

Oedax: Loopring's Open-End Dutch Auction Exchange Model



Daniel Wang

Mar 9 · 10 min read

We introduce an enhanced Dutch Auction-based exchange model, Oedax, that allows both buyers and sellers to participate after an auction starts. In this post, we outline the design of such an exchange model and analyze some of its properties.



Dutch Auction

In a Dutch Auction, a fixed amount of an asset is put into the auction for sale. The initial ask price starts (much) higher than any open market prices and decreases gradually to zero, according to some pre-specified function. The dutch auction ends if and only if the actual price, calculated by dividing the accumulated money (sent by the bidders) by the number of assets (being sold by sellers), equals the ask-price at a given moment.

Dutch Auctions have some desirable properties. First of all, auctions always settle, guaranteed. Secondly, all participants in the same auction get the same price for settlement, regardless when they participate in the auction and the size of their contributions. This property avoids front-running issues that most order-based exchange models suffer. This same-settlement price property, however, does have a drawback as well—participants tend to withhold until a later time to participate when the ask-price becomes more reasonable instead of during the early phase of the auction.

Dutch Auction is an excellent means of price discovery, especially in a decentralized environment powered by blockchain technology. Specifically, the settlement prices in Dutch Auction can act as decentralized price oracles for smart contracts—assuming the auctions are well known and there are enough participants.

The original Dutch Auction model is designed for one-way asset offering and does not allow additions of assets being sold once the auctions start. We propose an enhanced version called *Open-end Dutch Auction Exchange*, or *Oedax*, which is more like a bi-directional exchange instead of a one-way auction.

Introducing Oedax

We designed Oedax to facilitate two groups of people, sellers and buyers, to exchange assets, especially cryptocurrencies or crypto tokens. Therefore, in this post, we use crypto tokens as examples for easier description, but Oedax is truly generic and can be applied to the exchange of other types of assets.

Suppose sellers want to sell *TokenA* for *TokenB*, and buyers want to buy *TokenA* with *TokenB*. Oedax allows sellers to auction off *TokenA* to buyers for *TokenB*, and in the meanwhile and the very same auction,

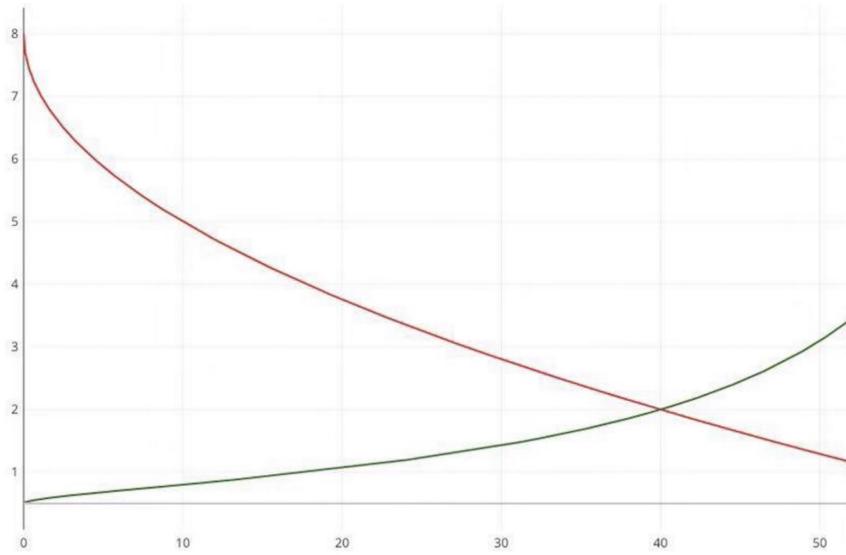
allows buyers to auction off *TokenB* to sellers for *TokenA*. Therefore an Oedax auction can be perceived as two internal-auctions seemingly integrated with shared parameters.

We further suppose that the fair market price for *TokenA*, with respect to *TokenB*, is P ; and the initial sell-price of *TokenA* in an Oedax auction is $M*P$ where $M > 1$ —we call P the *target price* and M the *price scale factor*. The *duration* of the Oedax auction is τ , which is the expected time the auction will end if no one ever participated in the auction.

Then we specify two price curves, one for the selling of *TokenA*, i.e., the *Sell Curve* (or SC), and one for the buying of *TokenA*, i.e., the *Buy Curve* (or BC). These two curves are designed to satisfy the following requirements:

- $SC(0) == P*M \ \&\& \ SC(\tau) == P/M$ (rule#1);
- $BC(0) == P/M \ \&\& \ BC(\tau) == P*M$ (rule#2);
- There exists a time t , $t \leq \tau$, such that $SC(t) == BC(t) == P$ (rule#3).

