Celo: A Multi-Asset Cryptographic Protocol for Decentralized Social Payments

Sep Kamvar, Marek Olszewski, Rene Reinsberg

DRAFT version 0.22

Abstract

Two of the biggest barriers to the large-scale adoption of cryptocurrencies as a means of payment are ease-of-use and purchasing-power volatility. We introduce Celo, a protocol that addresses these issues with an address-based encryption scheme and a stable-value token. We show how these attributes together can be used to foster a monetary ecology that includes global reference currencies, local and regional stable-value currencies, and a social dividend. Our first application is a social-payments system centered around mobile phones.

Contents

1 Introduction

Cryptocurrencies have several advantages to fiat currencies as a means of payment. They enable transfer of value that is much faster than a bank wire, at lower cost (especially for international payments), in a publicly auditable and secure manner, using a technology that is globally accessible so long as you have a smartphone. Further, cryptocurrencies can be programmed; allowing financial contracts, escrow, and insurance, all without intermediaries.

However, at the moment, there are several barriers to the mainstream adoption of cryptocurrencies as a means of payment. First, due to deterministic supply rules and unpredictable coin demand, successful coins[1] experience deflationary price instability. As a result, users rationally prefer to use them as a store of value rather than a medium of exchange. Second, even when people do wish to use price-volatile cryptocurrencies as a means of payment, they need to generate a private/public key pair to receive a payment, and enter in somebody's public key in order to send a payment. While these may seem small obstacles, experience has shown that small differences in user experience lead to large differences in usage outcomes.

For a cryptographic social payments system to prosper, sending a payment should be as easy as sending a text message and the value of the currency should be stable. We describe Celo, a protocol that addresses each of these issues. To address ease of sending payments, Celo introduces a cryptographic scheme that we call address-based encryption, in which participants verify a series of cell-phone number-to-public-key mappings, allowing users to then use their friends' cell phone numbers as public keys.

To address stability of value, Celo introduces a token whose value is stabilized using a monetary policy with elastic supply rules, backed by a variable-value reserve. Further, it introduces a governance structure that allows the protocol to create a family of local, regional, and utility stable-value currencies, where the introduction of new successful stable-value coins to the family strengthens the stability characteristics of the existing coins.

Finally, Celo introduces a mobile block reward mechanism in which all users involved in transactions are also able to participate in verifications, creating a broad participant base and making block rewards more accessible to day-to-day users.

Together, these underpin a compelling social payments protocol.

2 Ease of Use through Lightweight Identity

An important obstacle for the mainstream adoption of cryptocurrencies as a means of payment is the lack of intuitive, decentralized public key infrastructures. As a result, in order to send a payment in today's decentralized systems, users must know the public key of the intended recipient (unless they are operating through a centralized gateway). And in order to receive a payment, a user must first set up a private/public keypair and broadcast it. It would be far easier to send a payment directly to an email address or phone number, and to be able to receive a payment without having to first set up a wallet.

Identity-based encryption [18] holds promise towards this end. In this scheme, when Alice wants to send an encrypted message to Bob at bob@company.com, she can simply use the public key string bob@company.com, without needing to obtain Bob's public key certificate. While a cryptocurrency system based on identity-based encryption would lead to a much more seamless user experience, both the original proposal and subsequent implementations [4, 6] are hindered by the fact that they require a trusted third-party, called a private-key generator, to generate private keys. As a result, these schemes are less useful in open, permissionless systems.

[1] Academics, regulators, entrepreneurs and others use "coin" and "token" interchangeably to describe assets that function as a digital representation of value native to a distributed ledger. In this paper, we refer to 'digital assets,' 'coins,' 'cryptocurrencies' and 'tokens' with general interchangeability.

2.1 Address-Based Encryption

We propose a variant on identity-based encryption, called address-based encryption. Rather than directly using an e-mail address or phone number as a public key, and then relying on a trusted private-key generator to generate a corresponding private key, we have users generate their own private/public key pair in the traditional manner. The user then registers their public key in a public, append-only database that stores `[address -> public key]` tuples. This database is functionally decentralized, so that no central owner is responsible for storing, managing, or maintaining the database, but logically centralized, so that everybody can at any time see all the entries in the database. Crucially, the `[address -> public key]` tuples are verified by a peer-to-peer network. To perform verification, validators in the network send a signed and secure message to the registrant, who then signs the message with her private key and returns it to the validator. The validators then append the message, the signed message, and their own signature to the entry.

This protocol works not just with email addresses, but with any channel to which a secure message can be sent, for example, a cell phone number, an IP address, or even a bank routing and account number. Further, arbitrary strings may be appended to the address in the database key, allowing multiple public keys to be stored for each address, each for a different application. As a consequence, the encryption scheme supports a large number of cryptographic applications, from two-factor authentication to decentralized social networks, without relying on trusted third-parties.

For the social payments use case, it allows for two important features. First, a user can send Celo currencies to a friend by using her phone number as the public key, allowing easy payments to contacts. Second, a user can send Celo currency to a friend even if the friend has not yet downloaded a Celo wallet.

2.1.1 Single-Node Address-Based Encryption

For the purposes of explanation, we begin by describing a simplified version of the address-based encryption scheme in which a single node, called a validator node, maintains the state of the system.

