Money Creation With Elastic Supply Via Trust And Blockchain Assets In Global Digital Peer-to-Peer Environment

kushti
kushti@protonmail.ch

No Institute Given

Abstract. In this paper we introduce a blockchain-based protocol to create money in self-sovereign way via trust or collateral. The protocol allows for elastic money creation in peer-to-peer environment, where there is no party which can enforce for everyone rules for money creation and acceptance. Thus acceptance of notes created or signed by other peers made an individual choice. Similarly to spender-signed currencies, every spender is signing and backing every note he spends. Every signature is associated with a reserve, which can be made of any blockchain token, or be empty. A note may be redeemed against any reserve of its signers, and then re-redeemed against a reserve associated with earlier signature.

We have implemented our proposal in software in form of a payment server, supporting flexible rules for notes acceptance, based on blacklist, whitelist, collateralization ratio. We have shown how to implement some known monetary systems, such as a local exchange trading system (LETS), and how to do mutual credit clearing by using proposed stack made of the on-chain protocol and offchain software.

1 Introduction

Currently, most of monetary value is created by private banks (often, offshore banks as in so-called eurodollar system [1]), following central banks requirements. As an alternative, starting with Bitcoin [2] launch in 2009, a lot of cryptocurrencies and applications on top of public blockchains are experimenting with algorithmic money issuance. As another option, we also have alternative (usually, local), monetary systems, such as LETS (local exchange trading systems [3]), timebanks, local government currencies (such as famous Woergl demurrage-based local currency back in 1930s [4]), and so on.

Control in traditional fiat monetary systems is possessed by big players (with rich getting richer effect) creating money in non-transparent ways (especially when done in offshore circuits), on the other side, fiat monetary systems (in opposite to commodity money used before fiat, as well as alternative monetary systems) have best supply elasticity. Proof-of-work cryptocurrencies (and, sometimes, other tokens on top of public blockchains) have strict and publicly known emission schedule set in an algorithm for computers, which is making them perfect digital commodity assets, on the other hand, supply is disconnected from economic activity and so not elastic. Local currencies usually considered more fair in segniorage distribution than fiat currencies, they are often successfully boosting local economies, but, in opposite to fiat and crypto-currencies, they are not global and rarely surviving in the long term due to loss of active supporters. They are also usually hard to get into for external actors.

In this work, we propose a new global peer-to-peer network based money system, with decentralized issuance, individual acceptance rules and elastic supply, called ChainCash. A digital banknote in ChainCash can be created by any party and collectively backed by the means of collateral and trust. Collateral for a ChainCash note comes from reserves network peers may have, and on spending a note, a peer is attaching its reserve to collective backing. At the same time, a newly issued note could be accepted by a peer without any backing provided, for example, if such a note is issued by a friend or a trusted charity. Every peer in the system is having own individual rules for accepting notes (widely accepted standards may exist at the same time), which provides basis for elasticity of supply. We are providing details in the next section.

The paper is organized as follows. In Section 2 we describe design of the system. Section 3 provides details on implementation. In Section 4 we outline possible applications on top of ChainCash. In Section 5 we provide initial analysis of advantages and drawbacks of the proposal.

2 Design

We consider money here via its medium-of-exchange property [5]. For existing currencies, there are usually many options to represent value to exchange, such as coins, paper or plastic banknotes, digital records in different ledgers, etc. For ChainCash, we define money as a set of digital notes, each has arbitrary nominal. Some existing widely recognizeable unit-of-account is used to represent value, for example, a milligram of gold.

We consider that an economy is consisting of known agents $a_1, ..., a_n$. Then we can define medium-of-exchange property of money via a set of agents accepting monetary objects (i.e. notes). Usually, set of agents accepting some kind of money (e.g. local or foreign currency) is the same for every monetary object (e.g. a note), so an agent is accepting any note as a mean of payment, or rejecting it. In opposite, for ChainCash money, similarly to spender signed currency in [6], the set is individual for a note, so when agent a_i sees a note n, it applies its personal predicate $P_i(n)$ to decide whether to accept or reject the note.

