to-self-soverereign-identity.html>

Malik 2008. <https://www.forbes.com/sites/nikitamalik/2018/11/07/the-tradeoff-between-security-and-privacy-how-do-terrorists-use-encryption/#368d68862d8c>

Safford 2003. <https://www.linuxjournal.com/article/6633>

Anderson, Ross. 2003. *‘Trusted Computing’ Frequently Asked Questions*. [Accessed at <https://www.cl.cam.ac.uk/~rja14/tcpa-faq.html>]

Fecke 2018. <https://legal-tech-blog.de/the-problem-of-blockchain-oracles-interview-with-alexander-egberts>

Chohan, Usman. 2017. *The Decentralized Autonomous Organization and Governance Issues*. [Available at SSRN:<https://ssrn.com/abstract=3082055>]

Bitvalley. 2018. *IBISA: Disrupting Agriculture Insurance*. Luxembourg. [Accessed at <https://www.ibisa.network/>]

Rea, A., Fischer, A., du Rose, J. 2019. [Colony Technical White Paper. Accessed at <https://colony.io/whitepaper.pdf>]

DAOstack. 2019. An Operating System for Collective Intelligence. [Accessed at <https://daostack.io/wp/DAOstack-White-Paper-en.pdf>]

Merritt, R. 2003. *New group aims to secure PCs, PDAs, cell phones*. [Accessed at <https://www.eetimes.com/document.asp?doc_id=1202119>]

Aragon. 2019. *Aragon Network*. [Accessed at <https://github.com/aragon/whitepaper>]

Gentry, C. 2010. Computing Arbitrary Functions of Encrypted Data.

Bevir, M. 2012. Governance: A Very Short Introduction. Oxford University Press.

Rouse, M. 2019. *Access control.* [Accessed at <https://searchsecurity.techtarget.com/definition/access-control>]

Ostrom, Elinor. 1990. *Governing the Commons: The Evolution of Institutions for Collective Action*. [Cambridge University Press](https://en.wikipedia.org/wiki/Cambridge_University_Press). [ISBN](https://en.wikipedia.org/wiki/International_Standard_Book_Number) [978-0-521-40599-7](https://en.wikipedia.org/wiki/Special:BookSources/978-0-521-40599-7).

Antonopoulos, A. 2017b. *Money as a System-of-Control*. [Accessed at <https://www.youtube.com/watch?v=FyK4P7ZdOK8&vl=en>]

Wilber, K., 2001. *A Brief History Of Everything*. Gill & Macmillan.

Buterin, V. 2014b. <https://forum.ethereum.org/discussion/comment/3389/#Comment_3389>

Lloyd’s. 2019. *Triggering innovation: How smart contracts bring policies to life*. Queen Mary Centre for Commercial Law Studies. [Accessed at <https://www.lloyds.com/news-and-risk-insight/risk-reports/library/technology/triggering-innovation>]

Glaser, A., Barak, B., Goldston, R. 2014. A zero-knowledge protocol for nuclear warhead verification. Nature. [Available at <https://www.nature.com/articles/nature13457>]

Becha, H. 2019. *Smart Containers: Real-time Smart Container data for supply chain excellence*. UNECE - UN / CEFACT. [Accessed at <https://www.unece.org/fileadmin/DAM/cefact/GuidanceMaterials/WhitePapers/WP-SmartContainers_Eng.pdf>]

Beal, V. 2019. *DDoS attack - Distributed Denial of Service*. [Accessed at <https://www.webopedia.com/TERM/D/DDoS_attack.html>]

Allen, C. et al. 2015. *Decentralized Public Key Infrastructure*. [Accessed at <https://github.com/WebOfTrustInfo/rwot1-sf/blob/master/draft-documents/Decentralized-Public-Key-Infrastructure-CURRENT.md>]

Nuñez, D. 2018. *Umbral: A Threshold Proxy Re-encryption Scheme*. NICS Lab, University of Malaga, Spain. [Accessed at <https://github.com/nucypher/umbral-doc/blob/master/umbral-doc.pdf>]

Benet, Juan. 2004. *IPFS - Content-Addressed, Versioned, P2P File System*. [Accessed at <https://github.com/ipfs/papers/raw/master/ipfs-cap2pfs/ipfs-p2p-file-system.pdf>]

Cohen, B. 2003. *Incentives build robustness in bittorrent*. In Workshop on Economics of Peer-to-Peer systems, volume 6, pages 68–72.

