The Rainbow Network: An Off-Chain Decentralized Synthetics Exchange

Dan Robinson, Paradigm

March 2019
WORKING DRAFT

Abstract

This paper presents the Rainbow Network, a design for an off-chain non-custodial exchange and payment network supporting any assets for which two parties can agree on a price oracle. The Rainbow Network allows a user to trade, borrow, lend, and make payments in synthetic assets, entirely off-chain, while having only one on-chain payment channel collateralized by a single asset.

1 Introduction

This paper introduces a design for a non-custodial off-chain exchange, built on top of a new primitive called **Rainbow channels**. Rainbow channels are an extension of payment channels in which the participants can hold synthetic balances in *any asset*, rather than just the asset that is used as the collateral for the channel. When the channel is closed, the amount sent to each participant is computed based on the current prices of the synthetic assets in the channel.

A Rainbow channel can support long and short positions in any asset for which the participants can agree on a price oracle, and can support as much leverage as the participants are willing to accept, while only being collateralized by a single asset on the parent chain. As in any payment channels, payments and trades in Rainbow channels can happen nearly instantaneously and at essentially zero cost.

Users can execute all of their trades in a single bilateral Rainbow channel with one market maker. In order to hedge that trade, that market maker can enter into a offsetting trade in some other Rainbow channel, forming a scalable network of fully-hedged market makers: the **Rainbow Network**.

Rainbow channels can be implemented on Turing-equivalent blockchain platforms like Ethereum. A more limited and operationally-intensive variant can even be implemented on top of simple payment channels, such as the Bitcoin payment channels used in the Lightning Network.

1.1 Prior work

In most decentralized cryptocurrency exchanges, such as Uniswap [1] and 0x [2], trades are executed on a blockchain. This increases latency and requires users to pay transaction fees on every trade. Similarly, most solutions for leverage (such as dYdX [3], Dharma [4], and Compound [5]) and/or synthetics (such as MakerDAO [6]) require on-chain transactions for every state update.

Payment channel networks such as Lightning [7] and Interledger [8] can support non-custodial off-chain exchange. However, those exchanges require the user to already have channels that are collateralized by all of the assets they want to trade, and in which they have sufficient sending or receiving capacity in each of those assets. Plasma Cash [9] and related plasma constructions can help mitigate the difficulties of finding receiving capacity, but still require that senders have sufficient balances in the assets that they wish to send or sell. Additionally, these constructions only support assets that are already held on a blockchain, and none of them natively support leveraged trading or short-selling.

Abra [10] is a platform that offers off-chain synthetic positions in various currencies and cryptocurrencies, backed by collateral held in a single cryptocurrency. This is similar to the kinds of synthetic positions used in Rainbow channels. However, in the system described by Abra, this collateral is held in a 2-of-2 multisignature address between the platform and the user [11]. This means that each party is subject to counterparty risk with respect to their entire balance—either the user or the platform can prevent the other from withdrawing any collateral. This would make it difficult to use this model in a decentralized network, where participants would prefer to minimize the credit risk that they take on.

2 Background

2.1 State channels

A state channel is a construction in which two (or more) parties lock up some state and/or assets on some parent blockchain. The participants can make off-chain updates to the channel by signing messages committing to new states. Parties can cooperatively exit from a particular state instantly; otherwise, a party can unilaterally initiate an exit, which completes after a delay. If a party attempts to exit an outdated state, their counterparty can challenge by showing a more recent state.[1]

2.2 Simple payment channels

A payment channel is a state channel where the state being managed is a ledger representing the participants' ownership in some collateral that is locked up on the parent chain.

[1] This paper mostly abstracts away the details of the underlying state channels. For a more detailed explanation of how state channels work, see the Counterfactual paper [12].

Let's suppose Alice and Bob have a payment channel with each other. This payment channel is secured by a pot of 20 ETH on the parent chain, which means that the total value that can be safely allocated between the parties is 20 ETH.

Suppose Alice currently has a balance of 5 ETH and Bob has a balance of 15 ETH. This means that Alice and Bob each have a signature from the other on a message that represents the following state:

Example A, State 1	
Recipient	**Balance**
Alice	5 ETH
Bob	15 ETH

The signed state includes the current balances of the parties, along with a nonce (which is 1 in the above example) representing the recency of the state.