How ChainCash notes are different from each other then? Every note is collectively backed by all the previous spenders of the note. And every agent may create reserves to be used as collateral. When an agent spends note, whether received previously from another agent or just created by the agent itself, it is attaching his signature to it. A note could be redemeed at any time against any of reserves of agents previously signed the note. However, any agent after the first one in signatures chain is getting redemption receipt which is indicating debt of previous signers before him, and then he may redeem the receipt against a reserve of any previous signer, with a new redeemable receipt being generated, until the first signer is reached. Also, redemption fee should be paid from the note value, the fee is incentivizing both reserves provision and also using the notes instead of redeeming them. The protocol does not impose collateralization requirements, it is allowed for an agent to issue and spend notes with an empty reserve even. It is up to agent's counter-parties then whether to accept and so back after spending an issued note with collateral or agent's trust or not.

As an example, consider a small gold mining cooperative in Ghana issuing a note backed by (tokenized) gold. The note is then accepted by the national government as a mean of tax payment. Then the government is using the note (which is now backed by gold and also trust in Ghana government or maybe some national reserve, so convertible to Ghanaian Cedi as well) to buy oil from a Saudi oil company. Then the oil company, having its own oil reserve also, is using the note to buy equipment from China. Now a Chinese company has a note which is backed by gold, oil, and Cedis. It could be hard for Chinese company to redeem from a small cooperative in Ghana, so it can redeem from Ghana government, and the government may re-redeem its receipt from the cooperative.

Agent's note quality estimation predicate $P_i(n)$ is considering collaterals and trust of previous spenders. Different agents may have different collateralization estimation algorithms (by analyzing history of the single note n, or e.g. all the notes issued by previous signers of n, other options are also possible), different whitelists, blacklists, or trust scores assigned to previous spenders of the note n etc. So in general case payment sender first need to consult with the receiver on whether the payment (consisting of one or multiple notes) can be accepted. However, in the real world likely there will be standard predicates, thus payment receiver (e.g. an online shop) may publish its predicate (or just predicate id) online, and then a payment can be done without interaction needed to check shop's acceptance policy.

From desiderata above, we may describe the protocol with three basic entities, a reserve (an agent is associated with its reserve, possibly empty), a note, and a receipt. Possible actions involving those entities are deposit into reserve, note redemption, receipt redemption, and note spending. It is possible to witdraw funds from a reserve by simply issuing a note, then spending with a signature associated with the reserve, and then redeeming the note.

3 Implementation

We propose to implement ChainCash monetary system on top of a public blockchain as:

- a blockchain provides an instant solution for public-key infrastructure
- public blockchain allows for a global ledger solution with minimal trust assumptions [7]. As a consequence, global public ledger allows for simple analysis of notes in existence.

– smart contracts minimize trust issues in payment execution and redemption. If native blockchain currency and trustless derivatives on top of it (such as algorithmic stablecoins) are used in reserves, trust issues in redemption could be eliminated at all. If tokenized real-world commodities and fiat backed stablecoins (such as USDT) are used in reserves, redemption could not be completely trustless (as smart contracts do not have power off the chain), but at least there is transparent accounting in on-chain part of redemption

For efficient implementation, it is natural to use a blockchain with extended UTXO transactional model [8], to have one reserve per unspent transaction output (UTXO), to avoid having global state maintained on the blockchain, and notes progression is well described by underlying UTXOs progression also. Another feature which is critical for feasible implementation is possibility to manage signatures chain efficiently. We use Ergo as a blockchain to implement ChainCash as UTXO transactional model as well as AVL+ trees support are making notes and reserves implementation feasible.

For blockchain-based ChainCash implementation, we consider implementation of the following two main parts:

– contracts for notes, reserves, and redemption receipts. Here, for now we consider on-chain contracts as the most straightforward option. Then we may consider more scalable options, such as having reserves (and maybe receipts) only on chain, and have notes making progress on a side-chain or off-chain (on top of some Layer 2 solution)
– client software (which we refer to as ChainCash Server), which is interacting with the blockchain (in future, possibly, also a p2p network where notes are making progress off-chain). This software is implementing a_i agent's functionality from the Section 2, including note quality estimation predicate $P_i(n)$. For that, the client may potentially track all the reserves and notes on the blockchain. Client's $P_i(n)$ may be configured via whitelists, blacklists, collateralization requirements provided in a configuration file. ChainCash Server can be seen as a self-sovereign digital bank in peer-to-peer free banking system, issuing own private money using common unit-of-account.