The key role of the validator node is to maintain a public, append-only database of verified `[address -> public key]` mappings. In the single node case, the validator node is similar to a traditional key server except that it not only stores the `[address -> public key]` mappings, but also verifies them, as follows:

When a user wishes to register a public key with the scheme, they generate a private/public key pair, and then submit their `[address -> public key]` mapping to the validator node. (In our use case, the address is the cell phone number of the user, but in the general case it could be any address to which a secret message can be sent.) The validator node sends a secret message to the address in the entry, and waits for the user to sign that message and send it back in response. When the node receives the signed response, it verifies the signature by decrypting it with the public key in the corresponding entry. If the decrypted message matches the secret message, the validator node writes the following entry to the database `[address, user public key, secret message, user signed secret message, validator signature]`.

2.1.2 Drawbacks

This simplified version has the following drawbacks:

Address harvesting. A publicly viewable database with unencrypted phone numbers allows spammers to harvest the cell phone numbers of all of the users. As a solution to this, we can store a one-way hash of the address rather than the address itself. This allows address lookups without enabling harvesting. To increase the entropy of the underlying string (to make reversing the hash more difficult) we may append a pepper to the string to be hashed[2].

[2]Even with an appended pepper, the following scenario is possible: a spammer one-way hashes every possible 10 digit number along with every possible pepper, and then checks to see which hashed values are in the database. However, harvesting at high cost is possible even today, by taking every possible 10 digit number, sending an SMS to each, and seeing if it goes through. Therefore, our goal would be to make the effective cost of decryption more expensive than the cost of sending a bulk SMS.

Single key per address. In practice, people may want to store multiple public keys associated with their address. The simplified protocol gives no mechanism to do so. As a solution to this, we can allow the key to be the hash of an address concatenated with an optional arbitrary string. This allows, for example, Bob to store an application key at `hash("4155551212 || application_name")`, or an ephemeral application key at `hash("4155551212 || application_name || 20171117")`.

Node failure. Any model that relies on a single node to maintain state is susceptible to that node failing. We can address this by having multiple nodes participate in maintaining the state. (In doing so, we must also ensure that only a small number of nodes send a secret message to a user issuing a verification request, to avoid overloading the user.) In this model, the secret message must also be verifiable by other validators, even if they did not construct it. This is achieved by signing the message with the private key of the validator sending it. To avoid repeat-attacks, each message from the same validator must be unique.

Malicious Validator. A malicious validator node may choose to bypass the message/response step, and instead, write an entry to the ledger in which they choose somebody else's address, generate their own key pair for that address, and then sign the secret message with the private key that they generated. Doing so allows the validator to spoof an address, claiming payments intended for somebody else. We can address this by requiring consensus between multiple validators who have no mechanism to collude.

Transaction Transparency. If we are using hashed phone numbers as public keys, then a traditional bitcoin-style blockchain will allow a user to see the transactions of the contacts in their address book. We can address this by implementing the computationally efficient version of zk-snarks as described in [12].

DDoS. Finally, a malicious user may submit thousands of bogus requests to the validator, both tying up the validator and effectively using the validator as a spam agent. We can mitigate this by introducing a cost to verification.

2.1.3 Distributed Scheme

We introduce here a distributed scheme that introduces each of the features suggested above. In this scheme, rather than the single validator node we describe in Section 2.1.1, a peer-to-peer network of multiple validator nodes maintains the database. The network is open and permissionless; anybody may join as a validator, and validators may leave and rejoin the network at will. Each validator maintains a full copy of a verification pending queue and a verified user database. Messages, including verification requests and verifications, are broadcast on a best-effort basis. Because each verification request includes just the hash of the address and not the actual address itself, users wait for their verification request to be added to the latest block, and then send their actual address directly to the validator that processed that block. The validator can easily verify the authenticity of such addresses by computing the hash and comparing it to the original broadcasted verification request.

A verification workflow would then look like this. First, a user will issue a verification request by broadcasting the request to the network, along with a verification fee. Validators compete through a proof-of-stake-based leader election protocol for the right to write a block. For each block, the block creator must perform all verifications for that block (by waiting for the addresses and then sending a secret message to the requesters), and also verify the validity of the signed responses (verifications) for each of the verification requests from the previous block. The user broadcasts three consecutive verification requests to the network, and after three successful verifications, the last validator may write that the user is verified into the latest block.

Requiring consensus verification amongst three different validators addresses the node failure and malicious validator scenarios above. The verification fee addresses the DDoS issue. And the verification requests are issued as a hash of the `(phone number | application string)`, so as to avoid address harvesting, and to allow for multiple keys per address.

2.1.4 Summary of Operations

An alternative way of framing the protocol is in describing the roles and operations allowed to each node in the system.

Any user may:

- request verification of a public key associated with her address, by broadcasting her `[hash(address | optional appended string) -> public key]` tuple to the verification pending queue

A verified user may:

- add a new public key by creating a `[hash(address | optional appended string) -> public key]` mapping
- revoke any public key associated with their address
- change any public key associated with their address

A validator may:

- compete with other validators for the right to write a block and send a secret message to the addresses on the verification pending queue, and validate the signed responses of the previous block's verifications.

Anybody may:

- look up the public key for a given address hash (or address hash || string concatenation) in the verified user database.

2.2 Aggregating Reputation Signals through Encrypted EigenTrust

Once there exists a decentralized mapping of phone numbers to public keys, it brings about interesting potentialities for trust computations in the payment network. For example, it can be used to bootstrap a reputation system that helps users determine the trustworthiness of any new users they may transact with.

