Anoma: Undefining Money

A protocol for private, asset-agnostic digital cash and n-party bartering

Christopher Goes, Awa Sun Yin, Adrian Brink

April 26, 2021

Abstract

Money, the tokenized representation of value, enables heterogenous goods and services to be compared against each other. Currency facilitates exchange in virtue of this comparative capability by providing a common unit of account so that parties need not find a counterparty with the exact opposite goods for purchase and sale in order to conduct a trade. However, this dimensionality reduction which is the source of money's usefulness is simultaneously the cause of money's failure to accurately represent the underlying value. Dependence on a fiat authority as canonical currency issuer reinforces unipolar power, the hierarchical nature of which leads to capture, surveillance capitalism, and the subjugation of individual rights. Money fails to capture the full implications of economic agents' behaviour, resulting in systemic divergence from the values of the participants at scale. A monetary system aiming to remedy this abstractive deficit must permit anything be used as money, allow multiple parties to discover and trade directly with each other, and provide comprehensive privacy. In this paper, we introduce a protocol for private, asset-agnostic cash, automatic counterparty discovery, and private bartering among any number of parties, and describe how to instantiate it using contemporary distributed ledger technology and cryptographic primitives.

1 Introduction

Money, the tokenized representation of value, has emerged across many cultures in a variety of forms, from stone tablets to zeroes and ones on a rotating disk. Money is deeply embedded in the structure of modern commercial activity, legal structures, and social relations, and taken for granted to such a degree that participants in those structures rarely step back and examine from first principles the many roles which the abstraction of money plays and whether or not such an abstraction serves their collective interest.

1.1 Benefits of money

Money facilitates exchange in the absence of a double coincidence of wants, enabling wide markets and efficiency gains from specialization. Purchases transmit demand up the supply chain, providing information to producers about consumer preferences.

1.1.1 Money facilitates exchange

Consider three residents of Berlin: Albert, Bertha, and Christel. Albert has a fresh crop of potatoes from his schrebergarten, and wants veal for tonight's wiener schnitzel. Bertha has too many slices of veal, and wants a bag of apples for her apfelstrudel. Christel has a bag of apples from her parents' farm, and wants a bag of potatoes for her kartoffelpuffer. If Albert gets the veal, Bertha gets the bag of apples, and Christel gets the potatoes, everyone is happier.

With a representative unit of value, Albert, Bertha, and Christel can offer to buy and sell their goods, and if they can better satisfy their preferences by doing so, trade will occur [1]. The introduction of a common currency provides this sort of representative unit, and the wider the applicability of the currency the larger the market which can take advantage of it to facilitate positive-sum trades between many parties occurring through a succession of single purchases and sales for the standardized unit of account.

1.1.2 Price information enables decentralized coordination

When you choose a bag of potato chips at the supermarket, the shop owner can track which brand you selected when you scan the barcode at checkout and use this information to choose how many of this sort they will supply next month. The potato chip supplier, having estimated aggregate demand, can then place orders with their suppliers for the raw ingredients — potatoes, oil, salt, flavoring, etc. — and the suppliers can then source natural resources and design their supply lines to satisfy this demand. In this way your purchase of potato chips conveys information all the way upstream to energy suppliers and raw goods producers, all without any intentional effort

on your part. As each step along the information transmission line involves a purchasing decision being made by an entity who is themselves competing in a market, the information thereby conveyed about demand is likely to be accurate. Thus, a price system enables invisible coordination at scale without central direction or total knowledge on the part of any individual participant. [2]

1.2 Drawbacks of money

The dimensionality reduction which is the source of money's usefulness is simultaneously the cause of money's failure to accurately represent the underlying value. Dependence on a fiat authority as canonical currency issuer can lead to manufactured economic crises resulting from mismanagement, speculation, or manipulation, and the hierarchical nature of unipolar power lends itself to capture, surveillance capitalism, and the subjugation of individual rights, particularly privacy and freedom of exchange. Money fails to express the full implications of economic agents' behavior (trade, production, and consumption), resulting in systemic divergence from the values of the participants at scale.

