LOOPRING
Decentralized Token Exchange Protocol
v1.3

daniel@loopring.org
jay@loopring.org
alex@loopring.org

Loopring Project Ltd
foundation@loopring.org

September 26, 2017

This document is for informational purposes only and does not constitute an offer or solicitation to sell shares or securities. Any such offer or solicitation will be made only by means of a confidential offering memorandum and in accordance with the terms of all applicable securities and other laws.

Abstract

Loopring is an open, multilateral token exchange protocol, for decentralized exchange on the Ethereum blockchain. Loopring is intended to serve as common building block with open standards, driving interoperability among decentralized applications (DAPPs) that incorporate exchange functionality. Trades are executed by a system of Ethereum smart contracts that are publicly accessible, free to use, and that any dApp can hook into.

Contents

1 Background

Blockchain[1][2] technology was created to facilitate the cryptocurrency Bitcoin[3]. It was originally intended for use as a decentralized system to enforce financial agreements[4][5]. The technology that underlies such agreements is able to spread into other transactions: trading stock, IP, purchase and sale of real estate, purchasing music, and much more. Both consortium blockchain and private blockchain have been developed and implemented during the last few years, however the value only exists within a closed set of entities. Conversely, fully public blockchain operates by having a large number of participants, resulting in trust through numbers. According to coinmarketcap.com stats, the total cryptocurrency market cap value has reached to 113 billions USD, including 32 billions USD from Ethereum[6] on June 12, 2017.

Blockchain has a massive influence across many domains, particulary in the finance sector. It is strongly believed that tokenization[7][5][2] offers many solutions across these domains. Asset tokenization can reduce cost, and globalize the asset and increase the liquidation thereof. As this technology expands, an increase in dApps will arise that require the use of these different tokens. As a result, an open standard for exchanging tokens is critical to support this open economy.

A regular exchange platform is based on peer-to-peer IOUs, and blockchain technology. Firstly, users need to deposit their money or tokens into an exchange's bank account or wallet. Then their account will be credited an IOU. Thus, users are in fact trading their IOU in the exchange. Users have to file a ticket when they want to withdraw or sell the tokens.

Multiple exchanges have faced problems in past years. In February 2014, the largest Bitcoin exchange "Mt. Gox" suspended trading, closed its website and exchange service, and filed forbankruptcy protectionfrom creditors[8]. Mt. Gox announced that approximately 850,000 Bitcoins belonging to customers and the company were missing and likely stolen; an amount valued at more than $450 million at the time. Research showed that less than 1 percent (7000 btc) of missing funds were lost due to attacks. In 2016, theBitfinex Hack occured in which $72 million in Bitcoin was stolen from customer's accounts. As such, it is evident how a lack of regulation and protection has been injurious toward Bitcoin and other cryptocurrencies across many regions. It also illustrates how centralized exchange platforms carry unavoidable risk.

Loopring epitomizes a protocol that facilitates a decentralized exchange mechanism of ERC20 tokens on the Ethereum blockchain to solve the aforementioned issues. One of the strengths of decentralization is that of mitigated risk, wherein tokens are not held by a central authority. Therefore, asset theft or confiscation is impossible, leading to greater trust between customers and exchanges at a very low cost. Additionally, this mechanism has no time or regional limitations, is highly transparent, and has traceable features. Such features lead to effecting transactions that are more liquidatable, and to minimize price spread.

2 Market and Industry

Decentralized exchange protocols built on blockchain technology do already exist. Examples of these include Ripple, BitShares, Openledger, Bancor, and 0x.

Ripple[9]is areal-time gross settlementsystem,currency exchange,andremittancenetwork operated byRipple (the company). Also called theRipple Transaction Protocol (RTXP)orRipple protocol,it is built upon adistributedopen sourceInternetprotocol known as the consensus ledger. Ripple's solution is built around an open, neutral protocol (Interledger Protocol or ILP[10]) to power payments across different ledgers and networks globally. It offers a cryptographically secure end-to-end payment flow with transaction immutability and information redundancy. Architected to fit within a bank's existing infrastructure, Ripple is designed to comply with risk, privacy, and compliance requirements.

BitShares[11][12] is an industrial grade financial blockchain smart contract platform. TheBitSharesdecentralized exchange - also known as "The DEX" is a next-generation cryptocurrency trading platform. The DEXis inherently decentralized, enabling one to trade the BitShares core token (BTS) and a range of trustless price-stable, market-pegged assets, such as bitUSD, bitCNY, bitBTC, bitGold, and others. These assets can all be traded withzerocounter-party risk, putting the user in total control of funds. However, the Bitshares project has many limitations.