A person's cell phone contact list is a rough first-order proxy for a list of people that in whom she has a certain level of trust. One can imagine refining this trust proxy through explicit signals (for example, a user may rate people in her contact list in an application-specific manner, or attest to whether a contact in their address book is a person or not), and implicit signals (for example, if a user makes a payment to somebody in her contact list). These signals can be maintained locally, on the user's cell phone, without sharing them with anybody else.

Such address-book based trust signals define a trust network that is both logically decentralized and functionally decentralized. No single entity stores or has visibility into the entire trust network; each user simply knows the people whom they trust, and the level to which they trust them. We describe below how to compute sybil-resistant, privacy-preserving aggregate reputation scores given this decentralized trust network.

2.2.1 EigenTrust

EigenTrust [14] is a decentralized algorithm for computing global reputation scores, given pairwise local trust scores. The key intuition behind EigenTrust is that a person's reputation score can be defined as the number of people who trust that person, weighted by their reputation scores. This recursive computation converges for all nodes to the principal eigenvector $\vec{t}$ of the trust matrix T, where T_{ij} is number between 0 and 1, and whose magnitude is proportional to the relative level that node i trusts node j [3].

[3] An alternative way to frame the problem is to compute the stationary distribution of the ergodic Markov chain described by the trust network.

In EigenTrust, the principal eigenvector of T is computed using a distributed variant of the Power Method [20]. In the context of a social payments network, it would proceed as follows: The trust network T_{ij} would be some variant of the payment network, where T_{ij} would be nonzero if node i has paid node j, and node j is in the address book of node i. Each node stores their own current t_i, and has access to the values of T_{ij} in row i and column j (the people with whom the node has interacted). The principle eigenvector $\vec{t}$ would then be computed in an iterative fashion as follows. At each iteration, each node send across their $t_i \cdot T_{ij}$ scores to each node j that they've paid in the past. The nodes j wait to receive all of the scores from the nodes that have paid them in the past, and then compute their own t_j, and then pass their $t_j \cdot T_{jk}$ along to the nodes k that they have paid.

2.2.2 Privacy-Preserving EigenTrust through Zero-Knowledge Proofs

There are two differences between the algorithm we propose and the original EigenTrust algorithm.

First, the simplified description above allows nodes to lie about their own t_i. The original EigenTrust algorithm addresses this by relying on score managers to steward the computation of t_i for each node. In the original scheme, each node has three score managers, assigned at random through a distributed hash table, who store the T_{ij} values for each node and compute and store t_i for each node. While this addresses the dishonest node attack, it is not ideal in the social payments scenario, as it requires sharing transaction information with other peers in the network. We address this by having each peer perform the computation themselves, as per the simplified version, but also prove, to a high probability, to all adjacent nodes that they have performed the computation correctly. One can do so by constructing a zero-knowledge proof using a variety of cryptographic means, including [10, 3, 5].

2.2.3 Personalized Pre-Trusted Peers

Second, in order to break malicious cliques, and to ensure convergence of the power method and uniqueness of the principal eigenvector, EigenTrust introduces the notion of pre-trusted peers, a group of peers that are active and assumed to be universally trusted. This ensures that the graph is acyclic and strongly connected (and that the matrix is irreducible and that the problem is well-conditioned). However, it requires the system to define a set of universally trusted peers, and concentrates outsized power to confer reputation in those pre-trusted peers.

We can address this through personalization. Rather than computing a single global reputation vector, the system can compute a personalized global reputation vector for each peer, that gives the reputation score of each peer j in the network from the point of view of a single peer i. To compute personalized EigenTrust for peer i, one can simply perform a traditional EigenTrust computation, but use the contact list of peer i as the set of pre-trusted peers.

This is far more computationally expensive than a single EigenTrust computation; however, we apply many of the computation-saving techniques that enabled personalized PageRank [13] to a personalized EigenTrust computation.

2.2.4 Practical Implications

For the social payments case, in which people text money to friends, the address-based encryption scheme suffices as a lightweight identity proxy, allowing people to send money directly to people's cell phone numbers, even if they have not signed up for a wallet.

As people are interested in using the protocol to pay people outside of their direct circle of contacts, it is useful for a user to be able to aggregate the trust signals of those in their network to make purchase, payment, and credit decisions, and to mitigate bad actors.

Further, a reputation scheme as we described enables a more robust identity scheme. Most identity schemes are based on attestations from others, and it would be useful to be able to weight those attestations by the reputation score of the attestor.

3 Stabilizing Value

Perhaps the biggest hurdle to the use of cryptocurrencies as a means of payment is their volatility. Consumers are unlikely to want to buy a volatile cryptocurrency to spend it, since the purchasing power of their accounts would fluctuate widely with market demand for the currency. Merchants who accept cryptocurrencies are likely to convert to fiat upon payment, because their business model does not involve speculating on cryptocurrencies. And the most successful cryptocurrencies today are not just volatile but deflationary – their success leads to their price rising; as a result, prices denominated in the currency fall. Rational behavior would be to use such currencies as a store of value rather than a medium of exchange, and in practice that is what has happened.