Merriam Webster. 2019. [Accessed at <https://www.merriam-webster.com/dictionary/synthetic>]

Ericsson. 2019. *Internet of Things Forecast*. [Accessed at <https://www.ericsson.com/en/mobility-report/internet-of-things-forecast>]

Shirer, M, MacGillivray, C. 2019. *The Growth in Connected IoT Devices Is Expected to Generate 79.4ZB of Data in 2025, According to a New IDC Forecast*. [Accessed at <https://www.idc.com/getdoc.jsp?containerId=prUS45213219>]

Chen, J. 2019. *Agency Problem.* Investopedia. [Accessed at <https://www.investopedia.com/terms/a/agencyproblem.asp>]

Ehrsam, F. 2017. *Blockchain Governance: Programming Out Future*. Medium. [Accessed at <https://medium.com/@FEhrsam/blockchain-governance-programming-our-future-c3bfe30f2d74>]

jnnk. 2015. *What is Gas Limit in Ethereum?* Stackexchange.com. [Accessed at <https://bitcoin.stackexchange.com/questions/39132/what-is-gas-limit-in-ethereum>]

Merriam-Webster. 2019j. *Precompetitive*. [Accessed at <https://www.merriam-webster.com/dictionary/precompetitive>]

Buterin, V. 2017. *The Meaning of Decentralization*. Medium. [Accessed at <https://medium.com/@VitalikButerin/the-meaning-of-decentralization-a0c92b76a274>]

Protocol Labs. 2019. *Content Identifiers (CIDs).* Accessed at <https://docs.ipfs.io/guides/concepts/cid/>]

Boyd-Rice, Jade. *New A.I. application can write its own code*. Futurity. [Accessed at <https://www.futurity.org/artificial-intelligence-bayou-coding-1740702/>]

@santisiri. 2019b. *ed snowden:.* [Accessed at <https://twitter.com/santisiri/status/1163779311221923840>]

Harris, M. 2019. *Pentagon testing mass surveillance balloons across the US*. The Guardian. [Accessed at <https://www.theguardian.com/us-news/2019/aug/02/pentagon-balloons-surveillance-midwest>]

Weigert, M. 2015. *The surveillance state is inevitable*. Meshed Society. [Accessed at <https://meshedsociety.com/the-surveillance-state-is-inevitable/>]

Ethereum Foundation. 2019. *Design Rationale*. Github. [Accessed at <https://github.com/ethereum/wiki/wiki/Design-Rationale>]

Gutierrez, Daniel. 2016. *Why Time-Value of Data Matters*. insideBIGDATA. [Accessed at <https://insidebigdata.com/2016/04/08/why-time-value-of-data-matters/>]

van Rossum, G. 1995. *Python tutorial, Technical Report CS-R9526.* Centrum voor Wiskunde en Informatica (CWI), Amsterdam..

Pérez, Fernando, Granger, Brian E. 2007. *IPython: A System for Interactive Scientific Computing*. Computing in Science and Engineering, vol. 9, no. 3, pp. 21-29, doi:10.1109/MCSE.2007.53. [Available at <https://ipython.org>]

Oliphant, Travis E. 2006. *A guide to NumPy.* USA: Trelgol Publishing. [Available at <https://www.numpy.org/>]

McKinney, Wes. 2010. *Data Structures for Statistical Computing in Python.* Proceedings of the 9th Python in Science Conference, 51-56.