If Alice wants to pay Bob 1 ETH, she and Bob can sign a state that includes updated balances, along with a higher nonce:

Example A, State 2	
Recipient	**Balance**
Alice	4 ETH
Bob	16 ETH

While Alice can use this channel to make payments to Bob in ETH, in traditional payment channel constructions, she would not be able to purchase ETH using other assets, such as USDC (a dollar-backed stablecoin [13]).

On Ethereum and other sufficiently programmable blockchains, it is possible to have payment channels that are collateralized by multiple assets. Suppose Alice had a payment channel that was collateralized by 300 USDC and 20 ETH. In this channel, as long as her current USDC balance was higher than 150 and Bob's ETH balance was higher than 1, she would be able to purchase 1 ETH for 150 USDC from Bob by updating their channel balances like so:

Example B, State 1	
Recipient	**Balance**
Alice	5 ETH 200 USDC
Bob	15 ETH 100 USDC

Example B, State 2	
Recipient	**Balance**
Alice	6 ETH 50 USDC
Bob	14 ETH 250 USDC

2.3 Payment channel networks

Payment channel networks like Lightning [7] and Interledger [8] loosen these requirements a little, by allowing the assets to be held in different payment channels, and even in channels with different counterparties.

But for Alice to purchase ETH for USDC through one of these payment channel networks, she would need to have a channel in which she has sending capacity in ETH, and one in which she has receiving capacity in USDC. Additionally, her counterparties in those channels would need to be connected by some route. And, of course, she would only be able to trade assets that were already issued on a blockchain.

3 Rainbow channels

What if Alice could temporarily "transmute" some of the ETH in her payment channel with Bob into USD, and use that to purchase ETH from Bob?

Rainbow channels are an extension of payment channels that allow parties to enter into *synthetic positions.*

Rainbow channels can be implemented on any blockchain with Turing-equivalent smart contracts. It is even possible to implement a limited version of Rainbow channels on Bitcoin, as described below in section 3.8.

3.1 Turning gold into lead

Suppose Alice has an ETH payment channel with Bob, in which she has a balance of 15 ETH and Bob has a balance of 5 ETH.

Example C, State 1	
Recipient	**Balance**
Alice	5 ETH
Bob	15 ETH

Alice wants to buy 5 ETH from Bob for 750 USD (with an implied price of 150 USD/ETH). They can update the balances of their channel as follows.

Example C, State 2	
Recipient	**Balance**
Alice	10 ETH -750 USD
Bob	10 ETH 750 USD

This update happens entirely off-chain, between the parties. The underlying collateral on the parent chain is still 20 ETH. So how can this channel be settled? The key is that the balances can be settled entirely in ETH, *based on the price of the assets at the time the exit is completed.*

If the channel is exited while the price is still 150 USD/ETH, then Alice will receive 5 ETH and Bob will receive 15 ETH:

Example C, State 2, Settling at 150 USD/ETH			
Recipient	**Balance**	**Value**	**Exit**
Alice	10 ETH -750 USD	10 ETH -5 ETH	5 ETH
Bob	10 ETH 750 USD	10 ETH 5 ETH	15 ETH

Alternatively, if the channel is exited when the price is 300 USD/ETH, Alice will receive 7.5 ETH and Bob will receive 12.5 ETH:

Example C, State 2, Settling at 300 USD/ETH			
Recipient	**Balance**	**Value**	**Exit**
Alice	10 ETH -750 USD	10 ETH -2.5 ETH	7.5 ETH
Bob	10 ETH 750 USD	10 ETH 2.5 ETH	12.5 ETH

Finally, if the channel is exited when the price is only 75 USD/ETH, Alice will receive 0 ETH and Bob will receive 20 ETH:

Example C, State 2, Settling at 300 USD/ETH			
Recipient	**Balance**	**Value**	**Exit**
Alice	10 ETH -750 USD	10 ETH -10 ETH	0 ETH
Bob	10 ETH 750 USD	10 ETH 10 ETH	20 ETH

3.2 Price oracles

Note that this mechanism requires the existence of a difficult-to-manipulate price oracle. This is an area of ongoing research, which is far too deep to explore in this paper. For ERC20 tokens, a sufficiently hardened price feed from a decentralized exchange like Uniswap may be sufficient. For assets like USD, the parties could rely on price feeds from exchanges, or piggyback on other USD-pegged assets on the parent chain.

Critically, however, there is no need for a universally agreed-upon price feed—the parties in each channel can agree upon the price oracle to be used to settle that channel.

If parties cannot agree on a trusted price oracle, they could use the variant described below in section 3.8, which has some drawbacks but does not depend on such an oracle.