Stable-value cryptocurrencies would bring a number of benefits to the cryptocurrency ecosystem. For one, stable prices remove a considerable barrier for using cryptocurrencies as a medium-of-exchange; salaries, prices of goods, fixed obligations, can all be set in a stable value cryptocurrency without requiring either party to speculate on the future value of the currency. Further, financial contracts are more easily built with a stable value coin, because the issuer can separate the function of the contract from the price risk of the currency in which it's denominated.

While a single stable-value currency would be helpful, a thriving cryptoeconomy is best-served by a family of stable-value currencies, much as it is well-served by the family of variable-value crypto-assets that we have today. Certainly a cryptocurrency pegged to the US Dollar has several uses, from social payments in the US, to user-initiated dollarization in hyper-inflationary markets, to the efficient settlement of high-frequency crypto-asset trades. At the same time, a cryptocurrency pegged to the Euro would also be useful for many purposes, as would a cryptocurrency pegged to the price of a basket of goods in Greece, as would a cryptocurrency pegged to the price of a barrel of oil, or housing in San Francisco. Stable-value local, regional, and utility currencies allow people to hedge price risk in their lives by denominating a portion of their personal economy in currencies that are stable vis-a-vis the price of the goods they regularly use.

3.1 Elastic Coin Supply and Shifting Volatility Risk

Several protocols have been proposed for a stabilized value cryptocurrency (for example [17, 2, 1, 19]). While a full review of these proposals is outside of the scope of this paper, they generally share two properties. First, rather than a deterministic coin supply rule (in which the coin supply and growth rate are determined in advance, independent of exogenous information), they each introduce an elastic coin supply rule, that stabilizes the value of the coin by adjusting the supply of the coin to match the demand. Second, they each introduce a multi-asset ecology, in which one coin is intended to be stable, while one or more complementary crypto-assets bear the risk of a decrease in stablecoin demand (and receives a reward in the case of an increase in stablecoin demand). In essence, they each shift volatility risk from the coin holders to the complementary asset holders.

Our protocol utilizes the same two key intuitions, with five novel features: (a) it introduces a multi-asset tiered reserve that supports several local and regional stable value currencies, (b) it sets expansion and contraction parameters that are tuned to the reserve ratio defined by the tiered reserve, (c) it introduces a decentralized exchange in which the different local and regional currencies and the reserve currency can be traded amongst one another without a central party, and that the protocol can use to perform expansions and contractions, (d) it releases block rewards and other incentives in the reserve currency, and (e) it has a governance mechanism in which long-term stakeholders in the reserve currency are responsible for governing the assets held in reserve and the new local currencies that are introduced.

3.2 Protocol Summary

At a high level, the protocol proceeds as follows:

1. The protocol establishes a fixed supply of reserve tokens, called Celo Gold, a portion of which is distributed over time. From the initial token distribution, 50% is allocated towards a reserve: the protocol places half of that Celo Gold directly in reserve, and sells the remaining 50% for

diversified basket of crypto assets such as Bitcoin and Ether, which also get deposited in the reserve.

2. The protocol also establishes a means-of-payment currency, called the Celo Dollar, that is intended to be pegged roughly to the US Dollar, that adheres to the following elastic coin supply rule:

   When coin supply needs to expand (when the price of Celo Dollar is above the peg), the protocol creates new coins, as in [17, 1, 2]. But rather than distributing them to token holders, it uses them to purchase a basket of cryptocurrencies[4] at market rates through a smart contract. These purchases get added to the reserves. This is analogous to a central bank expanding the money supply by buying financial assets on the open market and depositing them in the reserves.

   When the coin supply needs to contract, the protocol uses reserve assets to buy Celo Dollars on the open market. This is analogous to a central bank selling financial assets on the open market in order to contract the money supply.

3. Optionally, to further give contraction elasticity, one could imagine introducing a bond-like mechanism, as per [1], in which third-parties can use Celo Dollars to purchase future claims to the reserve at a discount. This allows the protocol to contract the Celo Dollar supply without relying on the reserve in the near-term. From the perspective of the buyer, it would be rational to purchase these claims so long as the purchaser believes that the aggregate value of the reserve will increase over the term of the claim [5].

   We refer to the version of the protocol that includes this mechanism as the elasticity-enhanced version of the protocol, because it can allow for larger short-term contractions in demand. For simplicity, in this paper we will refer to the discounted reserve claims in the elasticity-enhanced protocol as bonds (and the holders of these digital assets as bondholders), even though in practice these assets will look quite different than bonds. We plan to launch the protocol without this elasticity enhancement, but we mention it here because it would be a natural extension of the protocol.

4. The protocol has a variable rate transfer fee on Celo Gold, to encourage long-term holding of the reserve currency. The proceeds from the fee goes to bolster the reserves, and the rate is based on the reserve ratio – the lower the reserve ratio, the higher the transfer fee.

5. The protocol uses a proof-of-bonded-stake model for governance. The weight of a node in governance decisions is dependent on the amount of Celo Gold they own and the length of time remaining in the bonding of that Celo Gold. This further encourages the long-term holding of the reserve currency, and aligns the interests of those making governance decisions around long-term stability of Celo Dollars.

6. Every time a block reward is distributed, an equivalent portion of Celo Gold is released. If the reserve ratio is substantially higher than the target reserve ratio, then the released amount is largely allocated for incentives (e.g. to developers and users). If the reserve ratio is substantially lower than the target reserve ratio, the released amount goes mostly to towards bolstering the reserves.