The OpenLedger Dex[13] is a cryptocurrency exchange. It allows users to exchange Bitcoin into smartcoins and then withdraw the smartcoins and convert them into cash through PayPal, Ripple, or NanoCard. Additionally, Openledger relies greatly on the BitShares 2.0 platform and Graphene Toolkit's operation.

The Bancor[14][15] protocol enables built-in price discovery and a liquidity mechanism for tokens on smart contract blockchains. These "smart tokens" hold one or more other tokens in reserve and enable any party to instantly purchase or liquidate the smart token in exchange for any of its reserve tokens. This is done directly through the smart token's contract, at a continuously calculated price according to a formula which balances buy and sell volumes.

"0x"[16] is a protocol that facilitates low friction peer-to-peer exchange of ERC20[17] tokens on the Ethereum blockchain. The protocol is intended to serve as an open standard and common building block driving interoperability among decentralized applications (dApps) that incorporate exchange functionality. Trades are executed by a system of Ethereum smart contracts that are publicly accessible, free to use and that any dApp can hook into. DApps built on top of the protocol can access public liquidity pools or create their own liquidity pool and charge transaction fees on the resulting volume. However, 0x protocol has limitations including only being able to accept simple OTC orders, having an unclear competing mechanism among different exchanges, and lacking a protection mechanism for miners.

Taking in to account the advantages and limitations stated above, it is clear that centralized exchange still plays an important role in the cryptocurrency market at present. Nevertheless, Our team, inspired by both 0x protocol and payment channel, have conveied a new solution for a decentralized exchange protocol.
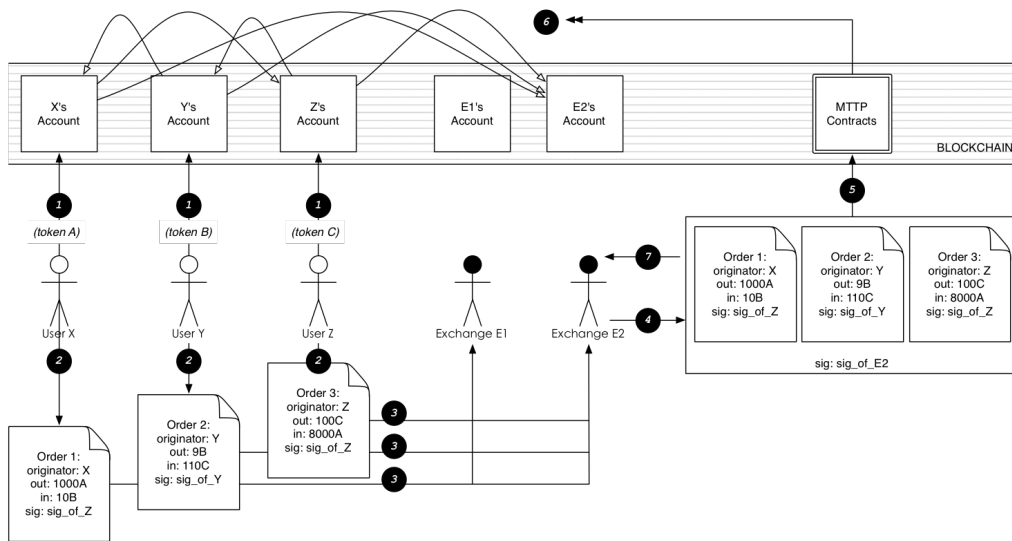
3 Design Protocol

Figure 1: Figure shows mix and match 3 orders

Figure 1 presents the general sequence of steps used for three parties transaction under Loopring:

1. User X, Y, and Z authorize the Loopring smart contract to manipulate their accounts for token trading and exchanging. From the above figure, such a contract may transfer out 1000 token A from User X's account, transfer out 9 token B from User Y's account, and 100 token C from User Z's account;

2. User X, Y, and Z place their own orders with signature using their private keys. Thus, all orders go into a medium and are ready to be exchanged - Order 1 is selling no more than 1000 token A and purchase no less than 10 token B; if the order is partially matched, then the exchange rate between A to B should be no less than 1000/10=100.00 (selling tokens divided by purchasing tokens). Furthermore, to illustrate other parameters involved in chapter 3.7;

3. User X, Y, and Z continue to send their order to one of the other multiple exchanges;

4. After the exchange received three separated orders, they will replace them into a corresponding order-book, while updating a new block and calculating each orders status to match the set order - creating a loop we delineate as a ring exchange or matching exchange. Once all the orders are confirmed and successfully mix-matched;