Hunter, John D.2007. *Matplotlib: A 2D Graphics Environment.*, Computing in Science & Engineering, **9**, 90-95, DOI:10.1109/MCSE.2007.55

Seabold, Skipper, Josef Perktold. 2010. *Statsmodels: Econometric and statistical modeling with python*. Proceedings of the 9th Python in Science Conference.

1. As in, us humans. [↑](#footnote-ref-0)
2. Though not an end, as it will be difficult to maintain. [↑](#footnote-ref-1)
3. Synthetic (as in, produced artificially (Merriam Webster 2019)) informational entities [↑](#footnote-ref-2)
4. Addition, subtraction, multiplication and division. [↑](#footnote-ref-3)
5. It could be argued that Lovelace foresaw such cutting-edge computational techniques as the application of generative adversarial networks to music composition (Engel 2019): “Supposing, for instance, that the fundamental relations of pitched sounds in the science of harmony and of musical composition were susceptible of such expression and adaptations, the engine might compose elaborate and scientific pieces of music of any degree of complexity or extent.” -Ada Lovelace (Toole 1998, pp 694) [↑](#footnote-ref-4)
6. According to inventor Francis Ronalds, the system offered "a mode of conveying telegraphic intelligence with great rapidity, accuracy, and certainty, in all states of the atmosphere, either at night or in the day, and at small expense." Privacy was also greater, as the optical channels semaphores used were public; accessing the electrical signal carried on an electrical conduit would require additional specialized equipment and knowledge. Expense was reduced not least due to the reduction in the required line-of-sight infrastructure and personnel required to relay an optical message. [↑](#footnote-ref-5)
7. A one dimensional channel, as opposed to a parallel, two dimensional channel - see Discussion. [↑](#footnote-ref-6)
8. Just one of many factors contributing to this author’s sense of intellectual inadequacy ... [↑](#footnote-ref-7)
9. For clarity, this theory pertains to the measurement and communication of units of information: by Shannon’s definition, a message consists of a sequence of symbols - Shannon information (Floridi pp 38). “If base 2 is used, the resulting units may be called binary digits, or more briefly bits, a word suggested by J. W. Tukey” (Shannon 1948 pp 380). In the Discussion the limits of this theory of digital information will be explored, contingent upon on a definition of “message” broader than Shannon’s. This is not to discredit the brilliance and profound impact Shannon’s work has had on the world by applying a rigorous conceptual framework to the mathematical treatment of information. [↑](#footnote-ref-8)
10. A carrier wave of energy propagated through a medium. [↑](#footnote-ref-9)
11. Also, public. [↑](#footnote-ref-10)
12. i.e. it is a “compression function”. [↑](#footnote-ref-11)
13. And, ideally (for security purposes), a dramatically different hash - “diffusion”. [↑](#footnote-ref-12)
14. This is termed “pre-image resistance”. [↑](#footnote-ref-13)
15. Though not exactly: “note that the compression property implies that the function must be many-to-one, because the domain is infinite and the codomain finite, so infinitely many collisions exist” (Ker 2014 pp 48). In collision resistant hashing algorithms “it should be computationally infeasible to find any” collisions. [↑](#footnote-ref-14)
16. Often hashes are used as checksums to ensure that a file has downloaded in full and uncompromised (Wikipedia 2019d). [↑](#footnote-ref-15)
17. This phenomenon provides one of the best opportunities to understand the relationship between matter and meaning, in this author’s view. This will be cursorily explored in the Discussion. [↑](#footnote-ref-16)
18. While explanation of the mathematical processes specific to various key cipher algorithms is beyond both the scope of this paper and the intellectual capacity of this author, it should be noted that these algorithms rely heavily on the application of the XOR (“exclusive or”) operator. Because of this operation’s properties: that it is “commutative, associative and self-inverse” (phant0m 2013), the application of the operation to the ciphertext with the same key outputs the plaintext. To improve security, more sophisticated symmetric key ciphers have been developed, such as the Advanced Encryption Standard (Daeman 1999, NIST 2001). [↑](#footnote-ref-17)
19. which is always assumed, according to Kerckhoffs’s Principle, concisely stated by Shannon as “the enemy knows the system” (Kerckhoffs 1883, Shannon 1949) [↑](#footnote-ref-18)
20. For example, key generation in the RSA cryptosystem, directly from Ker (2014 pp 74):