3.3 Shared Reserves

While a single stable coin would be useful for several purposes (for example in cryptoasset trading and internet commerce), a more robust ecosystem would involve a family of local, regional, and utility stable value coins. The benefits of such a monetary ecology has been discussed broadly, for example in [9, 16, 15], but here we focus on one: a stable currency is only meaningful if it is stable vis-a-vis the price of goods and services that are purchased using that currency. Using a global currency for local

[4]Initially, Celo Gold, and longer-term through a basket of cryptoassets via cross-chain decentralized exchanges once available

[5]either through an increase in demand for Celo Dollars, or an increase in the value of the reserve assets

transactions would introduce price volatility in regions where regional consumer price dynamics vary from the global consumer price dynamics [6].

From a protocol perspective, we are interested in two mechanisms here: (a) a governance scheme that determines how the protocol makes decisions on introducing new regional stable coins, and (b) a structure in which the introduction of a new successful stable coin increases the stability characteristics of the coins in the family.

As a starting point, we can imagine a protocol where each new stable coin is independent – there is a blockchain and reserve for each new currency introduced. In this scheme, the governance question is straightforward – teams will independently choose to introduce new stable value coins outside of the protocol, and people can choose independently to purchase the new coins and their complementary reserve assets. Governance on this issue is determined outside of the protocol, by the market.

However, this simplicity comes at a cost: the introduction of a new successful stable coin has no stabilizing effect on existing stable currencies, and on the margins it has a small destabilizing effect [7].

To address this issue, we introduce the idea of shared reserves. When the protocol introduces a new stable value coin – for example, a stablecoin pegged to the Euro – the reserves for that coin are the same reserves for Celo Dollars. When the supply of Celo Euros needs to expand, it expands using the same mechanism as with Celo Dollars – the protocol creates new Celo Euros, and uses those to purchase a basket of crypto assets for its reserves. When the supply of Celo Euros needs to contract, the protocol uses the same mechanism as before: it sells reserve assets in exchange for Celo Euros and retires the Celo Euros.

The protocol can make this process more efficient in the following manner: before selling the reserves, it first looks to see if the supply of Celo Dollars needs to expand. If so, it creates Celo Dollars, exchanges them directly for Celo Euros at the prevailing exchange rate, and retires the Celo Euros. This is functionally equivalent to selling reserves in exchange for Celo Euros, retiring the Celo Euros, and then buying reserves in exchange for Celo Dollars; it just disintermediates the reserves. It only uses the reserves directly if the need for contraction of the Celo Euros is greater than the need for expansion of all the other stablecoins supported by the protocol.

There are several benefits to a shared reserve model. Celo Gold holders are rewarded not just when the long-term demand of the dollar-pegged stable coin expands – they are rewarded when the long-term aggregate demand of all stable coins supported by the protocol expands[8], including coins that are not yet supported by the protocol. Therefore, the introduction of a Euro coin can benefit Celo Gold holders even if it decreases demand for a dollar-pegged stable coin, so long as the aggregate demand for dollar-pegged and Euro-pegged coins is higher than that of dollar-pegged coins alone. This will lead to a natural introduction of regional stable-value currencies where they are useful (where their introduction will increase aggregate demand for a stable coin).

Further, it allows a natural means for seamlessly introducing innovations. For example, if a CPI-pegged currency is deemed useful at a certain point, the protocol may introduce a CPI-pegged coin even if it competes directly with a fiat-pegged coin. If people find the CPI-pegged coin to be more useful than the fiat-pegged coin, the demand for the CPI-pegged coin will expand as the demand for the fiat-pegged coin contracts.

A shared reserve system must come together with a thoughtful method of governing decisions on what new stable coins to introduce, and when to introduce them. If a new stablecoin is introduced that has negative utility to the ecosystem, it can have a marginal negative impact on the stability of the other currencies if the demand for that currency is high enough and volatile enough (for example, a celebrity vanity stablecoin early on), or if the coin decreases aggregate demand for other coins supported by the protocol (for example, the introduction of several duplicative regional currencies in the same region with no differentiating features, causing confusion). For this reason, it is useful to have a governance model that introduces a new stablecoin only if there is a widespread expectation

[6] For example, with Greece and the Euro, or with dollarization in Uruguay

[7] If the demand for the new stable coin is high enough, it could potentially cause a contraction in demand for existing stable coins, reducing the value associated with the complementary assets of those existing coins, and increasing uncertainty around long-term demand of the existing coins.

[8] The actual reward is slightly more nuanced than that, they are rewarded when the stable coin demand expands and the value of the underlying reserves appreciates

that its introduction would increase the aggregate demand for the family of coins over the long run. We describe this governance model in Section 4.4.2.

It is useful to note that the shared reserve system does not require all new currencies to use the shared reserve. In fact, for local or functional currencies, there are several reasons why it would be useful to not engage in the shared-reserve model; we discuss these in Section 4.4.4. To support these currencies, we also allow for new tokens to be created with their own reserve; we call this partitioned reserves. At a high level, the mechanism works in the same manner as the single stable-value coin case, except that a third party can create the token and initiate the reserve for that token. For the partitioned-reserve case, each reserve allocations are initialized at 25% Celo Gold, 25% a local reserve currency, and the remainder the same allocations as the shared reserve.

