Smart Contracts on Algorand

Authors[1]
Archie Chaudhury and Brian Haney

Abstract

This Paper makes three main contributions. First, this Paper surveys Algorand Smart Contracts and the Algorand Network, including software systems and algorithmic architectures. Second, this Paper discusses various software mechanisms enabling developers to execute transfers on the Algorand Network. Third, this Paper advances Algorand Smart Contracts by introducing the Algogeneous Smart Contract. Algogeneous Smart Contracts are a new type of Algorand Smart Contract, which are simpler to develop and utilize artificial intelligence to ensure contracts are legally compliant and enforceable.

[1] Authors contributed equally to this work. We are thankful to the Algorand Foundation, Sean Lee, Addie Wagenknecht, Ryan Fox, Doug Broughton, Joel Carben, Autumn Moss Penaloza, Liz Baran, Fabrice Benhamouda, Arry Yu, Eric Brunngraber, and Dan Fitzgerald.

Table of Contents

Introduction *3*

I. Smart Contracts *4*

A. Origins **4**

B. Single State Contracts **5**

C. Algogeneous Contracts **6**

II. Network Application *9*

A. Software **9**

B. Choice Coin **11**

C. Security **12**

Conclusion *16*

Introduction

Fundamentally, contracts have three components. First, the parties must intend to exchange something of value. Second, there must be a meeting of the minds as to the exchange of value.[2] Third, there must be a physical representation for the value exchange.[3] Contract laws and theories have evolved through the ages[4] and still a complete theory remains evasive.[5]

The law has its limits. As Justice Oliver Wendell Holmes wrote, it is a fallacy to think "the only force at work in the development of the law is logic."[6] Consider not much has changed about contract law in over a Century since Justice Holmes sat on the Court. Of course, at least some things have change since the late 19th Century. Perhaps most significantly, digital systems of communication and information transfer[7] have spawned a new global economy.

This Paper addresses the Decentralized Asset Transfer Problem (DATP). The DATP refers to the challenges in successfully transferring assets between and among both centralized and decentralized financial systems. The central purpose for this Paper is introducing Algogeneous Smart Contracts, a new type of smart contact that is inherently simple while being able to perform complicated tasks. Algogeneous Smart Contracts are built on the Algorand Blockchain, using multi-party transactions to drive on-chain applications. Coupling stateful and stateless smart contracts, the Algogeneous framework combines simple smart contract logic, all the while infusing intelligent analysis to create software that is efficient, validated, and secure.

This Paper provides an overview of existing Algorand Smart Contracts and introduces Algogeneous Smart Contracts, a solution to the DAPT. Part I details the Algorand Network, including software synthesis, asset analysis, and security scrutiny. Part II introduces Algogeneous Smart Contracts with reference to both stateless smart contracts and stateful smart contracts. Moreover, Part II introduces novel artificial intelligence technology for processing smart contracts as a keystone to the Algogeneous architecture.

---

[2] Restatement (Second) of Contracts § 17 (1979).
[3] Hugh E. Willis, Restatement of the Law of Contracts of the American Law Institute, 7 Ind. L. J. 429, 430 (1932). ("The legal relation which exists in a contract is a right-duty relation, and a contract, therefore, should be defined either as a right in *personam* or as a legal obligation.")
[4] S.J. Stoljar, A History of Contract at Common Law, 4 (1975). ("It is first mentioned by Bracton but not yet by Glanvill, which makes the year 1201, the date of its earliest recorded instance, an accurate enough indication of its time of origin.") *See also* Morton J. Horwitz, The Historical Foundations of Modern Contract Law, 87 Harv. L. Rev. 917, 917 (1974). ("Beginning with the first English treatise on contract, Powell's Essay Upon the Law of Contracts and Agreements (1790), a major feature of contract writing has been its denunciation of equitable conceptions of substantive justice as undermining the "rule of law."")
[5] Alan Schwartz and Robert E. Scott, Contract Theory and the Limits of Contract Law, 113 Yale Law Journal 1, 2 (2003). ("Contract law has neither a complete descriptive theory, explaining what the law is, nor a complete normative theory, explaining what the law should be.")
[6] Oliver Wendell Holmes, Jr., *The Path of the Law*, 10 HARV. L. REV. 457, 465 (1897).
[7] C.E. Shannon, A Mathematical Theory of Communication, Bell Systems Technical Journal (1948). In fact, even in the mid 19th century, the roots of probability theory which lay the foundation for modern mechanistic models of computation. *See also* P.L. Chebyshev, Démonstration élémentaire d'une proposition Générale de la théorie des probabilities, 33 J. Reine Angew. Math. 259 (1846).

I. Smart Contracts

Now, code is law[8] and computable contracts are a better way to do business, removing the inherent attenuation between syntax and semantics in legal analysis. A smart contract is a computer program which automatically executes, transferring cryptocurrency.[9] Computationally, smart contracts are programs that are logically executed on a blockchain without a central oversight.[10] Algorand Smart Contracts facilitate global payment systems and financial transactions, with instantaneous processing and *de minimis* fees.

A. Origins

The traditional conception of contracts is a legal obligation between parties.[11] Smart contracts build upon this initial conception; a smart contract is a computer program which automatically executes, moving cryptocurrency or some other digital asset. They allow for automation and law to be written in transactional programs. Specifically, smart contracts on the Algorand Blockchain avoid the high fees and computational costs associated with smart contracts developed on other blockchains.