    (1) Choose two prime numbers each b/2 bits in size, call them p and q. Compute their b-bit product n = pq and the so-called totient φ = (p − 1)(q − 1).  
    (2) Choose an integer e which is coprime to φ.  
    (3) Publish the public key, which is the pair pk B = ⟨n, e⟩.

    (4) Find an integer d such that ed ≡ 1 (mod φ).  
    (5) The private key is the pair skB = ⟨n,d⟩2. [↑](#footnote-ref-19)
21. This is, of course, a significant simplification of the complexity of these cryptosystems, but hopefully captures the relevant features of these algorithms for the purposes of this paper. [↑](#footnote-ref-20)
22. A big “if”, and a primary, and valid, critique of Web3 technologies. If a private key is lost, in these systems it is objectively impossible to retrieve the assets encrypted or controlled by it. [↑](#footnote-ref-21)
23. This is an abstraction. [↑](#footnote-ref-22)
24. via Message Authentication Codes, MACs. [↑](#footnote-ref-23)
25. via digital signatures and keyed hashes. [↑](#footnote-ref-24)
26. binary messages [↑](#footnote-ref-25)
27. i.e. Servers and computing hardware; energy, water, real estate; development talent. [↑](#footnote-ref-26)
28. “A conflict of interest inherent in any relationship where one party is expected to act in another's best interests” (Chen 2019). [↑](#footnote-ref-27)
29. Really, the readers of metzdowd.com, an obscure cryptography message board. [↑](#footnote-ref-28)
30. Unfortunately, a thorough explanation of the evolution of the Bitcoin protocol and the functioning of Bitcoin clients is beyond the scope of this paper. [↑](#footnote-ref-29)
31. 1. A store of value. 2. A medium of exchange. 3. A unit of account. (Antonopoulos 2017b) [↑](#footnote-ref-30)
32. “Bitcoin can be thought of as a state transition system, where there is a ‘state’ consisting of the ownership status of all existing bitcoins and a ‘state transition function’ that takes a state and a transaction and outputs a new state” (Buterin 2013) [↑](#footnote-ref-31)
33. The bitcoin enters their wallet either by being sent there, or as a result of that wallet owner mining a new block, at which point they add a number of bitcoin to their wallet as a reward. [↑](#footnote-ref-32)
34. See *Mastering Bitcoin* Chapter 4 and Chapter 5 (Antonopoulos 2017) for a thorough accounting of Bitcoin key and wallet generation. These keys were generated using the bitcoin python library; the private key is the SHA256 hash of the string “ucl” in utf-8 encoding, 01110101 01100011 01101100 (see Table x). [↑](#footnote-ref-33)
35. A bitcoin address is “derived from the public key through the use of one-way cryptographic hashing”; it is Base58Check encoded for readability and error protection (Antonopoulos 2017 Ch 4). [↑](#footnote-ref-34)
36. Required to be smaller than the block size of 1 megabyte, the “block size”. [↑](#footnote-ref-35)
37. This inclusion of the prior block hash, due to the properties of cryptographic hashing algorithms described, means that once a block is mined changing it would entail re-mining all subsequent blocks, including the sequence of new block hashes. This is the basis of the term “blockchain”, which is, in the strictest sense, simply a database in which the current state is cryptographically linked to the prior state, providing a guarantee of immutability. The term has come to be understood, however, as the combination of “the state transition system with a consensus system in order to ensure that everyone agrees on the order of transactions” (Buterin 2013). [↑](#footnote-ref-36)
38. The inclusion of this Merkle root hash further strengthens the security of the system. If a dishonest node changes a single bit (say, giving itself more bitcoin), the deviation is detected by the rest of the network, and rejected. [↑](#footnote-ref-37)
39. i.e. has a number of leading 0s - formalized as the “block difficulty”. By adjusting this block difficulty the network can raise or reduce the difficulty of finding a nonce yielding an output hash, thereby self-adjusting to the amount of computing power operating on the network. The hash rate of the Bitcoin network is a measure of its security (Long 2019) meaning, sadly, that the security of the network - a strength - is proportionate to its energy consumption - a substantial drawback (Vincent 2019). [↑](#footnote-ref-38)