3.4 Price Discovery and Mechanics of Reserve Asset Purchasing

The Celo protocol is implemented as a fork of Ethereum. Celo Gold is the utility token for computation on the Celo network, just as Ether is the utility token for the Ethereum network. Celo stable tokens are implemented as the equivalent of ERC20 tokens. One difference between Celo and Ethereum is that while Ether itself is not compliant with the ERC20 token standard, Celo Gold is. This allows a decentralized exchange, through smart contracts, between Celo stable value tokens as well as Celo Gold, much like 0x [21]. This allows the automatic purchasing of reserves and distribution of coins without cross-chain decentralized exchanges.

To determine the price of Celo stable currencies, the protocol will use a Schelling-point scheme amongst bonded stakeholders, with the weight of the a stakeholder's vote dependent on the amount of Celo Gold at stake and the time remaining in the bond.

Bonded stakeholders are strongly incentivized to be truthful in reporting the price of Celo stable currencies. In the near term, if they report that the price of a Celo stable currency is above the peg (when in reality it's not), they will see the value of their Celo Gold increase, since the protocol would purchase more Celo Gold, leading the price of Celo Gold to go up. However, doing so would lead the price of the Celo stable currency to go down below the peg. Long-term depegging would cause people to lose confidence in Celo, likely causing the price of Celo Gold to drop in the long term. It is therefore in the interest of the bonded stakeholders to be truthful in price reporting, optimizing for the long-term price of Celo Gold rather than the short-term price of Celo Gold.

3.5 Qualitative Stability Analysis

Intuitively, most decentrlaized multi-asset stablecoin proposals shift volatility risk from the stable coin to the other assets in the ecosystem. The primary risk to the stability of these coins is if the volatility becomes large enough that the other assets in the ecosystem default and are unable to insulate the stable coin.

In the elasticity-enhanced version of the Celo protocol [9], the following three things would need to occur at the same time for that to happen. First, there would need to be a widespread perception that there will be a long-term contraction in aggregate demand of the currencies supported by the protocol, causing the bond markets to freeze up. Second, the reserve ratio would need to be low enough so that the reserves cannot contract the currencies in circulation. And finally, there would need to be a large-scale run on the currencies. We address these one by one:

Potential for a Prolonged Contraction in Demand. If there is a large contraction in demand, and too few people who believe in the long-term growth in demand to buy bonds, the protocol would then rely on the reserves to handle the remaining contraction. If the reserve ratio remains above 1, the protocol can handle any contraction in demand, including a contraction to zero demand for all currencies supported by the protocol. Even at times where there are fractional reserve ratios, the protocol is able to handle large and prolonged contractions.

[9] The analysis for the base case is a simplified version of this. And a more detailed, quantitative analysis, is given in [7].

Potential for the Reserve Ratio to go Meaningfully Below 1. The reserve ratio can dip meaningfully below 1 if there is a black swan event on the underlying reserves. If the reserve ratio does fall below 1, there are several mechanisms to push it higher. First, the transfer fee on Celo Gold would increase, bolstering the reserves. Second, released amounts of Celo Gold will start to go towards bolstering the reserves. Third, if demand for Celo's stable currency is growing, new reserves will be purchased at a 1:1 ratio with the new coin that is created, pushing the reserve ratio up towards 1.

Potential for a Run on the Currencies. The reserve ratio going below 1 is not in itself a cause for concern. The bigger issue is if there were a large-scale run on the stable currencies at a time when the reserve ratio is meaningfully below 1 and has not had the time to recover through the mechanisms of reserve appreciation, block rewards, and transfer fee.

Here, there are two mechanisms that are helpful. First, if potential bondholders believe that there will be growth in demand for the contracting stable currency in the future, they will be incentivized to counteract a run by buying up the stable currency to purchase high-yield bonds. If they do so, much of the contraction can be handled by bond sales before needing to go to the reserves at all, thus allowing a fractional reserve to handle the remainder.

Second, if the run on one of the stable currencies is happening at the same time as growth in demand for another, then the two will balance each other out as described in Section 3.3.

Risk. This is not to say that the Celo stable value currencies are risk-free. If there is a broad-based black swan event across the reserve currencies and a concurrent run on all Celo stable currencies, as well as a broad-based loss of faith in the long-term demand for Celo stable currencies, the contraction in demand would have to be handled by a weaker bond market, a more limited reserve, and the bolstering of the reserves through transaction fees and block rewards denominated in a weaker Celo Gold. These are not uncorrelated events, and carry with them a risk of depegging. However, under these conditions, any stable value currency proposal would also lose value, as would almost all variable value cryptocurrencies.

Our aim, therefore, is to minimize volatility, not to eliminate it entirely. Our general point of view is that there are several use cases for denominating some spending in a stabilized value cryptocurrency even if it's not risk-free (for example, in markets in which the fiat currency itself is not stable, or in cases where electronic payments cannot be made in fiat because of lack of access to banking, etc.).

Our more detailed analysis, with simulations across a range of market assumptions are given in [**?**].

4 Governance and Incentives

A primary incentive mechanism in Celo is the distribution of block rewards, which are allocated to the various contributors to the system – those contributing to the robustness of the reserves, those taking on risk in the case that there is a contraction, those using the protocol as their means of payment, those inviting others to use the protocol, those who maintain the protocol (by validating transactions, verifying users, and participating in the Schelling-point price discovery mechanism), and those who improve the protocol.