1.2.1 Centralized intermediaries lead to monetary capture

The increased trade made possible by a larger single market in conjunction with the requirement of artificial scarcity creates conditions ripe for the emergence of fiat currencies with issuance controlled by a central authority. Such an authority can operate the infrastructure required for physical and digital monetary provisioning and enforce non-counterfeiting rules. However, the centralization of control brings with it inflexibility and the opportunity for abuse. Centrally issued fiat currencies tie individuals and firms to the local economic and political system, and they can be leveraged by the issuing authority as an instrument to enforce economic and social policies and sanction individuals. Cash, the physical variant of fiat, is relatively private but cannot be easily stored or transported in large amounts. Electronic fiat variants can handle large amounts, cross-border exchange, and digital automation more easily, but introduce new intermediaries and create irresistible opportunities for censorship and privacy violations.

1.2.2 Computational intractability of impact

Money mediates causal action, so in order to choose in a way which accurately reflects one's preferences one must understand the impact of spending money — when one purchases a book on Amazon or a carton of bananas at the local grocery store, what are the downstream causal results? Is the author of the book duly compensated? Are the banana farmers able to employ sustainable agriculture practices? Are warehouse workers in the de-

livery supply chain subject to serious workplace safety risks? Does part of the purchase price get redirected to corporate lobbyists in the halls of parliaments? This problem rapidly becomes computationally intractable due to the complexity of the supply chains involved in producing the majority of modern goods and the informational dimensionality reduction at each step of intermediation. This reduction in conjugation with competition then leads to divergence from the values one intended to express — a rubber manufacturer who pollutes the local river may be able to make cheaper rubber than one which does not – thus the purchaser of a new pair of sneakers who picks the cheaper option may unwittingly contribute to this environmental damage, which then impacts their quality of life later on.

1.3 Analysis of existing directions

Since Satoshi Nakamoto's first release of Bitcoin [3], for a little more than a decade, plenty of peer-to-peer electronic cash initiatives have emerged, but none have yet realized this goal. Frail anonymity, insufficient privacy, and confusing user experiences have often deterred adoption. Recent advancements in zero-knowledge proof systems [4] have greatly increased the level of privacy available, but cryptocurrencies have also often combined technical systems for digital payments with particular monetary policies (e.g. fixed-supply deflationary) which do not make for a stable means of payment.

General-purpose distributed virtual machines [5] provide a useful platform for distributed applications, but the base currency economics are also ill-suited to a means of exchange, and the von Neumann step-execution virtual machine model required for general imperative programming is inefficient for cash or barter. Secondary currencies implemented on top of a virtual machine can instantiate a more suitable monetary policy, but users are still required to acquire the ledger's native asset to pay transaction fees, an awkward hindrance. Existing systems with public transactions also grant the block proposer a game theoretical advantage over transaction authors and are thus susceptible to front-running [6], rendering them ill-suited for use-cases where transactions carry price information such as decentralized exchange.

2 Vision

Anoma aims to create a system that allows any digital asset to function as means of exchange or payment, enabling individuals to choose what asset or combination of digital assets are used in their transactions and what values they signify. Unlike existing financial platforms, the goal of Anoma is not to introduce a specific asset intended to be used as money, but rather to facilitate private payments using arbitrary assets, independent of how and by whom they were issued. Anoma can also be utilized by individuals to barter:

directly exchange goods, services, or any digitally representable valuable, including assets created on Anoma, assets sourced from other blockchains transferred to Anoma via interoperability protocols, and stablecoin forms of fiat currencies.