40. This so-called Proof of Work consensus mechanism serves to effectively randomize which validator node wins the right to update the database state, and puts in place a substantial computational cost to change prior states. Solution nonces for all subsequent blocks would need to be computed, as changing the prior data would most likely result in the hash of subsequent block headers not fulfilling the block difficulty requirement to be accepted as valid, according to the protocol. [↑](#footnote-ref-39)
41. According to the protocol. Block solutions broadcast to the network that violate the protocol will be rejected by the rest of the network. [↑](#footnote-ref-40)
42. Bitcoin is inherently deflationary. [↑](#footnote-ref-41)
43. Smart contracts - “computerized transaction protocol[s] that execute the terms of a contract” - were first conceptualized by Nick Szabo (1994), who envisioned sophisticated systems of ownership and custodianship subject to conditions implementable in code. Buterin has since expressed “regret [at] adopting the term ‘smart contracts’. I should have called them something more boring and technical, perhaps something like ‘persistent scripts’.” (@VitalikButerin 2018). [↑](#footnote-ref-42)
44. Hyperbolic? Only time will tell ... [↑](#footnote-ref-43)
45. For example, the establishment of a (necessarily) open protocol - TCP/IP - by Cerf and Kahn (1974) decentralized the ability to develop software compliant with that protocol. This sort of precompetitive (Merriam Webster 2019j) coordination is of deep interest to this author. [↑](#footnote-ref-44)
46. Initially - though an alternative consensus mechanism, Proof of Stake, is being developed for Ethereum 2.0 (Kim 2019). [↑](#footnote-ref-45)
47. Externally-owned account, sometimes EOA. [↑](#footnote-ref-46)
48. Deploying a contract requires a special “contract creation transaction” to be sent to the zero address (0x0), with compiled bytecode included (Antonopoulos 2018 Ch 7). [↑](#footnote-ref-47)
49. A second type of address in the Ethereum system, contract addresses reference the contracts deployed there. They do not have a private key - it “in fact does not exist - we can say that smart contract accounts own themselves” (Antonopoulos 2018 Ch 7) [↑](#footnote-ref-48)
50. A note fundamental to the premise of this dissertation: the system is agnostic to the data being passed into these functions or written to the chain; so long as the data type is compliant, it will be accepted. It makes no judgments about the veracity of the information the data represents. Within the blockchain vocabulary, external entities that provide data to a blockchain are called “oracles”; these entities represent a potential weakness in the system (Fecke 2018). [↑](#footnote-ref-49)
51. As specified by the transaction originator. [↑](#footnote-ref-50)
52. To enable the market to determine the value of performing operations on the Ethereum Virtual Machine, and to act “as a buffer between the (volatile) price of Ethereum [sic] and the reward to miners for the work they do”, account holder submitting transactions specifies the price (in a subunit of ether - “wei”) they are willing to pay per unit of gas (Antonopoulos 2018 Ch 13). This informs miners, who receive transaction fees of validated blocks, which transactions will provide them the highest rewards. [↑](#footnote-ref-51)
53. Externally-owned accounts must have some ether in them for any transactions submitted to be validated successfully. [↑](#footnote-ref-52)
54. And the reward for participating in the validation process. [↑](#footnote-ref-53)
55. Linus’s Law: ”Given enough eyeballs, all bugs are shallow.” (Raymond 1999) [↑](#footnote-ref-54)
56. An example demonstrating the benefit of this diversity is the development of multiple implementations of a blockchain protocol - In the Ethereum ecosystem, geth, parity, pyeth, cpp-ethereum, etc. (Antonopoulos 2018 Ch 3) - which reduces the impact of a common mode failure (Buterin 2017c) in any one of those implementations, enabling a network to respond to such a failure without going offline. (Stackexchange.com 2019). [↑](#footnote-ref-55)