5. Exchange will send out a signature to the given Loopring smart contract address;

6. Loopring smart contract will verify quadruple signatures in order to verify three orders closing. If closing is failed, then the contract will be terminated (certain exchange gas cost exempt); otherwise, Loopring smart contract needs to calculate the proceeds and cost for each users, then complete the token exchange — as illustrated in the figure below. During each step, Loopring smart contract will use Loopring Registration Contract to calculate all the fees and discount before closing, the system will also need to use Loopring Stats Contract to update the database.
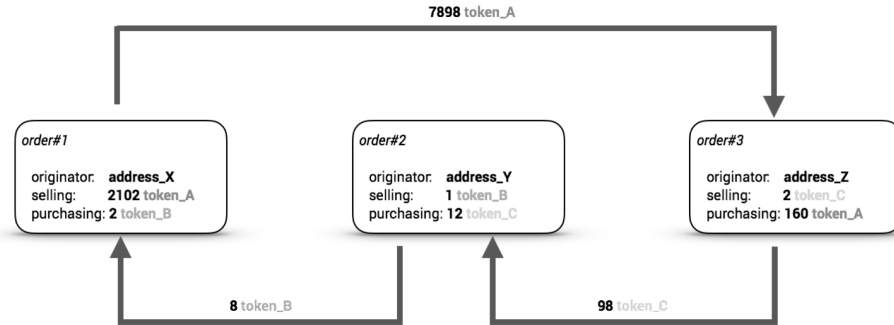

Figure 2: LoopringMatch-Ring Settlement

7. Exchange begins receiving new block and new data from the chain in order to upgrade the order-book then to mix-match new and existing orders.

3.1 Definition of Symbol

Symbols are defined as follows.

C_i: stands for the i-th token.

$O_{i \to j}$: stands for an order selling token C_i for token C_j .

$s_{i \to j}$: selling token upper limit in order $O_{i \to j}$.

$b_{i \to j}$: buying token lower limit in order $O_{i \to j}$.

$r_{i \to j}$: max exchange rate in order $O_{i \to j}$, which is $s_{i \to j}/b_{i \to j}$.

We underlined the symbols to place emphasis on their original numbers. For example $\bar{s}_{i \to j}$ and $\bar{b}_{i \to j}$ stands for the number of tokesn from the original order.

3.2 Rate Immutability

Loopring demands that the max-return exchange rate in an order stay immutable until the order is closed: $s_{i \to j}/b_{i \to j} = \bar{s}_{i \to j}/\bar{b}_{i \to j}$. This guarantees that after an order is partially filled, the remaining order still satisfies the user's original intention.

3.3 Order Reducibility

We can use token C_j to connect two orders ($O_{i \to j}$ and $O_{j \to k}$), regard it as one single order for selling token Ci for buying token C_k. we use $O_{i \to j \to k}$ to represent this order. This order $O_{i \to k}$'s properties can be calculated as:

$$s_{i \to j \to k} = min(b_{i \to j}, s_{j \to k}) \cdot r_{i \to j} \tag{1}$$

$$b_{i \to j \to k} = min(b_{i \to j}, s_{j \to k})/r_{j \to k} \tag{2}$$

$$r_{i \to j \to k} = r_{i \to j} \cdot r_{j \to k} \tag{3}$$

Here we introduce a concept of order-chain. It contains two or more orders, each orders selling token is the following orders purchasing token except the last one in the chain. Additionally, final orders purchasing token should be different from the first orders selling token (otherwise it will become a ring).

$$s_{0 \to \ldots \to n} = \begin{cases} s_{0 \to 1} & \text{as } n = 1 \\ min(b_{0 \to \ldots \to n-1}, s_{n-1 \to n}) \cdot r_{0 \to \ldots \to n-1} & \text{as } n > 1 \end{cases}$$

$$b_{0 \to \ldots \to n} = \begin{cases} b_{0 \to 1} & \text{as } n = 1 \\ min(b_{0 \to \ldots \to n-1}, s_{n-1 \to n})/r_{n-1 \to n} & \text{as } n > 1 \end{cases}$$

$$r_{0 \to \ldots \to n} = \prod_{i=0}^{n-1} r_{i \to i+1}$$

3.4 Match-Ring

Most, if not all, centralized exchanges match orders from the two sides of a trading pair. Loopring, however, involves detecting a ring of orders that may involve multiple tokens/currencies. With one order Match-Ring, multiple orders can be filled instantly.