2.1 Digital private cash

In order to protect user privacy and prevent the creation and collection of traceable metadata, Anoma aims to provide private, asset-agnostic, digital cash. Private transfers are transfers where the transaction sender, recipient, amount, and asset denomination are all encrypted, and the relevant invariants are ensured by a zero-knowledge proof. The anonymity set for transfers is a unified shielded pool and thus shared across all assets, as opposed to individual shielded pools per asset. Anyone should be able to send their assets to Anoma in order to transfer them privately.

2.2 Bartering

Bartering is an exchange scheme where parties in a trade directly swap goods or services for other goods or services without requiring a medium of exchange. Bartering and bartering-like systems appeared long before the conception of money. The simple case of bartering consists of two participants who meet and trade, where a double coincidence of wants is required: party A wants what party B has, party B wants what party A has. This kind of bartering requires that not only both parties are available but also that the goods or services in question can be easily transferred. Because of these constraints, bartering is found rarely, only in economies in which there is no standard means of exchange or where the available means of exchange are unreliable.

2.2.1 N-party trades

Anoma implements a digital bartering scheme that can facilitate the trade of goods, services, or digitally represented value. Subject only to computational constraints, n-party trades allow for the exchange of value in cases when such exchange is desired by all parties, without requiring the intermediate usage of a particular currency as a means of exchange and without requiring a double coincidence of wants.

Consider first an example with three parties (`1`, `2`, `3`) and three goods (`A`, `B`, `C`), where party `1` wants to trade `A` for `B`, party `2` wants to trade `B` for `C`, and party `3` wants to trade `C` for `A`. There is no double coincidence of wants, so no 2-party trade can happen in this system, unless the parties elect to agree on some common currency. However, if 3-party trades can be performed, Party `1` can give good `A` to party `3`, party `2` can give good `B` to party `1`, and party `3` can give good `C` to party `2`, atomically, which satisfies all parties' preferences.

The 3-party circular trade generalizes to `n` parties. Consider parties `P_1 ... P_n` and goods `G_1 ... G_n`, where party `P_n` has good `G_n`, wants to trade it for good `G_n-1` (modulo `n`). Then, with an `n`-party trade, each party `P_n` gives good `G_n` to party `P_n+1` (modulo `n`) atomically, and everyone's preferences are satisfied.

2.2.2 More complex state transitions

The goods traded need not have any particular representation; they can be arbitrary state transitions on the ledger. The acceptance criteria for an `n`-party trade are simply that (1) all `n` parties have the combined permissions to effect the relevant state transitions and that (2) all `n` parties acquiesce to the atomic combination of state transitions being made. Thus, the preference functions of the parties themselves need not fix specific goods that they wish to trade, but can rather encode predicates over such goods. For example, Alice may want to trade her meownificent CryptoKitty (created on the Flow blockchain) for either a CryptoPunk with nerd glasses or one with a mohawk, and Bob may wish to trade CryptoPunks (created on Ethereum) for any meownificent CryptoKitty — their preferences can be satisfied by the specific trade "Alice's CryptoKitty (which is meownificent) for Bob's CryptoPunk with a mohawk" — those specific goods are only fixed at the time the trade is settled.

State transitions need not merely be asset transfers at a single instant in time. For example, in a subscription marketplace implemented on Anoma, users could support content creators (of various flavors) in return for exclusive or early access. Creators would receive regular, predictable subscription income and would not be reliant upon sales of particular merchandise. Using the Anoma validity predicate account system, it is possible to support any type of subscription, such as fixed-term (for X months).

As validity predicates can execute arbitrary validation functions, Anoma can easily adopt more complex logic, such as automated market makers [7] or new zero-knowledge circuits for different flavors of private trades.

2.3 Antifragility and fractal scaling

The frequency of commerce is inversely correlated with the distance (of any form) between the counterparties: most buyers in Shanghai are buying from a seller in Shanghai, and most buyers in Berlin are buying from a seller in Berlin. The topology of a digital barter network should reflect this, for both reasons of latency and local sovereignty: transactions in Berlin should be settled on a Berlin-controlled ledger, transactions in Shanghai should be settled on a Shanghai-controlled ledger – and the (far more infrequent) transactions between Berlin and Shanghai should be settled when necessary on a global ledger.