For example, the central innovation for the Ethereum[12] network is a software stack for smart contracts. The Ethereum software is implemented in the new programming language, Solidity – which was constructed with influence from C++, JavaScript, and Python.[13] Ethereum created smart contracts as a way to improve the transaction protocol on the Bitcoin Network.[14] A platform for applications development and financial transactions, Ethereum is the second largest blockchain in the world. Indeed, Ethereum is home to more than 400,000 smart contracts for novel tokens assets within its ecosystem.

---

[8] Lawrence Lessig, Code is Law (January 1, 2000). ("This regulator is code--the software and hardware that make cyberspace as it is. This code, or architecture, sets the terms on which life in cyberspace is experienced.")

[9] Fabrice Benhamouda, et al., Supporting Private Data on Hyperledger Fabric with Secure Multiparty Computation, IBM Journal of Research and Development (April 2019), DOI: 10.1147/JRD.2019.2913621. ("Nearly all blockchain architectures support the notion of smart contracts, namely a programmable application logic that is invoked for every transaction.")

[10] Massimo Bartoletti, A formal model of Algorand smart contracts, 1 (2021), https://arxiv.org/abs/2009.12140v3. ("Smart contracts are agreements between two or more parties that are automatically enforced without trusted intermediaries.")

[11] Hugh E. Willis, Restatement of the Law of Contracts of the American Law Institute, 7 Ind. L. J. 429, 430 (1932). ("A contract is the legal obligation created by the law as the result of a promise or set of promises for the breach of which the law gives a remedy, or the performance of which the law in some way recognizes as a duty.")

[12] Vitalik Buterin, Ethereum Whitepaper (2013). Created in the year 2013, Ethereum is a blockchain, which uses its own cryptocurrency to reward miners.

[13] Ethereum, Solidity Documentation (March 20, 2021).

[14] Ananda Badari and Archie Chaudhury, An Overview of Bitcoin and Ethereum White-Papers, Forks, and Prices, SSRN Paper No. 3841827 (2021). ("Ethereum's primary focus is to provide a protocol for building decentralized applications (dApps) deployed on the Ethereum Virtual Machine. Ethereum proposes a different protocol than Bitcoin in which the above limitations are addressed. Just to be clear, this is a huge difference to Bitcoin.") *See also* Satoshi Nakamoto, Bitcoin: A Peer-to-Peer Electronic Cash System 2 (2008). ("We need a way for the payee to know that the previous owners did not sign any earlier transactions. For our purposes, the earliest transaction is the one that counts, so we don't care about later attempts to double-spend.")

However, Solidity contracts come at the expense of expensive transaction fees. Moreover, in certain circumstances, Solidity contracts may take days to execute a transfer. And, the negative environmental effects of mining are also a disadvantage of Ethereum.[15] As such, Algorand constructed a new blockchain, solving several problems associated with Ethereum, offering more efficient and environmentally friendly smart contracts using proof-of-stake technology.[16]

B. Single State Contracts

Algorand Smart Contracts (ASCs) are programs serving various functions on the Algorand Blockchain. Specifically, ASCs are software systems that can be used for transactions and applications development.[17] The cryptographic architectures making ASCs possible include several functions and methods encrypted within the Algorand Network. Generally, ASCs are separated into two main categories, Stateful Smart Contracts and Stateless Smart Contracts.

Stateless Smart Contracts govern the transaction of funds between two parties. In other words, Stateless Smart Contracts are essentially escrow functions. An escrow is a contractual arrangement in which a third party receives and disburses money or property for transacting parties. Usually, contractual performance depends on conditions agreed to by the parties.

As such, Stateless Smart Contracts validate transactions between two parties, replacing traditional escrow accounts. Another way, Stateless Smart Contracts are designed to approve or deny transactions.[18] On the Algorand Network, Stateless Smart Contracts also act as signature delegators, signing transactions, thus validating them on the main blockchain network. For example, a Stateless Smart Contracts may validate a transaction between the user and the asset manager, ensuring that the transaction is signed with a cryptographic private key.

Stateful Smart Contracts are the Algorand Network's backbone, controlling the logic for value volatility. The term stateful refers to the contract's ability to store information in a specific state on the network. Stateful Smart Contracts are contracts that live on the chain and are used to store data. For example, one type of Stateful Smart Contract is an opt-in contract, allowing the user to elect to receive certain assets. The stateful opt-in contract stores data on the Network, associating

[15] Liana Badea, The Environmental and Economic Impact of Bitcoin (March 2021), DOI:10.1109/ACCESS.2021.3068636. *See also* Lucas Girard, Environmental Impacts of Cryptocurrency Mining (2018). *See also* Heidi Samford. Lovely-Frances Domingo, The Political Geography and Environmental Impacts of Cryptocurrency Mining (July 10, 2019).

[16] Ethereum, as well as Bitcoin are proof-of-work blockchains, which require miners to compete for rewards by producing massive amounts of computing power. Proof-of-stake blockchains are more efficient, with blocks validated using cryptographic proofs and rewards being distributed to asset holders. *See* Satoshi Nakamoto, Bitcoin: A Peer-to-Peer Electronic Cash System 3 (2008). ("The proof-of-work involves scanning for a value that when hashed, such as with SHA-256, the
hash begins with a number of zero bits.")

[17] Massimo Bartoletti, A formal model of Algorand smart contracts 1, arXiv:2009.12140 (2021). ("Smart contracts are agreements between two or more parties that are automatically enforced without trusted intermediaries.")