Definition 3.1 (Match-Ring) *Let $C_0, C_1, \cdots, C_{n-1}$ be n different kinds of token, $O_{0 \to 1}$, $\cdots, O_{i \to i \oplus 1}, \cdots, O_{n-1 \to 0}$ be n orders. Those orders can form a ring for trading:*

$$O_{0 \to 1} \to \cdots \to O_{i \to i \oplus 1} \to \cdots \to O_{n-1 \to 0},$$

where n is the length of the ring, and $i \oplus 1 \equiv i + 1 \mod n$.

Once the prices match the orders under circumstance, we could start to complete trading in this circle.

3.4.1 Price

We will introduce an example for a better understanding of the price mechanism. Assume three kinds of token are C_0, C_1 and C_2, three separated orders: $O_{0\to1}$, $O_{1\to2}$ and $O_{2\to0}$. Easy to approve: if and only if $r_{0\to1} \cdot r_{1\to2} \cdot r_{2\to0} = 1$, all three orders could be filled using their respective exchange rate; If $r_{0\to1} \cdot r_{1\to2} \cdot r_{2\to0} > 1$, all these orders can be filled using a rate lower than their implicit max exchange rate. We named the first situation as `original-price matching`, the second as `discount-price matching`.

According to Loopring protocol, each order in the ring would share the same rate (price) discount. For instance, if the discounted rate is γ, then the price for each order will be: $r_{0\to1} \cdot (1-\gamma)$, $r_{1\to2} \cdot (1-\gamma)$, $r_{2\to0} \cdot (1-\gamma)$, and satisfied:

$$r_{0\to1} \cdot (1-\gamma) \cdot r_{1\to2} \cdot (1-\gamma) \cdot r_{2\to0} \cdot (1-\gamma) = 1 \tag{4}$$

We can find out:

$$\gamma = 1 - \frac{1}{\sqrt[3]{r_{0\to1} \cdot r_{1\to2} \cdot r_{2\to0}}}.$$

In the other circumstance, if transaction cross n orders, the `discount` is:

$$\gamma = 1 - \frac{1}{\sqrt[n]{\prod_{i=0}^{n-1} r^i}},$$

where r^i is the order turnover rate of i-th order. Obviously, only when the discount rate is $\gamma \geq 0$, these orders can be filled; and the i-th order's O^i actual exchange rate $\hat{r^i} = r^i \cdot (1-\gamma)$, $\hat{r^i} \leq r^i$.

3.4.2 Fill Volume

finding the lowest value order can help to figure out the fill volume for each order. For instance, if the i-th order is the lowest value order, then the number of tokens sold from each order $\hat{s}$ and number of tokens purchased $\hat{b}$ from each order can be calculated as:

$$\hat{s}^i = \overline{s}_i, \ \hat{b}^i = \hat{s}^i/\hat{r}^i, \ ;$$
$$\hat{s}^{i\oplus1} = \hat{b}^i, \ \hat{b}^{i\oplus1} = \hat{s}^{i\oplus1}/\hat{r}^{i\oplus1};$$
$$\hat{s}^{i\oplus2} = \hat{b}^{i\oplus2}, \ \hat{b}^{i\oplus2} = \hat{s}^{i\oplus2}/\hat{r}^{i\oplus2};$$
$$...$$

where $\overline{s}_i$ is the the balance left after order partially filled.

During implementation we can safely assume any order in the ring to have the lowest value, then iterate through the ring at most twice to calculate each orders fill volume.

3.4.3 Cost and Fee

Exchanges normally charge a transaction fee. For instance, we assume the fee will be calculated in Loopring token LRC, order ID is i and total fee for completing the transaction is m^i:

$$f^i = b^i \cdot m^i/\overline{b^i}$$

In order to encourage an exchange to offer best rate for users, Loopring would distribute profit from `margin` to the given exchange. as an order O^i, if price for purchasing is b^i ($b^i \leq \overline{b^i}$), then we define margin as:

$$\Delta^i = b^i \cdot r^i \cdot \gamma$$

If Loopring requires every order to set up a margin split θ^i, and minimum margin split percentage is Θ. Then order O^i should pay to exchange:

$$f^i = \Delta^i \cdot \Theta = b^i \cdot r^i \cdot \gamma \cdot \Theta$$

Income from margin among each matching trade is explicated by:

$$F = \sum_{i=0}^{n-1} b^i \cdot r^i \cdot \gamma \cdot \Theta$$

In order to encourage LRC usage, if the order has no preset token fee m^i, or $m^i = 0$, then the actual ratio is 100% regardless of the relevant hash in this order. If none of the order has set up this rate $\Theta = 100\%$, then all proceeds from the margin will go to the exchange.