As increasing fractions of commerce occur within purely digital spheres of existence, frequency of commerce may decorrelate with physical proximity, but the same design principle should still apply: Animal Crossing and EVE Online should have their own sovereign ledgers, and only use a common one when necessary.

Fractal division of local ledgers provides not only local sovereignty over commerce but also a much easier path to alterations of the system itself, as different ledgers must only agree on the boundaries and semantics of the interfaces between them, but can otherwise specialize and alter their own logic as suits their community.

3 Instantiation

The Anoma protocol is implemented as a set of interacting subprotocols with clear abstraction boundaries and well-defined roles for actors within them, which together comprise a whole system designed to fulfill the desiderata heretofore articulated.

3.1 Ledger system

The Anoma base ledger system consists of a state machine tailor-made for asset-agnostic private cash and n-party bartering, which is then operated with Tendermint [8] consensus over ABCI [9]. Front-running protection is achieved by a distributed key generation and transaction threshold decryption system.

3.1.1 State machine

The Anoma state machine consists of a validity-predicate account system paired with a transaction execution model designed for n-party trades. Specialized accounts facilitate private transfers & trading.

3.1.1.1 Validity predicate account system

Anoma's state machine implements a validity predicate account model. Like a distributed virtual machine such as the EVM, the ledger contains many independent accounts with their own state subspace and code. Unlike a distributed virtual machine, execution does not proceed in a step-by-step flow where contracts initiate message calls to other contracts. Instead, transactions execute arbitrary code, read and write state as they please, and then all accounts whose state was altered in the course of the transaction decide whether to accept or reject it. The code associated with an account is referred to as a validity predicate, and the validity predicate can also be

changed in a transaction (the change being validated by the old validity predicate before being enacted).

3.1.1.2 Transaction execution model

In order, a transaction is executed as follows:

1. Arbitrary code is executed in a virtual machine (LLVM/WASM) which can read/write all public state.
    1. The ledger tracks which accounts' state was read or written.
    2. The ledger keeps writes in a temporary cache without committing them.
    3. Separate transactions are parallelized based on the read/write graph with a multi-version concurrency control approach similar to that used in PostgreSQL.
2. All accounts whose state was written during the code execution have the opportunity to accept or reject the transaction.
    1. The account's validity predicate is called with access to all state changes which occurred and any extra data in the transaction (e.g. proofs).
    2. This validity predicate check is stateless and thus can happen in parallel for any number of involved accounts.
3. The state changes are committed if and only if all involved accounts accept the transaction.

For the purpose of multi-party exchange, this model has several advantages over step-based execution and contract-originating message calls. First, no coordinating contract is required to handle multi-party exchange, and accounts involved in multi-party exchange do not necessarily need to know about each other's existence. Generally speaking, accounts accept transactions as long as certain invariants (e.g. ownership and preservation of supply for a token) are preserved. Transaction execution can be much more efficient, since state changes can be directly specified (instead of computed) and only validated by the involved accounts, which can be done in parallel. The possibility of code execution during the first phase allows for limited computation in the case of possibly contentious shared resources (e.g. a counter being incremented).

3.1.1.3 Specialized accounts

Specialized zero-knowledge circuits are integrated into Anoma's validity predicate account system as particular predicates. The multi-asset shielded pool [10], an upgrade of the Sapling circuit modified to support arbitrary asset denominations, allows for shielded transfers where sender, recipient, amount, and asset denomination are all private. Other circuits allow for particular kinds of private trades, customized for different use-cases, but all

integrated into Anoma's standard validity predicate interface for seamless interoperability.