3.3 Theory

In normal payment channel designs, a particular payment channel state represents a ledger—a mapping of owners to balances. Computing how this state will settle on-chain is trivial—each user receives the exact amount of ETH specified in the channel's state.

In Rainbow channels, each state represents a *contract for difference* that can be settled at any time. Settling one of these states involves computing the total current ETH value of each of the positions (positive and negative) that each party is entitled to under the swap, which involves looking up the current price. A purchase or sale inside of a channel effectively involves cancelling that contract and replacing it with another one that with the same current economic value.

In Example C, the swap entered into between Alice and Bob is similar to a contract for difference or total return swap, with USD as the underlying reference asset and ETH as the settlement currency.

3.4 Leverage, interest, and flows

Note that in the above example, Alice is effectively *levered long* ETH. She only put down 5 ETH in capital, which was worth $750 at the time. When the price of ETH doubled to $300, her position tripled in value, to $2250 (7.5 ETH at $300 each). This is a 1.5x levered long position. In effect, Alice has borrowed $750 from Bob and used it to purchase 5 ETH from him.

These kinds of channels are similar to the collateralized debt positions ("CDPs") used in the Maker system [6]. Alice plays the role of the CDP creator who "borrows" DAI (the USD-pegged stablecoin) and trades it for more ETH. Bob plays the role of the mechanism that lent the DAI to Alice, as well as the party that sold ETH to Alice in exchange for the DAI.

In addition to a fee or spread on the initial trade, Bob might reasonably want to charge Alice interest on the borrowed USD. To support this, we can add an additional feature to Rainbow channels. In addition to understanding formulas that compute final balances based on price oracles, the settlement logic of the payment channel could also understand *flows*, formulas where the final balance depends in part on the time of the exit.

Suppose Alice agreed to pay 4% interest annually, computed in USD, without compounding (which amounts to $30 per year).

Example D, State 1, Time T	
Recipient	**Balance**
Alice	10 ETH -750 USD -30 USD/year
Bob	10 ETH 750 USD 30 USD/year

If the channel is exited six months later, Alice would have to pay $15 in interest (0.1 ETH, if the price of ETH remains at $150).

Example D, State 1, Settling at 150 USD/ETH, Time T + 6 months			
Recipient	**Balance**	**Value**	**Exit**
Alice	10 ETH -750 USD -30 USD/year	10 ETH -5 ETH -0.1 ETH	4.9 ETH
Bob	10 ETH 750 USD 30 USD/year	10 ETH 5 ETH 0.1 ETH	15.1 ETH

The formula could be tweaked to allow interest to be computed in different ways, such as having it computed in ETH rather than USD, or with compounding, or even with a floating interest rate (using some agreed-upon oracle).

This feature could be used to support arbitrary flows, not just interest payments. For example, this protocol could support subscriptions, donations, or salaries that are paid continuously until cancelled.[2]

3.5 Other assets

Example C involves a trade between USD—a synthetic asset, in this context—and ETH. However, there is no reason that Alice and Bob would be limited to trades that involve ETH. For example, if Alice wanted to buy 1 Bitcoin (BTC) for 4000 USD, she and Bob could execute the following trade:

[2]Vitalik Buterin has suggested a similar protocol for continuous payments [14]. In his proposed protocol, however, changing the payment rate would require an on-chain transaction, rather than an off-chain state channel update.

Example E, State 1	
Recipient	**Balance**
Alice	10 ETH
Bob	10 ETH

Example E, State 2	
Recipient	**Balance**
Alice	10 ETH -4000 USD 1 BTC
Bob	10 ETH 4000 USD -1 BTC

Note that at prices of 150 USD/ETH and 4000 USD/BTC, 4000 USD and 1 BTC are each more valuable than the total amount of ETH locked up as collateral! Despite this, the channel is still safely overcollateralized for both parties, because at current prices, Alice's balance of 1 BTC is exactly nullified by her balance of -4000 USD, and vice versa for Bob. This is another example of how Rainbow channels can enable leverage.

If the price of BTC falls below $2500, however, Alice's portfolio will become undercollateralized (assuming that the price of ETH has not changed). The risk of undercollateralization is discussed in greater detail in section 3.7.