Outside of block rewards, several other incentive (and disincentive) mechanisms exist – for example, the appreciation of Celo Gold, the Celo Gold transfer fee, and the bond payouts. These all serve to incentivize behavior aligned with the long-term stability of the coin. We describe these below.

4.1 Bolstering Reserves and Contracting Stable-Value Currency Supply when Needed

The two parties that play a role in bolstering the reserves and contracting the stable-value currency supply when needed are: Celo Gold holders, and bond holders. Celo Gold holders, through their purchase of Celo Gold, give the initial value to Celo Gold and introduce other crypto assets into the reserves. Further, Celo Gold holders bear some risk in the case of contracting supply or a dip in the reserves: transfer fees are imposed if the reserve ratio goes below the target reserve ratio, and the value of Celo Gold may go down if there is a prolonged contraction in demand for Celo stable currencies.

Celo Gold holders are rewarded for playing these roles in two ways: first, as there is greater demand for Celo stable currencies, there will be more protocol-directed purchases of Celo Gold, increasing the demand for fixed supply. If this continues, over time, the value of Celo will likely appreciate. Second, if the reserve ratio is greater than the target reserve ratio, Celo Gold holders who have a long-term stake in Celo Gold are rewarded with a portion of the block rewards (provided that they are participating in consensus on transaction validation, sending verification messages when selected, and participating in Schelling-point voting for price discovery). These rewards are paid in proportion to the amount of Celo Gold at stake, and the time remaining in the stake[10].

4.2 Increasing User Base and Usage of the System

Active users (people who use the payments protocol, participate in block rewards through the mobile wallet, and maintain a nominal bonded stake in Celo Gold) are rewarded through block rewards. In effect, this reduces transaction fees for active users. One can even imagine a scenario in which these block rewards are issued by waiving transaction fees for a certain number of transactions in the stable currency per unit time, implemented through part of the block reward going to paying transaction fees of users, set at a rate to ensure a certain transaction speed, and prioritized based on the amount of their bonded stake.

The protocol waiving transaction fees for such users is analogous to banks waiving fees for customers that maintain a minimum balance. In practice, it encourages circulation of the stable currencies, and more fairly distributes the block rewards amongst all active users, not just through a proof-of-stake election protocol or a random selection of stakeholders.

4.3 Maintaining the System

The maintenance of the system requires a broad base of participants, who participate in the consensus scheme, send verification messages, and participate in the Schelling-point scheme for price discovery. The block rewards that go to the bonded stakeholders above serve as the incentive for system maintenance, since bonded stakeholders need to participate to receive their rewards.

4.4 Improving the Protocol

And finally a continuously evolving protocol requires incentives, and a governance scheme, for improving the protocol.

4.4.1 Technical Improvements

For technical improvements to the protocol, anybody may pay a nominal amount of Celo Gold to make a technical proposal, with a proposed fee-for-implementation, on a monthly cycle [11]. Proposals will be voted on by bonded stakeholders, similar to the voting scheme with Dash's masternodes [8], with their votes weighted by the amount of their stake and the time remaining in their bond. Funds that are not allocated in a particular cycle are added to the reserves.

The constant funding stream for development and marketing gives incentive for large numbers of developers to work on the protocol, even if they don't own much Celo Gold.

4.4.2 Introducing Regional Currencies and Broadening the Reserve Base

Over time, it would also improve the protocol to introduce more stable value currencies, and to broaden the reserve holdings. If new stable value currencies are introduced appropriately, they can increase the usefulness of the protocol, increase long-term growth in coin demand, and reduce aggregate demand

[10] In the elasticity-enhanced version of the protocol, bond holders play a role in contracting the stable value currency supply by, effectively, loaning Celo Dollars to the reserves during times of contraction. They are rewarded for this through a bond payout mechanism similar to [1], in which they are repaid at par on a first-in-first-out basis at a time of expansion.

[11] This mechanism can also be applied to other types of proposals, for example for marketing proposals.

volatility. And if new crypto-assets are chosen appropriately, they can decrease reserve volatility. Both of these have the effect of further stabilizing the coins supported by the protocol. The governance procedure for introducing these is similar to the governance around technical improvement.

Each month, any Celo Gold holder may stake a certain amount of Celo Gold to make a proposal on introducing a new stable value currency (by specifying a peg). Bonded Celo Gold holders vote in proportion to the amount of Celo Gold they own and the amount of time remaining in their bond. If a certain vote threshold is passed, a new stablecoin is introduced on the shared reserve.

Similarly, any Celo Gold holder may stake a certain amount of Celo Gold to make a proposal on introducing a new crypto asset to the reserves (by specifying a suggested percentage of future reserve purchases to be allocated to that asset). Bonded Celo Gold holders vote in proportion to the amount of Celo Gold they own and the amount of time remaining in their bond. If a certain vote threshold is passed, then future purchases for the reserves will include the new crypto-asset with an allocation given by the median percentage of all votes (with the allocation of all other assets being diluted pro-rata).

The criteria by which these proposals should be evaluated is the extent to which they would increase in the long-term stability of the stable currencies. Introductions of crypto-assets to the reserves that increase the expected appreciation of reserves and decrease the volatility of the reserves would have positive benefits to the long-term coin stability. Introductions of new stable-value coins that increase long-term aggregate coin demand and decrease the possibility of an aggregate crash in coin demand also increase the stability characteristics of the coin. All of these changes also increase the long-term value of Celo Gold. Therefore, bonded Celo Gold holders acting solely out of self-interest will vote in ways that increase long-term coin stability.