[18] Silvio Micali, Efficient Smart Contracts at Scale: Algorand's Stateful Teal Contracts 1 (2020), https://www.algorand.com/resources/blog/stateful-smart-contracts-teal.

the receiving account and the specified asset.[19] On the Algorand Network all assets are tagged with an Asset ID, which is a number corresponding to the asset on the blockchain.

Stateful Smart Contracts are called through transactions processed by the blockchain according to certain connectivity features. Moreover, Stateful Smart Contracts can be combined with other features to produce even more complex application types, may also be used for data storage, both global or local, and functional processing on the Algorand Blockchain. For example, a Stateful Smart Contract may be used as a voting method, storing data globally based on the result of several votes. However, Stateful Smart Contracts and Stateless Smart Contracts may both be adapted according to certain logical rules.[20]

C. Algogeneous Contracts

Algogeneous Smart Contracts are the next generation in both transactional and computable contracts on the Algorand Network. Start by assuming Stateful and Stateless are arbitrary terms, there are *n* number of ways to logically polarize smart contracts on Algorand.[21] But, Stateful and Stateless are still helpful to the extent they help to explain smart contract functionality.

Generally, Stateful Smart Contracts are not smart contracts, rather they are logical programs which store data on the blockchain. There is nothing contractual about their logical nature. Stateless Smart Contracts differ in that they validate transactions between parties, like an escrow and more like a contract in the traditional sense. However, both Stateless and Stateful Smart Contracts lack the ability to be written as a single script executable for digital asset transactions. Thus, Algogeneous Smart Contracts represent a technical convergence of Stateless and Stateful Smart Contracts and include an innovative integration with artificial intelligence.

Algogeneous Smart Contracts allow multiple tasks to be efficiently integrated within one function, all on the Algorand Blockchain. A smart contract by itself is a payment function, including functionality that both stores information and validates a transaction. Algorand's smart contracts can be linked through a reference pattern in which one smart contract's output can be dependent on another smart contract's logic. In short, an Algogeneous Smart Contract is a smart contract that manages to achieve the functionality of both a stateless and stateful smart contract in a singular system, with added intelligent validation and verification features.

Algogeneous Smart Contracts are structured in a way that enables stateless and stateful smart contracts to be bound to create applications that can handle more complex tasks with a simpler architecture. Figure 1 illustrates the relative convergence and innovation for Algogeneous Smart Contracts.

---

[19] Silvio Micali, Efficient Smart Contracts at Scale: Algorand's Stateful Teal Contracts 2 (2020), https://www.algorand.com/resources/blog/stateful-smart-contracts-teal. ("Furthermore, data, once written on the blockchain, cannot be changed, but efficiently updating state information requires overwriting.")

[20] Silvio Micali, Efficient Smart Contracts at Scale: Algorand's Stateful Teal Contracts, 3 (2020), https://www.algorand.com/resources/blog/stateful-smart-contracts-teal. ("In this approach, changing a piece of data from, say, *x* to *y*, requires presenting a proof of the value of x, relative to its previous commitment, and generating and storing a new commitment that reflects the replacement of *x* with *y*.")

[21] Massimo Bartoletti, A formal model of Algorand smart contracts, arXiv:2009.12140 (2021).

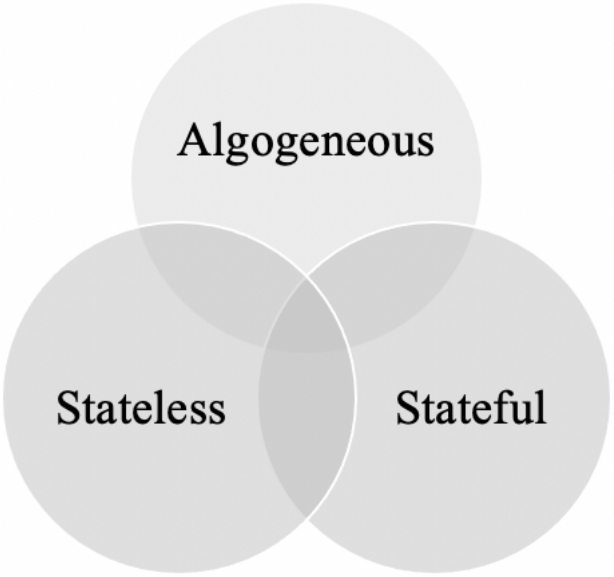

Figure 1

This interoperability enables Decentralized Applications (DApps) built on Algorand's blockchain to not only specifically validate transactions and other important data, but to also add an additional layer of complexity and security. Despite being a relatively simple verification on the surface, there are additional cryptographic protocols, ensuring validity and security within the Algorand Network, such as hash functions and private key pairs. Additionally, Algogeneous Smart Contracts incorporate a verification system in which individual parties and parts of the transaction are checked to ensure that data can be properly sanitized and stored on the Algorand Blockchain.

Where previous ASCs must be stateful or stateless, Algogeneous contracts may be stateful, stateless, or both.

(1) $$S_C = 0 \oplus 1$$

(2) $$H_C = 0 \otimes 1$$

Equation (1) defines a stateless smart contract, which may be a Boolean. Equation (2) defines a Algogeneous Smart Contract, which instead operates with inclusive OR function.