Three on-chain contracts are implemented, namely, reserve, note and receipt contracts. Reserve contract locks ERG native tokens on top of Ergo blockchain and allow to redeem native or custom tokens when a note is presented. Note contract ensures that a note (UTXO) under its control has proper history, that is, on every spending a valid signature of corresponding reserve owner is added. It is also allowing for a note to be split into two parts (payment and change), as well as note redemption. On redemption, where both reserve and note contracts are involved, an output with receipt contract is created, which contains history of ownership copied from the note input, as well as position of reserve redeemed in ownership chain and note's value. With receipt it is possible then to redeem againt an earlier reserve (reserve contract allows for that).

Basic contracts implementation described is good for starters, but can be extended in many ways. We note that it is possible to add new features without need for the whole network to update. New features, such as new reserve and note contracts, can be proposed in form of CCIPs (ChainCash Improvement Proposals). ChainCash Server may support new features, in particular, new forms of notes. If client is asked to accept a note with unknown contract, or a note backed by unknown contract, it is just refusing to accept the note.

On-chain contracts [9] and reference server implementation [10] are open-sourced and available under public domain license.

4 Applications

In this section, we show how ChainCash can be used as a foundation for other monetary systems, by providing some examples. In particular, we provide details on how to implement popular community currency systems, such as a local exchange trading system [3], and then a timebank [11] can be implemented as well, by using time as unit of account in a local exchange trading system. Then we show how a local currency can be created. We note that more currency systems can be implemented, for example, as ChainCash is deriving from spender-signed currency concept, such currencies, for example, iWat [6] and its variations [12,13,14] can trivially be built on top of it. For efficiency of ChainCash, it is important to reduce the aggregate indebtedness in the system, thus we describe how multilateral trade-credit set-off can be done in Section 4.3.

4.1 Local Exchange Trading System

Local exchange trading system (LETS) [3] is a form of non-collateralized community currency, where any member of a community can issue an IOC (I Owe Community) note and pay with it within the community. To implement a local exchange trading system on top of ChainCash, every LETS member needs to whitelist everyone else, so they will accept notes of each other regardless reserves backing the notes, and thus LETS can create money within the community (the LETS circle) with no limits (or with limits the community agreed on). On the other hand, unlike traditional LETS, notes can circulate outside the LETS circle easily, if there are actors willing to accept notes from community. Implementations may vary from LETS members whitelisting unconditionally only notes issued by other members to members whitelisting notes ever signed by LETS members.

4.2 Local Currencies

A local or even national government may issue notes and enforce their acceptance within its jurisdiction by enforcing economic agents to accept notes issued or spend by the government. As well as in a LETS implementation, enforced acceptance rules may vary, for example, there could be a limit for amount of money a government may create.

Often local currencies are introducing redemption fee, to promote local usage. In ChainCash, similar goals can be achieved via modifying the reserve contract in a way that non-locals need to pay redemption fee while locals need not, alternatively, the note contract could be modified in a way that spending to non-local addresses incurs a fee. Local currencies are often associated with demurrage, after well-known Woergl experiment [4]. Demmurage could be implemented by modifying note contract.

4.3 Multilateral Trade-Credit Set-off

Multilateral Trade-Credit Set-off [15] (MTCS) is a technique which allows invoices in closed loops to be cleared against one another. In ChainCash, it is possible to clear mutual debts by just burning atomically notes backed by counterparties in a single transaction. This will allow them to issue more notes after.

MTCS can be used along with LETS for multiplying economic activity from charity. One of possible options is as follows. Charity funds can be put in a reserve associated with a public key A. Then a local trade exchange system for a community in need is formed, and it is not including A. MTCS takes place regularly, to reduce total amount of debt created in the community. Only after enough time (and maybe some additional checks) a note can be accepted by A, then A is sending it back to a LETS member which sent the note to A, and the note can be redeemed against the reserve by the LETS member. Thus charity funds are used to cover trade disbalance in the community only, which, in many cases, could be much more efficient than other options of helping communities in need.

5 ChainCash Advantages and Drawbacks

In this section, we are providing some thoughts on possible advantages and drawbacks of ChainCash. Note that practice can show completely different picture from what we are providing here (as often happens).