In the next chapter we will introduce a token pledge policy, and explain how the smart contract will list out each exchanges depositing tokens and rank them up. It will also calculate a `mandatory discount cost` for each exchange, λ. This figure will affect the total cost. Meanwhile, an exchange can also offer a discount, η. Total cost for completion of a full trade:

$$F = (1 - \lambda) \cdot (1 - \eta) \cdot \sum_{i=0}^{n-1} (b^i \cdot r^i \cdot \gamma \cdot \Theta + b^i \cdot m^i / \overline{b^i})$$

3.4.4   Fee Discount

Loopring requires an exchange platform to offer a discount for each transaction. The discount fee is dependent upon the number of deposited token LRC. The higher the rank, the lower the charged fee. For example rank n's cost will be:

$$\lambda_n = 0.05 \cdot (\ln(n + e - 1) - 1).$$

Details below:

Deposit Ranking n	cost for discount λ
1	0%
2	1.57%
10	7.31%
20	10.39%
99	18.06%
100	18.11%
1000	29.55%
1001	30.00%*

Table 1: Deposit LRC Ranking and cost for discount

For those exchanges ranked under 1001 and those undeposited exchanges, 30% cost will apply.

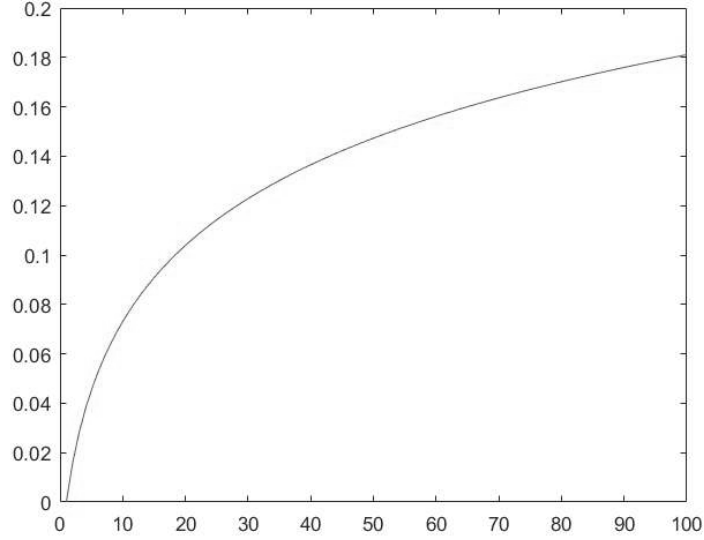Figure 3 shows, $\lambda_2 - \lambda_1 \gg \lambda_{100} - \lambda_{99}$.

Figure 3: *LRC* token deposit rank and cost for discount

3.5 Fraud and Attack Protection

3.5.1 Exchange Covered Interest Arbitrage

Loopring endevours to create a fair ecosystem and find a balance between customers (users) and exchanges. Firstly, we will explain how an exchange could archive a zero risk covered interest arbitrage.

Assume there are two orders $O_{a \to b}$, $O_{b \to a}$ that form a loop, $r_{a \to b} \cdot r_{b \to a} > 1$. An exchange can input three new orders between those two. $O_{b \to c}$, $O_{c \to d}$, $O_{d \to b}$, to create a five order loop, $r_{a \to b} \cdot r_{b \to c} \cdot r_{c \to d} \cdot r_{d \to b} \cdot r_{b \to a} = 1$. An exchange could bring the possible cost down to zero once the transaction completed, implementing zero risk covered interest arbitrage and $O_{b \to c} \to O_{c \to d} \to O_{d \to b}$. In order to stop these parameters, Loopring requires: **a verified loop cannot create a further sub-loop to continue trading**.

3.5.2 Denial-of-Service

Loopring allows exchanges to selectively handle orders. An exchange can set up their own criteria and may choose to hide or reveal these criteria. Therefor Loopring does not see denial of service as a form of unethical behavior.

3.5.3 Massive Tiny Order Attack

A user can send out a large amount of tiny orders to attack exchanges. Exchanges however, will reject most of these tiny orders because they do not yield satisfying profit when matched. As denial-of-service is not deemed as a form of attack, massive tiny order attack is not feasible.

3.5.4 Insufficient Balance

Malicious users may sign and spread out orders whose value inside the order is not zero but whose address actually has zero balance. This again is a not a good way to attack exchanges. Exchanges will monitor and notice that some orders actual balance is zero and update these orders states accordingly, then discard them.

Exchanges do have to spend time to update order status, but can also choose to minimize these effort by, for example, blacklisting some addresses and drop all related orders.

9

3.5.5 Ring Filch

A devient exchange could monitor all unconfirmed Match-Rings and broadcast the same rings with their own digital signature. We call this Ring Filch. In order to prevent Ring Filch Loopring allows exchanges to use two steps in order to submit the order:

- Submit the hash of a Match-Ring, wait for confirmation.