(3) $$H_C \rightarrow A_N$$

Equation (3) defines the transition function for the Algogeneous Smart Contract to the Algorand Network.

$$H_C \rightarrow A_N = \prod_{i=0} F_i^n \tag{4}$$

$$F_i = [0, \dots, n] \tag{5}$$

Equation (4) defines a value model for software quality optimization. Equation (5) defines the factors by which quality is evaluated.

While traditionally ASCs are separated into two main categories, stateful and stateless, the Algogeneous Smart Contract offers a third and interoperable ASC. Stateless smart contracts are primarily used for signature authorities and stateful smart contracts are used for data storage and functional processing on the Algorand blockchain. However, Algogeneous Smart Contracts may be used for signature validation, data storage, and functional processing on the Algorand blockchain.

The term artificial intelligence (AI) has been discussed at length by various scholars and industry leaders. Generally, AI refers to any machine capable of learning, remembering, and taking actions.[22] For example, in the legal industry, technology assisted review is changing the discovery process. In other words, AI programs now complete tasks previously only lawyers could do, like classify documents based on relevancy during discovery.[23]

The Algogeneous contract utilizes an embedded intelligence, a type of AI in for contract analysis. The AI checks to ensure the technical smart contract is legally valid according to law.

$$F_i^{W_i} = [f^{w_1} \cdots f^{w_n}] \tag{6}$$

$$ai = \sqrt[\Sigma_{j=1}^{n} w_j]{\prod_{i=1}^{n} F_i^{W_i}} \tag{7}$$

[22] Jeanne C. Fromer, *Machines as the New Oompa-Loompas: Trade Secrecy, the Cloud, Machine Learning, and Automation*, N.Y.U. L.R., 706, 720 (2019), https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3359746. ("In recent years, these techniques have been among the most successful and prominent ways of imbuing computers with artificial intelligence, or human-like cognitive abilities.") *See also* Maria Schuld, et al., The quest for a Quantum Neural Network, 3 (2014), https://arxiv.org/abs/1408.7005. ("Computing in artificial neural networks is derived from our neuroscientific understanding of how the brain processes information in order to master its impressive tasks.") *See also* RAY KURZWEIL, THE AGE OF INTELLIGENT MACHINES 14 (1992). (Artificial Intelligence is "the art of creating machines that perform functions that require intelligence when performed by people.") *See also* NILS J. NILSSON, ARTIFICIAL INTELLIGENCE: A NEW SYNTHESIS 1 (1998). (Artificial Intelligence is "concerned with intelligent behavior in artifacts.")

[23] Brian S. Haney, Applied Natural Language Processing for Law Practice, 2020 B.C. Intell. Prop. & Tech. F. 25 (2020). *See also* Chris D. Birkel, The Growth and Importance of Outsourced E-Discovery: Implications for Big Law and Legal Education, 38 J. LEGAL PROF. 231 (2014). *See also* Michael A. Livermore et al., Computationally Assisted Regulatory Participation, 93 NOTRE DAME L. REV. 977, 1006 (2018).

Equation (6) defines an array of factors and Equation (7), the AI Equation, defines a weighted average processing the array according to instructions from an embedded agent. The embedded agent formalizes knowledge for contractual analysis – assuring the contract is logically, legally, and transactionally valid.

$$(8) \qquad if \begin{cases} 1 - ai = 0; return = true \\ else; return = false \end{cases}$$

$$(9) \qquad s = \begin{cases} f_i \to 0 \otimes 1 \\ f_{\dots} \to 0 \otimes 1 \\ f_n \to 0 \otimes 1 \end{cases}$$

Equation (8) evaluates the *ai* analysis. Equation (9) is an algorithm for searching and evaluating the factor array in Equation (6) in the event the contract fails under the *ai* analysis.

II. Network Application

Algorand is a proof-of-stake blockchain, which improves security and power efficiency across blockchain networks by eliminating miners and validating transactions proportional to an ownership share.[24] In other words, proof-of-stake blockchains are better by an order-of-magnitude measured according to computing costs, energy expenses, and predictable profit. One important technical problem Algorand solves specifically is the majority override attack, a cryptographic hack which results from a competitive computing advantage in mining. Algorand is technically the most advanced and sophisticated blockchain technology, utilizing advance post-quantum cryptographic mechanisms for everyday transactions.

A. Software

A smart contract is a software program which automatically executes, and in some instances transfers cryptocurrency. The software structures and hardware systems facilitating global networks and transactions are altering finance similar to the way in which the Internet changed the nature of information.[25] This is particularly true in the developing world – for example, data regarding consumer retail markets in developing countries such as Malaysia are relatively limited.[26] However, with Algorand, data retention for all transactions is transparent, automatically aggregated, and recorded on a public ledger. In general, the smart contract account address can be sent Algos or other Algorand Standard Assets (ASAs) from any account using a standard transaction.

[24] Yossi Gilad, et al., Algorand: Scaling Byzantine Agreements for Cryptocurrencies, 53 (2017).
[25] Dr. Thibault Schrepel, Collusion by Blockchain and Smart Contracts, 33 Harv. J. L. & Tech. 118, 118 (2019). ("Blockchain may transform transactions the same way the Internet altered the dissemination and nature of information.")
[26] Wei Mei Wong, Consumer Preferences Between Hypermarkets and Traditional Retail Shophouses: A Case Study of Kulim Consumers, 43 Asia Profile 6, 559, 559 (December 2015), http://d-scholarship.pitt.edu/39780/. ("Most data relating to retail choice in Malaysia is from surveys taken in urbanized areas of Malaysia.")