At the moment of writing this paper, we can highlight following advantages:

- ChainCash is unique, to the best of our knowledge, framework, where trust and backing with collateral are seamlessly and transparently combined in money issuance.
- unlike native cryptocurrrencies and algorithmic stablecoins, ChainCash provides elasticity of supply without enforcing individual users to accept notes of lower quality - it is always up to users what to accept.
- as Section 4 shows, a variety of known monetary systems can be built on top of ChainCash. This possible foundational nature of the proposal worth further research.

At the same time, at this moment the proposed design has following drawbacks:

- ChainCash notes are non-fungible, while they share the same unit-of-account, each note has unique backing. This prevents ChainCash usage in many DeFi applications, such as liquidity pools, existing lending pools etc. We note that, similarly, DAI [16] stablecoins issued against CDPs (collateralized debt posistions) with different level of collateralization also should be priced differently. And like a DAI protocol is assigning the same price to DAIs of different quality, there could be index fund like services combining notes of certain quality, by e.g. providing index fund tokens in exchange for such notes, and the tokens can be used then freely in DeFi services.
- There is no privacy in ChainCash payments now. This topic is fully left for further research.

References

1. F. Machlup, “Euro-dollar creation: A mystery story,” *PSL Quarterly Review*, vol. 23, no. 94, 1970.
2. S. Nakamoto and A. Bitcoin, “A peer-to-peer electronic cash system,” *Bitcoin.– URL: https://bitcoin.org/bitcoin.pdf*, vol. 4, no. 2, 2008.
3. C. C. Williams, “The new barter economy: an appraisal of local exchange and trading systems (lets),” *Journal of Public Policy*, vol. 16, no. 1, pp. 85–101, 1996.
4. M. Unterguggenbercer, “The end results of the woergl experiment.,” *Annals of Public and Cooperative Economics*, vol. 10, no. 1, pp. 60–63, 1934.
5. N. Kiyotaki and R. Wright, “On money as a medium of exchange,” *Journal of political Economy*, vol. 97, no. 4, pp. 927–954, 1989.
6. K. Saito, “Peer-to-peer money: Free currency over the internet,” in *Web and Communication Technologies and Internet-Related Social Issues—HSI 2003: Second International Conference on Human. Society@ Internet Seoul, Korea, June 18–20, 2003 Proceedings 2*, pp. 404–414, Springer, 2003.
7. “Know your assumptions.” `https://www.ergoforum.org/t/know-your-assumptions/4198`. Accessed: 2023-01-30.
8. M. M. Chakravarty, J. Chapman, K. MacKenzie, O. Melkonian, M. Peyton Jones, and P. Wadler, “The extended utxo model,” in *Financial Cryptography and Data Security: FC 2020 International Workshops, AsiaUSEC, CoDeFi, VOTING, and WTSC, Kota Kinabalu, Malaysia, February 14, 2020, Revised Selected Papers 24*, pp. 525–539, Springer, 2020.
9. “Chaincash contracts.” `https://github.com/ChainCashLabs/chaincash/tree/master/contracts/onchain`. Accessed: 2023-11-11.
10. “Chaincash server.” `https://github.com/ChainCashLabs/chaincash-rs`. Accessed: 2023-11-11.
11. R. McQuaid, S. Bond, and B. Christy, “A review of local exchange and trading schemes (lets) and time banks in scotland,” 2004.
12. K. Saito, “Wot for wat: Spinning the web of trust for peer-to-peer barter relationships,” *IEICE transactions on communications*, vol. 88, no. 4, pp. 1503–1510, 2005.
13. “Multiplication over time to facilitate peer-to-peer barter relationships,” in *16th International Workshop on Database and Expert Systems Applications (DEXA’05)*, pp. 785–789, IEEE, 2005.
14. K. Saito, E. Morino, and J. Murai, “Reduction over time to facilitate peer-to-peer barter relationships,” *IEICE TRANSACTIONS on Information and Systems*, vol. 89, no. 1, pp. 181–188, 2006.
15. E. Bottazzi, C. N. Ngo, and M. Tsutsumi, “Multilateral trade credit set-off in mpc via graph anonymization and network simplex,” *Cryptology ePrint Archive*, 2024.
16. B. Woltzenlogel Paleo, “Stablecoin,” in *Encyclopedia of Cryptography, Security and Privacy*, pp. 1–5, Springer, 2023.