3.1.2 Front-running prevention via threshold decryption

Adoption of distributed ledgers for use-cases of decentralized exchange is hindered by the ability of block proposers to "front run" transactions. A proposer choosing whether or not to include transactions has an asymmetric game-theoretical advantage over the authors of the transactions, in virtue of their ability to include, exclude, or reorder transactions based on their contents. In order to prevent front running, this asymmetric advantage must be eliminated. The most straightforward way to do this is to ensure that the proposer must make ordering choices without any knowledge of the contents of the transactions which are being ordered.

In Anoma, this is provided cryptographically by threshold transaction decryption. Periodically, validators run a byzantine-fault tolerant distributed key generation protocol [11], generating a shared public key and private key shards split among the validators. Before submitting them to the peer-to-peer network, users encrypt transactions to this shared public key. The proposer then includes a set of encrypted transactions in a block, committing to a particular execution order. Once the block has been finalized, the validators run a threshold decryption protocol, each generating and gossiping their share of the decrypted transaction. Once the threshold is reached, the decrypted transactions can then be included in a future block, where it will be executed as soon as all prior transactions (in the previously committed-to execution order) have been decrypted and executed. The block proposer only operates on encrypted transactions about which they have no information and thus possesses no game theoretic advantage over the users. This mechanism introduces a small amount of additional latency between transaction submission and execution, but as the threshold decryption protocol requires no interactions between nodes (only aggregation) this should not exceed an additional consensus round. Threshold decryption has approximately the same fault-tolerance and fault accountability guarantees as BFT consensus.

3.2 Intent gossip & matchmaking system

Before any number of parties who might together be able to execute a mutually beneficial trade can do so, they must first be able to find each other. The intent gossip system functions as a means of communicating intent to execute a particular trade, and is designed to allow parties to find each other based on the trade they wish to execute while also preserving efficiency and privacy where possible.

3.2.1 Liquidity routing

The intent broadcasting system consists of a peer-to-peer gossip network paired with a liquidity incentivization system which routes a small portion of the surplus value from any executed trade to the peers who gossiped constituent components (signed intents) required to execute it, based on the incentive-compatible protocol described in *On Bitcoin and Red Balloons* [12]. Participants in the gossip network continuously broadcast binding expressions of intent (signed data, e.g. price quotes on a particular token pair, a bid for a non-fungible CryptoKitty, an acceptable carbon tax rate) and relay and store expressions of intent received from other peers in accordance with their operational costs and expected returns should those expressions result in transaction settlement on the ledger.

When two nodes first connect to each other, and periodically thereafter, they engage in a negotiation process to determine what sort of liquidity each node is interested in, which the counterparty node will subsequently filter forwarded expressions of intent in accordance with. Nodes which do not respect this filter will be disconnected from. Nodes are expected to update their intents quickly, likely by broadcasting binding intends with expiry dates (based on block height or timestamps) soon in the future and continuously rebroadcasting new intents. Most intents will never result in transaction settlement individually, but bandwidth and short-term storage are expected to be cheap.

3.2.2 Privacy preservation

Privacy can be achieved in two ways, depending on the nature of the expressions of intent involved: nodes can construct expressions of intent which do not reveal certain involved parties but include a transfer authorization which can be settled against the ledger and can be verified to have certain contents (e.g. X tokens of Y denomination), and/or nodes can selectively connect to known peers and encrypt data in-transit in order to limit the visibility of their intents. This second method is particularly useful for negotiations which occur between logically proximate devices just prior to settlement, such as at a physical point-of-sale or in an online transaction between known parties, and in the case of physical colocation it is quite possible to use local point-to-point networking protocols (wireless LAN, bluetooth) to perform the negotiation without touching the internet.

3.2.3 Matchmaking

Nodes with access to the intent gossip system can elect to operate a matchmaking algorithm which continuously scans the intent set for compatible counterparty intents which can be combined together into a transaction which can then be executed on the base ledger, settling asset transfers

appropriately. For certain kinds of trades, matchmakers may be able to collect a spread, for others, intent authors may include a small fee in order to encourage matchmakers to matchmake on their behalf. The role of matchmaker is not permissioned — intent authors can also matchmake for themselves or directly with each other.