Algorand's Transaction Execution Approval Language, (TEAL) is an execution language that validates transactions on the Algorand Network. Computer programs written in TEAL primarily function to return a Boolean, analyzing and approving transactions. TEAL code has two basic use cases, the first as a contract account – essentially an escrow function, and the second as a delegated signature. When sending from a contract account the logic in the TEAL program determines if the transaction is approved. TEAL programs can be written with any editor but can also be compiled using GOAL, a command line interface (CLI) for the Algorand blockchain.

When used as a contract account the TEAL code is compiled and returns an Algorand address, or TXID number, representing a hash for the transaction. Importantly, the TXID may appear on any of three Algorand Networks, TESTNET, DEVNET, and MAINET. All three Networks may be viewed on Algo Explorer – a real-time software providing Layer-1 blockchain access.[27] Additionally, the Algorand Virtual Machine (AVM) allows users to compile high-level and powerful code directly on a virtual machine, further saving costs while preserving efficiency and power. The AVM is also a CLI providing the ability to use compiled programs within transactions, which is useful for integrating stateless and stateful functionality when necessary. In addition, the AVM also provides the ability to test the program before Network deployment for transactions.

DApps and Blockchain systems have often faced a lack of mainstream acceptance due to both perceived security vulnerabilities and hyper-complex interfaces. The smart contract infrastructure that constitutes most DApps often relies on problematical algorithms for simple operations. These programs often have vulnerabilities that are exploited by attackers. The overarching applications constructed from these smart contacts include multiple barriers to entry for the general population. To have widespread impact, blockchain applications must become simpler while retaining safety and efficiency. This evolution requires fundamental changes in how the smart contracts that drive these applications are built.

Additional security features are also embedded within the Network – for example, Algorand Signatures. In a Proof-of-Stake blockchain such as Algorand, multi-signatures often are used to efficiently validate blocks. However, traditional multi-signatures assume integrity and honest participants. Such assumptions often lead to entire blocks being rendered invalid because the majority of nodes become corrupted due to multiple corrupt participants. While digital signatures are notoriously expensive on computing resources, mechanisms may be imported from the Algorand software stack for signatures. For example, Pixel signatures reduce the necessary bandwidth for standard digital signatures in proof-of-stake blockchains, such as Algorand.[28]

The main programming language used for Algogeneous Smart Contract development is Python.[29] There are two main mechanisms by which Python code is written and deployed, PyTeal and the Algorand Python-SDK. PyTeal is a Python compiler for Algorand's Transaction Execution Approval Language (TEAL).[30] PyTeal is used for both stateless and stateful smart

[27] Algo Explorer, The Algorand Blockchain Explorer (2021), https://testnet.algoexplorer.io/.
[28] Manu Drijvers, et al., Pixel: Multi-signatures for consensus (2019).
[29] Guido van Rossum, An Introduction to Python (2001).
[30] PyTeal Documentation (2021), https://pyteal.readthedocs.io/en/stable/overview.html.

contracts. Because of its democratic rewards-based staking platform, Algorand is the perfect choice to implement any DAO that has a stable voting system and powerful smart contracts.

Finally, the front-end interface for Algogeneous Smart Contracts is developed using Flask. Flask allows developers to have independence with regards to the backend packages they may want to use within Python's ecosystem.[31] Flask includes a Python Library designed for web-development and allows developers to render HTML files directly through a Python backend. Flask leverages Python's powerful libraries to enable the creation of simple yet effective web applications. Specifically, Flask is a Web Server Gateway Interface (WSGI) framework. As a result, Flask communicates effectively with a Python backend.

B. Choice Coin

Choice Coin[32] is a digital asset, which is used to solve the decentralized governance problem. The decentralized governance problem refers to the complex process by which assets are allocated across decentralized networks. Choice Coin solves the decentralized governance problem by providing a mechanism by which decentralized organizations can vote securely using post-quantum cryptography. One purpose for Choice Coin is to provide mechanism for which Algogeneous Smart Contracts may be deployed to transact on the Algorand Network. Choice Coin is a digital asset, which is tokenized (CHOI) and transferable, particularly between Algorand Standard Assets.

An Algorand Standard Asset (ASA) is a digital proof, which may be tokenized to represent value. Built on the Algorand blockchain, ASAs benefit from the inherent security and usability as the main Algorand token, Algo.[33] ASAs enable users to create tokens with specialized manager, freeze, and clawback functions.[34] For example, a freeze function allows one account to freeze the assets in another account according to a public address. Thus, this Paper introduces a new ASA, developed specifically for transacting with Algogeneous Smart Contracts.[35]

Choice Coin serves as the backbone and critical corpus for Algogeneous Smart Contract applications, developments, and transfers. Leveraging the Algorand Python-SDK, Algogeneous Smart Contracts can conduct exchanges between ALGO and any other Algorand Standard Asset, such as Choice Coin. A user, by entering their passphrase, can exchange Algo for Choice Coin and vice-versa. The Python SDK also allows for RSA verification to encrypt the mnemonic for security. Moreover, the Python SDK operates at a higher machine level than PyTeal, making it a simpler and more concise method for scripting.