4.4.3 Futarchical Governance

It is possible that in the future, we introduce prediction markets as a supplemental form of governance – where prediction markets will also weigh in on whether a change in the composition of the reserves or the composition of stablecoin portfolio would increase or decrease long-term coin stability. It is even possible to have the prediction markets serve directly as the voting mechanism, in a futarchical governance paradigm [11]. While this is possible, we would be unlikely to do this in the near term, to avoid unintended side effects.

4.4.4 Partitioned Reserves

The introduction of a new new local currency does not need to go through the governance process if it is not backed by the shared reserve. One can introduce a new local currency backed by its own reserve, with its own affiliated local reserve currency, similar to the single Celo Dollar and Celo Gold case. In these cases, the default reserve would include in its reserve a basket of diversified crypto assets that includes Celo Gold, the local reserve currency, Celo Dollars, and others.

Doing so opens many possibilities. First, these local protocols may choose to distribute some of the local reserve currency to all local inhabitants, effectively creating a social dividend that allows local residents to benefit from the increased adoption of a local currency.

These local protocols may also choose to implement the transfer fee in a different way; rather than having the transfer fee payable in the local reserve currency when the reserve ratio is low, they may choose to bolster the reserves by issuing the fee directly on the local stable currency, at regular intervals rather than just when the reserve ratio is lower than the target reserve ratio. This implementation of demurrage has the effect of bolstering the reserves and encouraging circulation of the local means-of-payment currency, at the expense of giving people a moderate incentive to switch out of the currency when possible. Despite this drawback, the literature on demurrage (see, for example, [9, 15]) suggests that more experiments with demurrage are useful.

And finally, as more assets get tokenized in the future, the partitioned reserve mechanism allows for the reserves to include real assets. This is helpful from a stability perspective, and also allows for natural-capital-backed means-of-payment currencies (for example, currencies backed by forestland), where the growth in demand for those currencies will increase the amount of natural capital backing them. For a detailed discussion of natural-capital-backed currencies, see [9].

5 Conclusion

We have introduced a protocol for social payments, called Celo. Celo combines an address-based encryption protocol that allows a sender to use a phone number or email address directly as a public key, with a reserve-backed protocol to minimize volatility through an elastic supply rule. Together, these allow for a more seamless experience using cryptocurrencies as a means of payment. Further, they enable a monetary ecology that includes local and regional currencies, social dividends, demurrage-charged currencies, and in the future, natural-capital-backed currencies.

References

[1] Nader Al-Naji, Josh Chen, and Lawrence Diao. Basis: A price-stable cryptocurrency with an algorithmic central bank. 2017.

[2] Ferdinando M Ametrano. Hayek money: The cryptocurrency price stability solution. 2016.

[3] Eli Ben Sasson et al. Scalable, transparent, and post-quantum secure computational integrity. 2017.

[4] Dan Boneh and Matt Franklin. Identity-based encryption from the weil pairing. In *Advances in Cryptology—CRYPTO 2001*, pages 213–229. Springer, 2001.

[5] Benedikt Bunz et al. Bulletproofs: Efficient range proofs for confidential transactions. 2017.

[6] Sanjit Chaterjee and Palash Sarkar. *Identity-Based Encryption.* Springer, 2011.

[7] Roman Croessman et al. An analysis of the stability characteristics of Celo. 2018.

[8] Evan Duffield and Daniel Diaz. Dash: A privacy-centric crypto-currency, 2014.

[9] Charles Eisenstein. *Sacred economics: Money, gift, and society in the age of transition.* North Atlantic Books, 2011.

[10] Shafi Goldwasser, Silvio Micali, and Charles Rackoff. Non-interactive zero-knowledge proof of knowledge and chosen ciphertext attack. 18:186–208, 1989.

[11] Robin Hanson. Futarchy: Vote values, but bet beliefs. 2000.

[12] Daira Hopwood, Sean Bowe, Taylor Hornby, and Nathan Wilcox. Zcash protocol specification, version 2018.0-beta-20. 2018.

[13] Sepandar Kamvar. *Numerical Algorithms for Personalized Search in Self-Organizing Information Networks.* Princeton University Press, 2009.

[14] Sepandar Kamvar, Mario Schlosser, and Hector Garcia-Molina. The eigentrust algorithm for reputation management in peer-to-peer network. In *Proceedings of the 12th international conference on World Wide Web. ACM.* ACM, 2003.

[15] Bernard A Lietaer. *Mysterium Geld: Emotionale Bedeutung und Wirkungsweise eines Tabus.* Riemann, 2000.

[16] Bernard A Lietaer. *The future of money: A new way to create wealth, work and a wiser world.* Century, 2001.

[17] Robert Sams. A note on cryptocurrency stabilisation: Seigniorage shares. Technical report, Working paper, 2015.

[18] Adi Shamir et al. Identity-based cryptosystems and signature schemes. In *Crypto*, volume 84, pages 47–53. Springer, 1984.

[19] Maker Team. The dai stablecoin system. 2017.

[20] Lloyd Trefethen and David Bau. *Numerical Linear Algebra.* SIAM, 1997.

[21] Will Warren and Amir Bandeali. 0x: An open protocol for decentralized exchange on the ethereum blockchain, 2017.