Any asset for which the counterparties can agree on a price oracle can be simulated in the channel. This could include other cryptocurrencies (such as BTC), fiat currencies (such as USD, EUR), commodities (such as gold or oil), or even more exotic assets, such as prediction market shares.[3]

3.6 Other collateral

The above examples use ETH as the entire collateral for the channel, but any ERC20 token can be held as part of the collateral, as long as the parties can agree on a price oracle for it relative to ETH. Parties might sometimes find it convenient to use USDC, DAI, or another relatively stable asset as collateral for their channels.

3.7 Risk of undercollateralization

In Example C, if the price of ETH falls by 50%, Alice's balance in the channel is effectively worth nothing. If the price of ETH falls further than that, the channel would become undercollateralized—Bob would no longer be able to withdraw his USD at its current value.

[3] Prediction market shares are particularly well-suited for Rainbow channels, because their prices have a fixed upper bound, which means it is possible to fully collateralize positions in those shares. In fact, trading prediction market shares in Rainbow channels only requires the same amount of collateralization as is required to trade those shares on-chain.

Bob should therefore ensure that the value of Alice's portfolio remains high enough that volatility in her assets will not cause it to dip below $0 too quickly. Alice could increase her collateral by "topping up" the channel with additional ETH when her portfolio falls in value. If her position falls too close to $0 and she fails to top up her channel or participate in a cooperative close, Bob should initiate an exit from the channel.

Since payment channels take some time to settle (often between one hour and three days, depending on the security parameters of the parties), there is a risk that the portfolio will become undercollateralized before the exit is finalized, since it will be exposed to a long period of volatility.

If Bob wants to preserve his long and short exposure to that basket of assets, then when his counterparty's portfolio goes below 0, he can try to immediately hedge his position by entering into the same positions in some other channel (or on some exchange), exiting that hedge if the position goes back above 0. Hedging is described in greater detail in section 4.1.

3.8 Rainbow channels on Bitcoin

The above discussion assumes that we are using state channels whose settlement balances can be dynamically computed based on some price oracle. This is relatively straightforward if a layer-1 platform is Turing equivalent, like Ethereum, but may not be possible on a more constrained platform like Bitcoin.[4]

However, it is possible for parties to implement some of the functionality of Rainbow channels on top of simple payment channels, such as the Bitcoin-based payment channels that are used in the Lightning Network. The approach has somewhat more off-chain computational and communication overhead, and likely imposes even higher capital requirements, but eliminates the need for sophisticated smart contracting capabilities or a trusted price oracle.

Suppose Alice and Bob have a simple Lightning payment channel, and want enter into a position like the one in Example C. Entering into this contract does not involve updating the state of the channel. Instead, during the pendency of the contract, the parties continually recompute the current channel balance based on the new price, and update the channel state to reflect that new balance.[5] Note that Alice and Bob are not executing new *trades*—they are *settling* their existing contract to the channel.

If Bob stops participating in these updates, Alice should immediately initiate a close of the channel. When Bob does this attack, he can only steal a negligible amount of *value* from Alice, since he can only take advantage of the price movement since their latest update.[6] However, Alice does immediately lose her

[4]It might be possible to construct oracle-dependent payment channels on Bitcoin using Discreet Log Contracts [15], but any such solution would likely have astronomically high computational and communication overhead.

[5]While this does require Alice and Bob to remain online and to constantly sign new updates to their payment channel, those are similar requirements to those already imposed on routing nodes in Lightning.

[6]In other words, an attacker has the option whether to accept or reject each channel update based on the price movement since the previous update. Since updates could conceivably

exposure to the position that she entered into in that contract, instead gaining exposure to Bitcoin. This is different from fully-featured Rainbow channels, where, unless the channel becomes undercollateralized, this shift in exposure only happens at the time Alice's channel close completes (which is also the time she regains access to the collateral and can put it into another channel or trade it on-chain).

Assuming Alice wants to maintain her exposure to those assets, she can immediately "novate" that position by entering into the same position in another one of her channels. If she is able to do so efficiently, then Bob's betrayal only costs her some limited amount, based on the price movements during the short period of exposure, as well as the costs of novating the position. However, this solution does require Alice to maintain some Bitcoin in another Rainbow channel, in case she ever needs to novate a position from another channel.

4 Rainbow Network

Rainbow channels provide a powerful primitive for trading, borrowing, and lending assets with a channel counterparty. But they would still be of limited use if you could only enter into a trade when you have a channel with someone who wants to make the opposite trade. By networking these channels together, we can construct a system in which market makers can give their customers execution on arbitrary trades without taking on significant additional risk themselves.