For Algogeneous Smart Contracts, participants are able to exchange the Algorand cryptocurrency, Algo, at a fixed price to acquire Choice Coin, after which the price volatility will be driven by demand. This creates two fundamental functional features, direct participation and

[31] Flask, Technical Documentation (2021), https://flask.palletsprojects.com/en/2.0.x/.
[32] Algo Explorer, ChoiceCoin (2021), https://testnet.algoexplorer.io/asset/17264161
[33] Additionally, ASAs may be fungible or non-fungible with diverse degrees of control.
[34] Silvio Micali, Efficient Smart Contracts at Scale: Algorand's Stateful Teal Contracts, 6 (2020), https://www.algorand.com/resources/blog/stateful-smart-contracts-teal.
[35] This asset was initialized using Algorand's suggested asset parameters.

robust returns – given the initial fixed price is relatively low to reduce risk. Moreover, users may immediately use acquired Choice Coin to directly participate in various decision making processes, by both staking or sending the Asset.

Choice Coin specifically uses Flask as a deployment mechanism because of its effective integration with the Algorand Smart Contract computer software code. Specifically, the integration is most compatible with the PyTeal and the Algorand Python SDK. All the while, Choice Coin's transactions and storage mechanisms are rendered thorough frontend Flask software. Moreover, Flask includes Jinja templets for scaffolding development.[36] Jinja templates allow for the use of both logical and iterative statements within the main frontend software architecture.

Ultimately, Choice Coin is meant to serve as a voting token that can power autonomous organizations and also serve as the main participation token for both centralized and decentralized organizations. Thereby, Choice Coin is configured to ensure a finite supply, which will also be greater than its total circulating supply. Our chosen supply metrics reflect a scalable strategy to ensure that Choice Coin may have vast use cases, while protecting price volatilities from market speculation. As such, Choice Coin may be aggregated in various silos, for bundled purchases and applications development. Critical to all functionality is security for both Choice Coin and the various smart contract mechanisms by which Choice Coin silos may be deployed.

C. Security

An essential element for privacy,[37] security is arguably the most important feature for blockchains. An advantage for the Algorand Blockchain is an architectural compliance with federal security standards published by the U.S. Department of Commerce for key pair management.[38] Still, optimizing security protocol remains an ongoing task.[39]

Conceptually, there are two ways to hack blockchains – malicious hacking and consensus change. The first is stealing a private key to siphon funds from a victim's wallet, which is likely criminal hacking.[40] For example, malicious hacking involves taking unauthorized control of private keys to secure protected funds.[41] Private keys essentially act as a digital password[42] for

[36] Jinja, Technical Documentation (2021), https://jinja.palletsprojects.com/en/3.0.x/.
[37] Fabrice Benhamouda, et al., Can a Public Blockchain Keep a Secret?, 3 (September 28, 2020). ("Our solution uses anonymous public-key encryption to establish a communication mechanism that allow anyone to post a message to an unknown receiver. We refer to this communication mechanism as "target-anonymous channels."")
[38] U.S. Department of Commerce, Digital Signature Standard, Information Technology Laboratory, National Institute of Standards and Technology Federal Information Processing Standards Publication, FIPS PUB 186-4 (July 2013).
[39] Fabrice Benhamouda and David Pointcheval, Verifier-Based Password-Authenticated Key Exchange: New Models and Constructions, IACR Cryptol, 2 (October 14, 2014). (A "main contribution is to propose two constructions of password hashing).
[40] United States Patent No. 10,891,600 to Rebernik, User private key control (January 12, 2021).
[41] U.S. Patent No. 10,354,236 to Wang, Methods for preventing front running in digital asset transactions (July 16, 2019).
[42] Fabrice Benhamouda and David Pointcheval, Verifier-Based Password-Authenticated Key Exchange: New Models and Constructions, IACR Cryptol, 5 (October 14, 2014). ("A password hashing scheme formalizes the way

blockchain users. So, if a malicious hacker gains access to a public address and the associated private key, then the hacker could siphon any funds stored at the address. Algorand uses participant replacement to combat malicious hacking, which allows malicious participants to be immediately removed from the Network upon identification.[43]

The second blockchain hack is a majority override, a hack resulting from a competitive advantage in mining. Majority overrides should not be considered criminal hacking because they are a consequence of the legitimately logical blockchain software code. In other words, one must first follow the rules to the change the rules on a blockchain for a majority override, which is a formally pure form of democracy. Changing rules fosters innovation. In other words, on the blockchain, software code defines the rules and new code means new rules.[44]

Consider an example, where Bob submits a smart contract to a blockchain network. Bob intends the contract to allow other developers to stake a cryptocurrency in exchange for an annual return paid in a second cryptocurrency. However, after the contract is deployed Alice comes across the contract and realizes there is a logical script which will move the entire reserve of the second cryptocurrency to their address and does so. This is not malicious, nor intentional hacking – rather, moving the reserve allows the natural evolution of the blockchain and promotes innovation. Ultimately, it is the fault of Bob for deploying a bad contract to the network because Alice had no way to know its intended purpose.

To combat the majority override problem, Algorand developed a proof-of-stake chain, differing from classical blockchains, which use a proof-of-work to validate transactions.[45] Quantum computers could also be used to gain an unfair mining advantage.[46] However, it is much less likely quantum computers will be able to overside the consensus mechanism which validates transactions across the Algorand Network because validation is distributed among a network of computers, rather than centralized and based on computational power. Additionally, at this time, quantum computers are secured by institutional protections, limiting public access.[47]

---

the salt and the hash value are generated in order to allow password verification on a server, so that the values stored on the server-side leak as little information as possible on the password.")