57. Of course, open source code can be examined and exploited more easily by malicious entities. It seems that on balance open source code is more secure, but counter-examples surely exist; some systems are more secure if source code remains private. [↑](#footnote-ref-56)
58. Albeit on an encrypted channel like HTTPS - sometimes (Vyas 2016) [↑](#footnote-ref-57)
59. Hopefully stored securely … though sometimes, even in the case of highly skilled and resourced firms, not (Krebs 2019). [↑](#footnote-ref-58)
60. as in the case of DreamHost and several government agencies ... [↑](#footnote-ref-59)
61. … JPMorgan Chase, Equifax ... [↑](#footnote-ref-60)
62. … Facebook, and Cloudflare, to name a few. [↑](#footnote-ref-61)
63. While a topic of intense fascination, an in-depth discussion of the ethics of the denial of access to informational services is beyond the scope of this dissertation. Web3 purists seem to err toward a complete rejection of the morality of authorities deciding to prevent behavior they deem harmful. This author takes a more nuanced view, but overall would rather see those (dis)incentivization mechanisms (such as censorship, sanctions and so on) built atop a system that is fundamentally oriented toward individual liberties. [↑](#footnote-ref-62)
64. i.e. smart contract source code. [↑](#footnote-ref-63)
65. Of course, some organizations require privacy for ethical, legal or business reasons. Due to its constraints, this paper cannot thoroughly explore the mechanisms and ethics of organizational privacy. [↑](#footnote-ref-64)
66. A common use case would be the collection of funds and collective decision-making about their disbursement, based on some voting mechanism (DAOstack 2019). [↑](#footnote-ref-65)
67. For example, IBISA is a project building a decentralized autonomous organization offering crop insurance to smallholder farmers in developing countries. Their customers are unserved by traditional insurers due to the high costs of offering products relative to the low revenue opportunity the customers represent (Bitvalley 2018). [↑](#footnote-ref-66)
68. Commonly experienced by this author. [↑](#footnote-ref-67)
69. In the spirit of removing single points of failure, and improving performance. [↑](#footnote-ref-68)
70. BitTorrent (Cohen 2003) forms some of the basis of Benet’s solution. [↑](#footnote-ref-69)
71. In IPFS, v0 Content Identifiers are 46 bytes in length (Protocol Labs 2019) [↑](#footnote-ref-70)
72. Desktops, laptops, tablets and smartphones, generally. [↑](#footnote-ref-71)
73. To this author’s knowledge no computer exists that is controlled entirely by software that was developed and deployed by another computer, though it is not unfathomable (Boyd-Rice 2018). It seems obvious, however, that all future software and hardware technology ultimately originated in a human action. [↑](#footnote-ref-72)
74. As well as the intellectual capacity of this author. [↑](#footnote-ref-73)
75. If the device securely maintains custody of the private key (i.e. it has the only instance of that private key in existence), all digital signatures with that key must have been performed on the device. [↑](#footnote-ref-74)
76. Should they? [↑](#footnote-ref-75)
77. Unfortunately beyond this paper’s scope, the combination of trusted computing modules with homomorphic cryptography techniques, which “[allow] computation on ciphertexts, generating an encrypted result which, when decrypted, matches the result of the operations as if they had been performed on the plaintext” (Wikipedia 2019h, Gentry 2010). Security and privacy are often viewed as a trade-off (Malik 2018): increased surveillance erodes privacy but increases the likelihood of authorities detecting threats to public safety. Acknowledging the constraints entailed by the computational intensity of homomorphic cryptography, this author is still wondering: could homomorphic algorithms analyze data about some entity to detect a threat, and only reveal the contents of the data to authorities if a threat is detected? Could software agents analyzing data about human activity be configured as a sort of benevolent panopticon - an artificial intelligence oriented toward minimizing harm and damage to life and property? [↑](#footnote-ref-76)