3.3 Fractal scaling

A single global intent gossip system and ledger will not scale to the intent gossip and settlement throughputs which can be reasonably expected should the protocol become widely adopted, and even if the scaling problems could be solved such a ledger would present a difficult-to-govern and fragile single point of failure. The topology of intent gossip and trade settlement should reflect the topology of the underlying commerce, both to facilitate scaling and to provide local sovereignty and antifragility, such that access to the local intent gossip system and local ledger is not dependent on globe-spanning internet networks and consensus algorithms. What constitutes "locality" may vary — locality may be geographical, topical, or cultural — geographical locality is most important for geographical fault-tolerance, but topical and cultural locality may also be important for self-sovereignty and memetic antifragility.

Trade settlement should generally happen at the most local layer possible. Separate instances of the protocol will be connected by a cross-chain message passing protocol [13] in order to facilitate asset transfer to and from different global and local layers. Assets can also be locked and directly traded cross-chain without prior transfer. All of this should be cleanly abstracted away from end-user interfaces with automatic selection and cross-chain transfer (subject to appropriate security considerations).

3.4 Upgrade system

The protocol utilizes a directed acyclic upgrade system which allows multiple new versions to be tested in parallel with a sliding scale of value-at-stake.

3.4.1 Motivation

Initial versions of the ledger, trade settlement, and intent gossip protocols will need to evolve and iterate over time in accordance with usage patterns, user feedback, technical developments enabling more performant, private, or secure constituent components, and any discovered bugs or vulnerabilities. Upgrades of distributed systems require both technical coordination, as operators such as validators, peer-to-peer gossip nodes, liquidity providers, and user frontends must agree on which version of the protocol they are using, and social coordination, as various parties wishing to make different

changes to different parts of the protocol must coordinate in order to ensure that their changes are compatible and to combine them together in a subsequent version of the software. For certain upgrades, coordination may also be required on liquidity, which will often be tied to a particular version of the protocol.

Countervailing concerns when designing an upgrade system include the requirement not to compromise the security model of the ledger, the desire to avoid subjecting a minority of users to a protocol upgrade they do not wish to enact, and the desire to avoid imposing high operational costs on infrastructure providers.

3.4.2 Prior efforts

Prior efforts to architect upgrade mechanisms [14] [15] [16] have focused primarily on voting mechanisms and automation of the processes responsible for delivery, compilation, and activation of new software versions. These efforts are helpful, but they do not address the social coordination problems. Specifically,, these systems require a binary switchover where a new code version is either not yet activated, at best operating in a testnet with nothing at stake, or immediately activated and responsible for stewardship of all asset value from the previous version. This linear version progression and binary switchover prevents multiple new versions from being meaningfully tested at once (with real value at stake and real assets) and leads to an extremely conservative software development process to mitigate the risk of bugs in a new software version which will immediately put at risk the entire network should there be a vulnerability.

3.4.3 Mechanism

The Anoma protocol instantiates a directed acyclic upgrade system. Multiple versions of the ledger can be run in parallel; they must only agree at the cross-chain interface boundary in order to exchange assets and conduct cross-chain trades. Anyone can launch a new version, changing the software as they wish —any validator can elect to validate, and anyone electing to use the new version can move assets over and start settling trades. Any number of new versions can be tested in parallel, and they can scale up from little value at stake to a reasonable fraction of the prior version of the protocol. The same mechanisms used for cross-chain liquidity sharing and trade settlement between fractal instances can be used between different versions, subject to agreement at the interface boundary.