[43] Yossi Gilad, et al., Algorand: Scaling Byzantine Agreements for Cryptocurrencies, 66 (2017). ("Algorand avoids targeted attacks at chosen participants using participant replacement at every step.")

[44] The better developer should be rewarded. This is in line with game theory approach to blockchain development, where tokens are points. *See* Jacqueline Morgan, Approximate solutions and Well-posedness in Multicriteria Games, University of Seville (July 1-3, 2002). *See also* Karen H. Wruck, Michael C. Jensen, Foundations of Organizational Strategy, Journal of Applied Corporate Finance, Vol. 10, No. 2 (1997).

[45] Jing Chen, Silvio Micali, Algorand, 72 (May 26, 2017), arXiv:1607.01341. ("Second of all, the partition may be caused by the Adversary, so that the messages propagated by the honest users in one part will not reach the honest users in the other part directly, but the Adversary is able to forward messages between the two parts. Still, once a message from one part reaches an honest user in the other part, it will be propagated in the latter as usual. If the Adversary is willing to spend a lot of money, it is conceivable that he may be able to hack the Internet and partition it like this for a while.") *See also* Fabrice Benhamouda, et al., Better Zero-Knowledge Proofs for Lattice Encryption and Their Application to Group Signatures, ASIACRYPT 2014, PART I, LNCS 8873, 551 (2014).

[46] Brian Seamus Haney, Blockchain: Post-Quantum Security & Legal Economics, 24 N.C. Banking Inst. 117, 130 (2020). ("A quantum computer is a physical system harnessing quantum effects to perform computation.")

[47] Vikas Hassija, et al., Present landscape of quantum computing, IET Quantum Communication (2020).

Algorand is a proof-of-stake blockchain, which is defined as a blockchain that improves integrity and adaptability across blockchain networks by limiting rewards based on proportional scaling. The proof-of-stake blockchain utilizes a new type of cryptographic proof to reap advantages including energy efficiency and structural security.[48] Algorand, by using a proof-of-stake, avoids the majority override problem and other attacks which may relate to computational power disparity. [49]

Most blockchain security and encryption methods use the RSA algorithm[50] or the SHA-256 hash algorithm.[51] Algorand is now developing post-quantum[52] measures for security to evolve the edge in cybersecurity as quantum computing[53] provides the potential to change the industry altogether. For example, quantum secure networks take account of elliptic curve cryptography using complex mnemonic patterns.[54] Indeed, Algorand is a democratic way[55] to implement a public ledger[56] because it uses a staking rewards mechanism.[57]

Moreover, new cryptographic security mechanisms continue to evolve on the Algorand blockchain. For example, The Pixel Signature (Pixel) framework proposes a specific solution in which forward-secure signatures are used.[58] Pixel provides additional security features, ensuring attackers who manage to obtain the private key cannot use it to forge signatures and thus corrupt

---

[48] Fabrice Benhamouda, et al., Better Zero-Knowledge Proofs for Lattice Encryption and Their Application to Group Signatures, ASIACRYPT 2014, PART I, LNCS 8873, 551, 554 (2014). ("Following this observation, we propose a "hybrid" group signature scheme, where unforgeability holds under classical assumptions, while privacy is proved under lattice- based ones.")

[49] Fabrice Benhamouda, et al., Better Zero-Knowledge Proofs for Lattice Encryption and Their Application to Group Signatures, ASIACRYPT 2014, PART I, LNCS 8873, 551 (2014).

[50] R.L. Rivest, et. al., A Method for Obtaining Digital Signatures and Public-Key Cryptosystems (1977) https://people.csail.mit.edu/rivest/Rsapaper.pdf. The RSA algorithm creates a mathematically linked set of public and private keys generated by multiplying two prime numbers together. While, multiplying two prime numbers is computationally inexpensive, figuring out which prime numbers were multiplied to get a number is computationally complex.

[51] National Institute of Standards and Technology, FIPS Pub 180-4: Secure Hash Standard. Federal Information Processing Standards Publication 180-4, U.S. Department of Commerce, 3 (August 2015). The SHA-256 algorithm is the foundation of blockchain mining. The SHA-256 is a one-way hash function, which processes any message of an arbitrary size into a condensed representation called a message digest.

[52] Stewart I., et al., *Committing to Quantum Resistance: A Slow Defence for Bitcoin Against a Fast Quantum Computing Attack*., R. Soc. open sci.5: 180410, at 3. (2018) http://dx.doi.org/10.1098/rsos.180410.

[53] Vikas Hassija, et al., Present landscape of quantum computing, IET Quantum Communication (2020).

[54] World Patent Publication No. 2019/126311 AI, Fast and partition-resilient blockchains (December 19, 2018). *See also* Xianhui Lu, et. al., LAC: Practical Ring-LWE Based Public-Key Encryption with Byte-Level Modulus (August 4, 2020).

[55] Jing Chen, et al., Algorand Agreement: Super Fast and Partition Resilient Byzantine Agreement 9 (April 25, 2018). ("In a permissionless system where the Adversary can corrupt users dynamically, the (small) committees of Steps 4 and 5 may all be corrupted during a partition after sending out their next-votes, and all their messages may be pocketed by the Adversary, in which case their votes may not be propagated to all honest users after the partition is resolved. Since the next-votes are the means of moving to the next period, we introduce new steps and committees in order to make progress after a partition.")