- Submit the ring itself.

Hash rate:

$$h = H(r, nonce),$$

where $H()$ is a one-way hash function, r is Match-Ring record. Hash Hash function contains a random number *nonce*.

3.6 Market Depth

Exchange do not need to offer market depth data. Under this ecosystem, it is possible for both single entities and corporations to possibly pool all unclosed orders into one instance of market depth data. We can ascertain trading data between any two ERC20 tokens according to the agreement in chapter 3.3.

3.7 Data Structure

All the orders can be represented by using one data structure due to adopting OTC module. This data structure contains both a digital signature and all parameters. Before the signature, the parameter data is connected from the orders into a set of data, the order's hash is calculated by using Keccak SHA3 method, and then signed by using this account's private keys with ECDSA.

```
message Order {
 address protocol;
 address owner;
 address outToken;
 address inToken;
 uint256 outAmount;
 uint256 inAmount;
 unit256 expiration
 unit256 fee;
 uint8 marginSplit;
 unit8 v;
 bytes32 r;
 bytes32 s;
}
```

Though there is no indicated price from the order, we are still able to find out through the formula: $outAmount/inAmount$ to determine the exchange rate r. The actual exchange rate must be less than r. A user-friendly exchange should allow user to input $outAmount$, $inAmount$, selling and asking price, and use any two of those numbers in order to calculate the missing $outAmount$ or $outAmount$ figure.

Actual orders can be defined in two different ways: Definition A - transaction can be completed once the number of tokens sold reaches $outAmount$; Definition B - transaction can be completed once the number of tokens purchased reaches $inAmount$; Therefore, we can setup a quote for exchange and mix-matching contract to help to define the trade. At our initial version, we would support Definition A only.

Exchange could create a Match-Ring by using this data structure:

```
message MatchRing {
  Order[] orders;
  address feeRecipient;
  unit256 additionalDiscount;
  unit256 nonce;
  unit8 v;
  bytes32 r;
  bytes32 s;
}
```

3.8 Order Status

An rrder cannot be modified once it has been signed and announced. Data will be updated on the blockchain once the smart contract finds the matched order. Thus *inAmount* and *outAmount* will modified correspondingly with the updated price. If *inAmount/outAmount* shows 0, it means that the order has been fully closed. For example, if the user wants to cancel the order, a special request will be filed, *inAmount/outAmount* will be 0 to close the order. An expired order will not be updated on the blockchain - it can be tracked through the final cutting time. Therefore, we expect most of the orders will expire or be invalidated.

3.9 Smart Contracts

Loopring consists of many smart contracts, including:

- **Mix-Matched Contract** is responsible for ensuring each order status in the loop, calculating the price and volume, transferring and interactionwith other smart contracts, API for Loopring;

- **Order Contract** updates order database and support cancelling policy;

- **Registration Contract** maintains and upgrades service for exchanges who accepted Loopring, support the token deposit from exchange and defaulted parameters backup;

- **Stats Contact** calculates the exchange volume and price between two tokens.

4 Protocol Token

We will issue a token based on ERC20 Ethereum Token Standard called *LRC* (displays in italics).

4.1 Token Application

LRC will be use in the following areas:

- **Gas Fees** — *LRC* can be paid as transaction fee for exchange. It will be simple and productive for the exchange to calculate all the cost in *LRC*. Same as request sender and receiver. We have mention this from previous chapter3.4.3.

- **Deposit for Exchange Registration** — The decentralized exchange mechanism has no limits on location or time. Consequently, exchanges with a high turnover would receive more orders and get more users. As a result of this, we have designed a policy for such exchanges that allow them to use *LRC* to deposit into a smart contract in order to increase the exchanges credibility. Moreover, it can also protect users from certain adverse circumstances.

4.2 Decentralized Governance

Regulation has been updated as well as the exchanges mechanism. Any LRC holders have the voting power S, and number of the pledging N and pledging time $CoinAge$

$$S = f(N, CoinAge),$$

where $CoinAge = H_c - H_s$. Joining CoinAge aids to protect customers from speculation.

The decentralized mechanism includes token registration, exchange registration, stat hash, deposit scale, maximum length, discount hash, subcontract address.

- **token registration** Loopring would adjust token; low trading volume will be eliminated and new popular token will be replaced. All the adjustments have to be recorded on smart contract.

- **exchange registration** Only those exchanges that accept Loopring would allow trading to begin.

- **stat hash** Data will increase to a certain amount after a long period of operation. The more data exchanges have, the more accurate the system computation ability will be.