4.1 Hedging

In Example C, Alice initiated the trade because she wanted to increase her economic exposure to ETH, relative to DAI. Bob essentially acted as a market maker, executing the trade in exchange for (presumably) some fee, spread, or interest rate.

What if Bob didn't want to change his market exposure? Bob can *hedge* this trade by executing the reverse trade (paying 750 USD to buy 5 ETH) somewhere else. He could do this on a centralized exchange, or on an on-chain exchange such as Uniswap, but a natural place to execute it would be *another Rainbow channel*.

Participants in Rainbow channels could therefore form a "hedging network", where market makers execute trades for users and then hedge and net those trades with each other, entirely off-chain. We can call this network the **Rainbow Network**, on the premise that rainbows are basically just multicolored lightning.

happen every few seconds, the value of this free option should be relatively small compared to the value of the channel. Indeed, these few seconds of optionality seem likely to be insignificant compared to the "free option" that one party gets in a multi-asset HTLC trade [16]. This does mean, though, that parties should be even more careful when entering into positions in volatile or levered assets.

There are many possible topologies for this network. One possibility is a hub-and-spoke model, where medium-size market makers like Bob would hedge their trades with end users by entering into offsetting trades with larger market makers. After netting these trades against each other, these very large market makers could hedge their own exposure by executing trades on a centralized exchange, or on-chain.

The flexibility of Rainbow channels means that currency exchange is significantly simpler in the Rainbow Network than in other payment channel networks. In a multi-asset Lightning Network, to purchase BTC for LTC, Alice would need to find a route starting with her BTC payment channel and ending with her LTC payment channel. In the Rainbow Network, rather than finding a path to a particular *channel*, the parties only need to find a path to someone willing to take a position in a particular *asset*. Additionally, unlike in Lightning, Rainbow Network trades do not require atomic updates of multiple channels—once Bob has agreed to trade with Alice, how he hedges it is not Alice's concern.

4.2 Payments

In addition to being used to buy and sell synthetic assets, Rainbow channels can easily be used like an ordinary payment channel, for making payments in any asset. If Alice wants to make a 100 USD payment to Bob, she can do so as follows:

Example E, State 1	
Recipient	**Balance**
Alice	5 ETH
Bob	15 ETH

Example E, State 2	
Recipient	**Balance**
Alice	5 ETH -100 USD
Bob	15 ETH 100 USD

Rainbow channels can also support multi-hop payments, using protocols like Lightning[7] or the Interledger Protocol. Unlike with traditional payment channels, Bob would not need to have a channel open in a particular currency in order to send or receive funds in that currency.

[7]Lightning supports multi-hop payments using a two-phase commit protocol, during which some of the assets in a channel are allocated to a hashed timelock contract (HTLC). This can be supported in Rainbow channels by allowing states to specify arbitrary smart contracts, rather than public keys, as the "recipients" of particular balances.

5 Areas for further research

5.1 Protocol specification

The above discussion informally proposes abstract features that could be supported by Rainbow channels, without attempting to specify the actual protocol.

For the state-channel functionality, it likely would be possible to implement it within a framework for generalized state channels, such as [12].

A full solution would also likely need to specify a domain-specific language for the payment channel states, which need to define a mapping from recipients to balances as a function of price (and sometimes as a function of time). It may be possible to adapt an existing protocol for on-chain margin positions, such as dYdX [3].

5.2 Other derivatives

Each Rainbow channel state corresponds to a specific kind of derivative: a swap contract that is cancellable at any time by either party. While this turns out to be an extremely flexible and powerful derivative, it doesn't come close to capturing the full range of rights that Alice and Bob could define with respect to the channel.

For example, Alice and Bob could enter into an *options contract* inside their payment channel. The option could be cash-settled (based on a price oracle) upon exercise, or it could be physically-settled, with the seller sacrificing their entire collateral if they fail to deliver the underlying asset.

As another example, the rest of this paper assumes that Rainbow channels, like typical state channels, can be closed and settled at any time by either party. This would make the synthetic "loan" in Example C different from typical loans (or CDPs), which usually are not cancellable by the lender. If Bob settled the channel early, Alice would no longer have the leveraged long exposure to ETH that she thought she had signed up for. While she could enter into a new trade, it may have a less favorable interest rate.

To allow Alice and Bob to lock in the terms of a long-term position, Rainbow channel states could allow parties to set conditions on when a channel can be closed. If such a condition were added, Bob would also want to ensure that he could initiate an early settlement (or "margin call") if the value of Alice's portfolio comes too close to undercollateralization.