At any point in logical time, there is one primary ledger, demarcated as the canonical controller of the protocol token and the canonical reporting location for protocol fault accountability remediation procedures. The primary ledger can be atomically switched by a two-thirds majority of the stake-

holder set using the usual threshold commitment mechanism (so commitments to different versions can be made; as soon as a version reaches two-thirds the primary ledger switches). New versions may elect to alter the token supply (e.g. allocating themselves some payment for authoring the new software, or paying initial validators of the new chain), and if the primary ledger switch occurs, these alterations are realized. Until then, the usual supply guards apply, so no more tokens can be transferred out of a new version than were initially transferred to it from the primary ledger. The primary ledger may also elect to subsidize testing by minting additional proof-of-stake rewards for software authors and validators, again using the intent-based threshold commitment mechanism. A cross-chain validation protocol allows the primary ledger to track the validator set and provide fault accountability for new versions in testing, as desired.

Old versions can remain running for a while, during which period users can continue to use the old version or transfer assets as they wish, although certain aspects of security will now be dependent on the new primary ledger. After a certain period, validators may prefer to cease validating the old version, at which time state will be automatically migrated by cross-chain message passing before the old version shuts down.

4 Conclusion & related work

This paper has described the Anoma protocol, covering the design motivations, vision, abstract model, and mechanisms of instantiation. The Anoma vision paper [17] constructs the fundamental game-theoretical model and coordination technology the protocol aims to provide, while the Anoma technical specification [18] details specifics of the architecture of the protocol, including the ledger, trade system, and the intent gossip and matchmaking protocols. The Anoma token economics paper [19] describes the design of Anoma's native token, the permanent incentives that aim to ensure that participants fulfill particular roles indefinitely and that their incentives are aligned, and the temporary bootstrapping incentives designed to shift future expected value into present value.

References

[1] N. Szabo, "Shelling out: The origins of money." https://nakamotoinstitute.org/shelling-out/, 2002.

[2] F. A. von Hayek, "The price system as a mechanism for using knowledge," 1985.

[3] S. Nakamoto, "Bitcoin: A peer-to-peer electronic cash system." http://bitcoin.org/bitcoin.pdf, 2009.

[4] C. G. Eli Ben-Sasson Alessandro Chiesa, "Zerocash: Decentralized anonymous payments from bitcoin," 2014.

[5] Dr. G. Wood, "Ethereum: A secure decentralised generalised transaction ledger." https://ethereum.github.io/yellowpaper/paper.pdf, 2014.

[6] T. K. Philip Daian Steven Goldfeder, "Flash boys 2.0: Frontrunning, transaction reordering, and consensus instability in decentralized exchanges." https://arxiv.org/abs/1904.05234, Apr-2019.

[7] H. Adams, "Uniswap V2." https://uniswap.org/docs/v2/, 2020.

[8] Z. M. Ethan Buchman Jae Kwon, "The latest gossip on BFT consensus." https://arxiv.org/abs/1807.04938, 2019.

[9] "ABCI | tendermint core." https://docs.tendermint.com/master/spec/abci/, 2021.

[10] "Multi-asset shielded pool specification." 2020.

[11] A. Kate, "DKG in the wild." https://eprint.iacr.org/2012/377.pdf, 2012.

[12] M. Babaioff, "On bitcoin and red balloons." https://arxiv.org/abs/1111.2626, Nov-2011.

[13] C. Goes, "The interblockchain communication protocol: An overview." https://arxiv.org/abs/2006.15918, Jun-2020.

[14] L. M. Goodman, "Tezos — a self-amending crypto-ledger." https://tezos.com/static/white_paper-2dc8c02267a8fb86bd67a108199441bf.pdf, Sep-2014.

[15] J. Kwon, "Cosmos: A network of distributed ledgers." https://cosmos.network/cosmos-whitepaper.pdf, 2016.

[16] Dr. G. Wood, "Polkadot: Vision for a heterogenous multichain framework." https://polkadot.network/PolkaDotPaper.pdf, 2017.

[17] "Anoma vision paper." 2021.

[18] "Anoma technical specification." 2021.

[19] "Anoma token economics paper." 2021.