- **deposit scale** Deposit for each exchange should be measurable. if the amount is huge, the liquidation gets worse; verse vice.

- **maximum length** Technically, more orders can create more profit, however the risk of failure also increases, as well as the trading cost.

- **discount hash** Discount hash will be adjust with the market. The below figure shows normal market (represented by the blue line), supply market (represented by the blue line), and the demand market (represented by the red line).
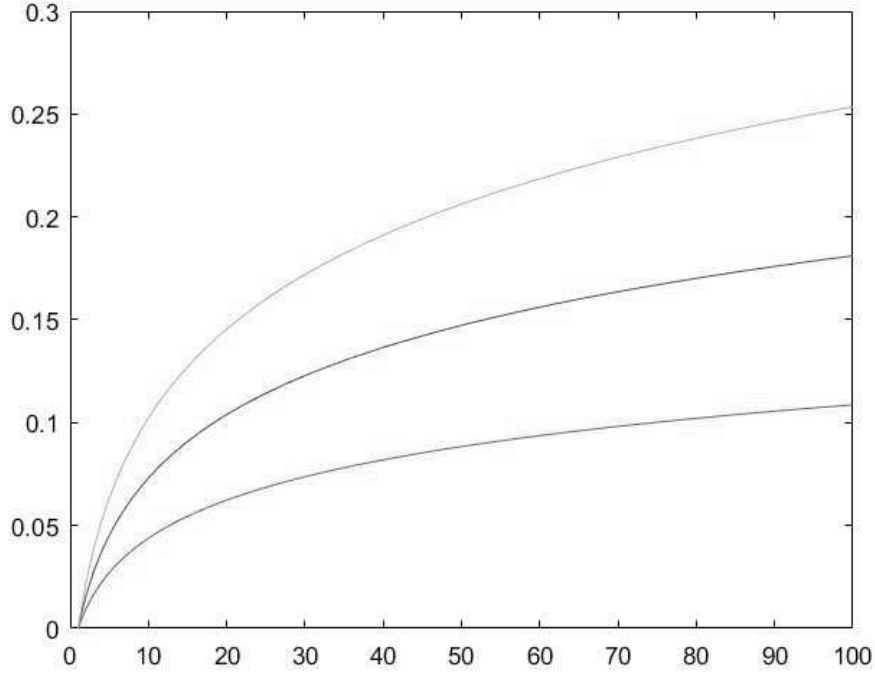

Figure 4: discount rate after adjustment

- **subcontract address** If Loopring exchange is based on the Ethereum ecosystem, then the smart contract cannot be modified. Therefore, one must update Loopring's subcontract in order to modify the subcontract address.

4.3   Token's Liquidability

Loopring's token is based on ERC20 Ethereum Token Standard and can be liquidated through a Loopring smart contract. This means that LRC trading can be done through a centralized exchange. All the ERC20 Ethereum tokens can be exchanged to LRC token (assume pre-order is LRC, with zero fee) by adopting Loopring's decentralized mechanism.

5   Exchange

An exchange is unable to guarantee all transactions could make profit after adopting Loopring. The first reason is a high operating cost. Secondly, high expectation cannot match the actual outcome. There are few other reasons that would cause this saturation. Overall, both the exchange platform and other parties have a reciprocal relationship: the exchange looks for a profitable order, while order senders look for an exchange with the lowest fee. An exchange is not responsible for users ERC20 token after accepting Loopring. The workload has moved from money deposit, withdrawal, and internal virtual account management, to mix-matched order service. Meanwhile, for the users, Loopring does not require the customer to deposit or lock any asset, that means an asset has zero risk to get stolen. Concurrently, a single order can mix and match multiple trades. For Non ERC20 asset, an exchange can offer a asset tokenization service.

5.1   Regular and Loopring Exchange Comparison

In a regular exchange, the "Maker" sends an order, and the "Taker" receive this order. The exchange's price highly depends on the sender's end. Under Loopring circumstances, it has adopted an Over-The-Counter (OTC) module. In the current market, there is a considerably high risk for users to trade in these platforms; there are no laws to regulate the exchange if they vanish. But with Loopring, users do not to deposit money to the exchange anymore. All the transactions will be made among a user's coin address. Another feature for Loopring is that it changes the "Trading Pair" concept. Transactions can be completed with multiple parties, rather than two parties in current exchange scenarios.