[56] A public ledger is a tamperproof data sequence that can be read and changed by everyone with access to the ledger. See Manu Drijvers, et al., Pixel: Multi-signatures for consensus 1 (2019). ("All PoS-based blockchains, as well as permissioned ones, have a common structure where the nodes run a consensus sub-protocol to agree on the next block to be added to the ledger.")

[57] Jing Chen, Silvio Micali, Algorand, 1 (May 26, 2017).

[58] Manu Drijvers, et al., Pixel: Multi-signatures for consensus (2019).

a node. As a proof-of-stake protocol, Pixel is computationally efficient, giving it potential to be used for security in commercial products using smart contracts.[59]

Algogeneous Smart Contracts utilize a mixture of security techniques within the existing Algorand security ecosystem which are inherent within the Algorand Blockchain, but with additional RSA protections at various network layers.[60] Moreover, as Algogeneous technology evolves,[61] further developments will focus on infusing AI within the cybersecurity mechanism. Specifically, the infusion will involve integrating neural networks for predicting and identifying security vulnerabilities to optimize the smart contract's security mechanism toward the safest possible transaction technology.

Guardians are perimeter neural networks for optimizing smart contract security. As AI becomes an increasing component in cybersecurity, solutions implementing stronger AI must be invented. Guardians meet this need, offering a mechanism by which smart contracts may be further secured by neural networks checking for security vulnerabilities in real time. This will help to reduce the risk of machine learning based attacks.[62]

A neural network is a method for generalizing to make predictions.[63] Every neural network has an input layer and an output layer; and a model's depth is defined by the number of layers between the input and output layer.[64] Each layer of hidden neurons acts as a feature extractor by providing analysis of slightly more complicated features.[65]

$$\begin{matrix} x_n \oplus x_{n+1} \\ x_m \oplus x_{m+1} \\ x_l \oplus x_{l+1} \end{matrix} \bigoplus x^{\circ} \quad x^{*} \quad x^{\circ} \bigoplus \begin{matrix} x_{l+1} \oplus x_l \\ x_{m+1} \oplus x_m \\ x_{n+1} \oplus x_n \end{matrix} \tag{10}$$

Equation (10) is a mathematical model for guardian neural networks using linear operators to optimize a security outcome, x^{*}.

---

[59] Manu Drijvers, et al., Pixel: Multi-signatures for consensus 2 (2019). ("We present the Pixel signature scheme, which is a pairing based forward-secure multi-signature scheme for use in PoS based blockchains that achieves substantial savings in band- width and storage requirements.")

[60] R.L. Rivest, et. al., A Method for Obtaining Digital Signatures and Public-Key Cryptosystems (1977) https://people.csail.mit.edu/rivest/Rsapaper.pdf.

[61] Derek Leung, Vault: Fast Bootstrapping for the Algorand Cryptocurrency, MIT CSAIL, 14 (2019), https://dx.doi.org/10.14722/ndss.2019.23313 ("Vault is a new cryptocurrency design based on Algorand that reduces storage and bootstrapping costs. Vault achieves its goals using three techniques: (1) transaction expiration, which helps Vault decouple storage of account balances from recent transactions and thus delete old account state; (2) adaptive sharding, which allows Vault to securely distribute the storage of account balances across participants; and (3) stamping certificates, which allow new clients to avoid verifying every block header, and which reduce the size of the certificate.")

[62] Hyrum S. Anderson, et.al., *Learning to Evade Static PE Machine Learning Malware Models via Reinforcement Learning*, Cornell University Library (2018), https://arxiv.org/abs/1801.08917. *See also* Thanh Thi Nguyen, Vijay Janapa Reddi, Adversarial Reinforcement Learning in a Cyber Security Simulation 4 (2019), https://arxiv.org/abs/1906.05799.

[63] EUGENE CHARNIAK, INTRODUCTION TO DEEP LEARNING, MIT PRESS 8-9 (2018).

[64] JOHN D. KELLEHER, BRENDEN TIERNEY, DATA SCIENCE, MIT PRESS 134 (2018).

[65] SEBASTIAN RASCHKA, VAHID MIRJALILI, PYTHON MACHINE LEARNING 18 (2017).

Finally, the Guardians' decisions may be reduced for further processing and optimization. Equation (11) represents the outcome as a variable vertical array, to allow further learning over time. Equation (12) is a function representing the relationship between the outcome x^*, and the functional outputs for each respective guardian x°.

(11) $$x^* = \begin{bmatrix} x_i^\circ \\ x_{\dots}^\circ \\ x_n^\circ \end{bmatrix}$$

(12) $$g = x^*(x^\circ * x^\circ)$$

Ultimately, security is a continuous and dynamic process. As such, Algogeneous Smart Contract programming includes adaptable acute adjustment parameters to flexibly scale secure smart contracts.

Conclusion

Most importantly, this Paper introduced Algogeneous Smart Contracts, a new type of simple smart contact, integrated with AI for contractual compliance. Addressing the DATP, which refers to the problem of transferring assets between centralized and decentralized financial systems and across blockchain networks, this Paper introduced Algogeneous Smart Contracts as a solution. Part I detailed the Algorand Network, including stateless and stateful smart contracts. Part II analyzed the Algorand Network, software, assets, and security. In concluding, Algogeneous Smart Contracts offer a new way to use AI on the Algorand Network to transact between parties and solve the DATP.