On Ethereum, it may be possible to allow parties to define some of these more advanced derivatives using an existing protocol like dYdX. Supporting more complex derivatives in *Bitcoin* payment channels is likely more difficult, and is left as a subject for further research.[8]

[8] One possibility is that the parties could treat *the option itself* as the underlying asset, and constantly update their payment channel based on the option's current computed value.

5.3 Plasma

One disadvantage of the above construction is that these synthetics are tied to the relationship between Alice and Bob. By combining this mechanism with the non-fungible off-chain transfers enabled by Plasma Cash, we could allow both the synthetics and their collateral to be transferred between parties. (This section only presents a sketch, and assumes some familiarity with Plasma Cash research.)

Suppose that rather than being an ordinary channel, this was a Plasma Debit [17] channel, with Bob serving as the operator. Plasma Debit allows users to transfer their interests in payment channels to other users. Using this construction, Alice would be able to transfer her entire portfolio (including its collateral, synthetics, and debt) to another party.

Further extensions could allow any two parties—where neither has to be the plasma chain operator—to enter into a swap position on a plasma chain. We could even allow channels to be split into ranges (as is supported in Plasma Cashflow [9]). If you split off a piece of the portfolio containing synthetic USD, as well as some of the corresponding collateral, then that piece could be transferred independently of the rest. This would allow the synthetic USD in a Rainbow channel to be transferred to other parties, mostly independently of the collateral.

All of these ideas can likely be implemented within generalized plasma [18], a framework that supports custom plasma variants within a single plasma chain.

References

[1] Hayden Adams. *Uniswap*. URL: `https://docs.uniswap.io/`.

[2] Will Warren and Amir Bandeali. *0x: An open protocol for decentralized exchange on the Ethereum blockchain*. Feb. 2017. URL: `https://0x.org/pdfs/0x_white_paper.pdf`.

[3] Antonio Juliano. *dYdX: A Standard for Decentralized Margin Trading and Derivatives*. URL: `https://whitepaper.dydx.exchange/`.

[4] *Dharma: A Generic Protocol for Tokenized Debt Issuance*. URL: `http://whitepaper.dharma.io/`.

[5] Robert Leshner and Geoffrey Hayes. *Compound: The Money Market Protocol*. June 2018. URL: `https://compound.finance/documents/Compound.Whitepaper.v04.pdf`.

[6] MakerDAO. *The Dai Stablecoin System*. URL: `https://makerdao.com/en/whitepaper/`.

[7] Joseph Poon and Thaddeus Dryja. *The Bitcoin Lightning Network*. Jan. 2016. URL: `https://lightning.network/lightning-network-paper.pdf`.

[8] Interledger. *Interledger Architecture*. URL: `https://interledger.org/rfcs/0001-interledger-architecture/`.

[9] Plasma Group. *Plasma Group's Plasma Spec*. Jan. 2019. URL: https://medium.com/plasma-group/plasma-spec-9d98d0f2fccf.

[10] Daniel McGlynn. *Abra's Synthetic Currency*. Oct. 2018. URL: https://www.abra.com/blog/abras-synthetic-currency/.

[11] Daniel McGlynn. *How non-custodial wallets let you be your own bank*. July 2018. URL: https://www.abra.com/blog/noncustodialwallet/.

[12] Jeff Coleman, Liam Horne, and Xuanji Li. *Counterfactual: Generalized State Channels*. June 2018. URL: https://l4.ventures/papers/statechannels.pdf.

[13] Coinbase. *USD Coin (USDC)*. URL: https://www.coinbase.com/usdc.

[14] Vitalik Buterin. Feb. 2019. URL: https://twitter.com/vitalikbuterin/status/1093091291066294272.

[15] Thaddeus Dryja. *Discreet Log Contracts*. URL: https://adiabat.github.io/dlc.pdf.

[16] ZmnSCPxj. *[Lightning-dev] An Argument For Single-Asset Lightning Network*. Dec. 2018. URL: https://lists.linuxfoundation.org/pipermail/lightning-dev/2018-December/001752.html.

[17] Dan Robinson. *Plasma Debit: Arbitrary-denomination payments in Plasma Cash*. June 2018. URL: https://ethresear.ch/t/plasma-debit-arbitrary-denomination-payments-in-plasma-cash/2198.

[18] Plasma Group. *Towards A General Purpose Plasma*. Feb. 2019. URL: https://medium.com/plasma-group/towards-a-general-purpose-plasma-f1cc4d49c1f4.