	Centralized Exchange	Loopring Exchange
Deposit for the order	Yes	No [1]
Frozen Account	Yes	No [2]
Deposit/Withdraw	Yes	No [3]
Internal Trading Risk	Yes	No[4]
Customer loss from exchange closing	Yes	No[5]
Transaction is the main income	Yes	No[6]
Accept Fiat Money	Yes	Yes[7]
Can be traded among multiple exchanges	No	Yes [8]
Fairness for Maker and Taker	No	Yes [9]
Mix and Match Trading	No	Yes[10]
Supervision	Strong	Weak[11]

Table 2: Contrast between centralized exchange and Loopring exchange

6   Summary

Loopring is a protocol that facilitates decentralized exchange of ERC20 tokens on the Ethereum blockchain. Loopring allows multi-token transaction exchange, as well as acceptance of exchange of liquidation on the blockchain under different circumstances. Loopring offers benefit to both users and exchanges by deferring risk from both parties in decentralized smart contracts, minimising fees and cost to create more profitable orders through ring-matching and order-sharing, and as a cross-platform protocol. Loopring protocol fits any ERC20 and smart contract blockchain platform. After many discussion, our team will develop Loopring on the Ethereum blockchain.

7   Acknowledgements

---

[1] Exchanges execute under Loopring ecosystem do not require any deposit - Tokens are kept in user's wallet, no transaction will be made before the full contract close. As a result, no account can be stolen, or asset lost risk.

[2] Loopring exchanges do not require freeze trading fund — If a user partially or fully modifies the fund, the contract will be withdraw automatically.

[3] The sender's order can be distributed to multiple receivers to be partially or fully fulfilled under Loopring ecosystem.

[4] All matching trades are based on smart contract on blockchain, data are immutable and transparent.

[5] Loopring exchanges are not responsible for tokenization, thus Loopring users will not be affected if an exchange becomes insolvent. For example, a blockchain account will not affected if mining is terminated. In conclusion, exchanges are responsible for matching trades. Loopring's smart contract will complete clearing and settlement, furthermore, assets are always kept in users blockchain account.

[6] Transaction fee is not a mainstream income for Loopring exchanges, mainstream comes from profit of transaction margin, because it can effectively encourage trade matching.

[7] Loopring exchanges fully support asset tokenization, hence, it requires legitimate currency being tokenized on ERC20 standard.

[8] Loopring allows multiple Loopring exchanges to partially or fully trade off one order at same time.

[9] Transaction price is closed to the balance price instead of being tendered to the makers offer price under Loopring protocol.

[10] Loopring exchanges multiple supporting feature can help sender to find the most profitable order.

[11] Loopring exchanges do not require a deposit. Clearing and settlement are made through the open source smart contract. Therefore, regulation is not necessaryif there is no asset tokenization occurrence.

References

[1] Economist Staff. Blockchains: The great chain of being sure about things. *The Economist. Retrieved*, 18, 2016.

[2] Melanie Swan. *Blockchain: Blueprint for a new economy.* " O'Reilly Media, Inc.", 2015.

[3] Satoshi Nakamoto. Bitcoin: A peer-to-peer electronic cash system. 2008.

[4] Leslie Lamport, Robert Shostak, and Marshall Pease. The byzantine generals problem. *ACM Transactions on Programming Languages and Systems (TOPLAS)*, 4(3):382–401, 1982.

[5] Konstantinos Christidis and Michael Devetsikiotis. Blockchains and smart contracts for the internet of things. *IEEE Access*, 4:2292–2303, 2016.

[6] Gavin Wood. Ethereum: A secure decentralised generalised transaction ledger. *Ethereum Project Yellow Paper*, 151, 2014.

[7] Paul Tak Shing Liu. Medical record system using blockchain, big data and tokenization. In *Information and Communications Security*, pages 254–261. Springer, 2016.

[8] Robert McMillan. The inside story of mt. gox, bitcoins 460 dollar million disaster. 2014.

[9] David Schwartz, Noah Youngs, and Arthur Britto. The ripple protocol consensus algorithm. *Ripple Labs Inc White Paper*, 5, 2014.

[10] Stefan Thomas and Evan Schwartz. A protocol for interledger payments. *URL https://interledger. org/interledger. pdf*, 2015.

[11] Fabian Schuh and Daniel Larimer. Bitshares 2.0: Financial smart contract platform.

[12] Fabian Schuh and Daniel Larimer. Bitshares 2.0: General overview, 2015.

[13] Open ledger. *URL https://openledger.info/*, 2017.

[14] Bancor protocol. *URL https://bancor.network/*, 2017.

[15] Robin Hanson. Logarithmic markets coring rules for modular combinatorial information aggregation. *The Journal of Prediction Markets*, 1(1):3–15, 2012.

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

[17] Fabian Vogelsteller. Erc: Token standard. *URL https://github.com /ethereum /EIPs /issues /20*, 2015.