78. In the Ethereum Yellow Paper, Wood (2014) recognized the potential of machines providing inputs to smart contracts: “A transaction (formally, T) is a single cryptographically-signed instruction constructed by an actor externally to the scope of Ethereum. While it is assumed that the ultimate external actor will be human in nature, software tools will be used in its construction and dissemination. [Footnote 1] Notably, such ‘tools’ could ultimately become so causally removed from their human-based initiation—or humans may become so causally-neutral—that there could be a point at which they rightly be considered autonomous agents” (Wood 2019 pp 4). [↑](#footnote-ref-77)
79. “‘We do not think that American cities should be subject to wide-area surveillance in which every vehicle could be tracked wherever they go,’ said Jay Stanley, a senior policy analyst at the American Civil Liberties Union.” Leaving aside the apparent inevitability of this, this author believes that such a capability is not intrinsically unethical. If such information is stored, analyzed and accessed by an open source system designed to respect the individual’s privacy, with appropriate anonymization, it is believed that such a system has the potential to fundamentally improve physical security in the service of the public interest. The key phrase: “designed to respect the individual’s privacy”; likely incorporating zero-knowledge and range proofs, as well as homomorphic cryptographic techniques, etc. If “the surveillance state is inevitable” (Weigert 2015), then the nature of the state must be adapted so human rights are respected. [↑](#footnote-ref-78)
80. Government, as in “the system or group of people governing an organized community” (Wikipedia 2019h). Note that this conception of government is much broader than the common state-based understanding of the term; it is “a means by which organizational policies are enforced, as well as a mechanism for determining policy” (Wikipedia 2019h). [↑](#footnote-ref-79)
81. This author is critical of the general economic definition of “non-rivalrous”, which forms the basis of the distinction between public goods and common goods (Vu 2015). The difference is appreciated: the use of a rivalrous good decreases the value of the asset, thereby making that value unavailable to future users (i.e. when a tree is cut down and lumber removed no future person can use that specific tree for lumber), whereas the utility of non-rivalrous goods is not diminished with use (i.e. a waterway can be used to transport a shipment without reducing the waterway’s utility for future users). The point missed within the definitions reviewed is the temporal rivalry of many public goods: *while I am using this waterway*, availability to others is reduced. Informational resources, especially in the decentralized, content-addressed web, may (almost) be the first non-excludable resources in history (as even high-bandwidth servers have a limit to the number of users who can access an informational resource in a given time). True non-excludability may be impossible - except perhaps in the case of an informational asset in the custody of its user. [↑](#footnote-ref-80)
82. “can data be owned?” (@santisiri 2019) [↑](#footnote-ref-81)
83. Methods were included in the `Blockchain` subclass to simulate on-chain compute operations, namely, gas costs. [↑](#footnote-ref-82)
84. It is acknowledged that this has been an area of active enquiry by philosophical thinkers since the origins of thought. This author does not, however, have a formal philosophical education, and is therefore hesitant to cite philosophical works he is not qualified to reference. While certainly influenced by these intellectual luminaries - philosophical thinking informs the paradigms by which we interpret reality - these theories are primarily derived from observation of reality rather than exposure to any philosophical thinker. That said, it should be stated that the author has been influenced by the thinking of the integral philosopher Ken Wilber (Wilber 2001), especially inspired by his efforts to make room for the validity of subjective reality - something that is apparently difficult for unimaginative adherents to the materialist paradigm. [↑](#footnote-ref-83)