We further assume the amount of *TokenA* in the auction at any time t is $Q_s(t)$, and the amount of *TokenB* at any time t is $Q_b(t)$. Then a horizontal line representing the *actual price* $p(t) = Q_b(t)/Q_s(t)$ can be drawn. We call this horizontal line the *actual price line*, or *APL*. Note that *APL* moves upward if more *TokenB* participates in the auction and downward if more *TokenA* participates in the auction.

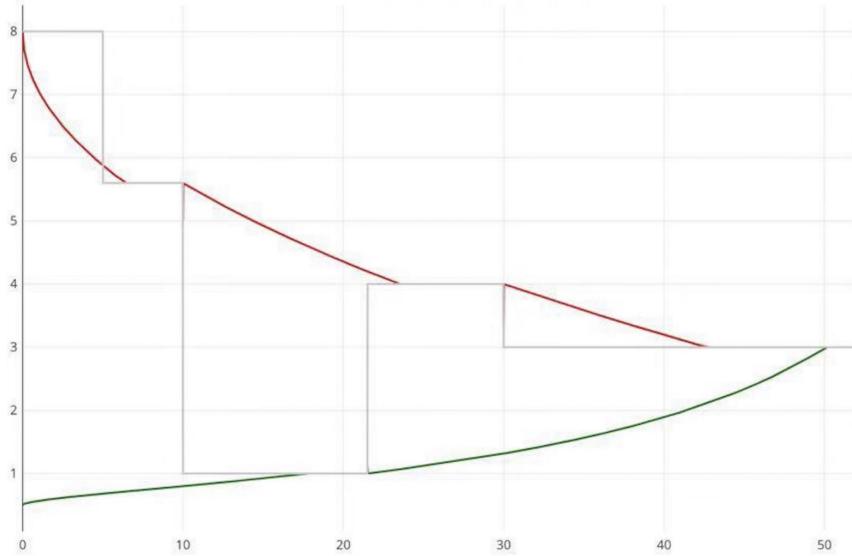


Price Curves (red: Sell Curve, green: Buy Curve) with $T = 40$, $P = 2.0$, $M = 4.0$

Price Bonding and Settlement

An Oedax auction can start with zero *TokenA* and/or *TokenB* deposit, i.e., $Q_b(0) == Q_s(0) == 0$. As time passes, participants can deposit either *TokenA* or *TokenB* to change the actual price p . But Oedax applies some important price bonding rules:

- Once the actual price p falls in between SC and BC curves at time T' , it should always stay in between the curves to make $BC(t) \leq p(t) \leq SC(t)$ hold for any $t \geq T'$ until the auction ends (rule#4);
- The Sell Curve cannot drop further down once it meets the actual price line, but it can resume dropping once the actual price line moves downward (so there is room from the Sell Curve to drop further). When resuming dropping down, the Sell Curve must drop from its previous value. If plotted, the Sell Curve seems like it's been cut into pieces and has been partially shifted to the right along the x-axis. The same rule applies to the Buy Curve as well, but in the opposite direction (rule#5);
- The converging of the Sell Curve and the Buy Curve marks the end of the auction. If the actual price line has never fallen in between the price curves, the auction will not settle and all tokens will be returned to participants; otherwise, the actual price line will certainly converge with the two price curves, and the settlement occurs. (rule#6);



An Oedax auction that ends and settles at time 50 and price 3.0—the horizontal gray lines represent the actual prices at different moments.

A well-designed Oedax auction should have the actual price line be inside the Buy and Sell Curves before or immediately after the start of the auction to make sure it settles. This will boost user participation. The final settlement price is irrelevant of target price but is supposed to be close to it.

Phases

Oedax auctions only settle if the actual price falls in between the two bonding curves and the actual price line at time τ' , we call the phase from time 0 to τ' *phase-1*. Phase-1 has the most uncertainties for participants and should be minimized if possible. τ' marks the start of *phase-2*, which has fewer uncertainties and guarantees the auction will settle.

We can introduce another parameter N , and once the gap between the two price curves is smaller than or equal to N^*P we enter *phase-3*. The other way to define phase-3 is to use time elapsed with respect to T as a measure, e.g., after $0.75T$ we enter phase-3. Phase-3 represents a period where participation in the auction is the least risky and the buy-sell price gap is small enough.

It is certainly possible to define more phases or introduce further finer granularity for classifying auction participation. The general idea is to encourage and reward early participation which is critical for creating more liquidity, and charge fees for later participation. Conceptually, we

can treat early participation as market makers and recent participation as takers.

Participation and Cancellation

In phase-1, participants can deposit any amount of *TokenA* or *TokenB* into the auction without restrictions. If withdrawal is permitted, participants can also withdraw any amount of tokens.

After phase-1 and before an auction ends, participants can still deposit more *TokenA* or *TokenB* in the auction, or withdraw from it, but there are limits on the amounts to assure $BC(t) \leq p(t) \leq SC(t)$ always hold. Those limits are:

- The deposit limit for *TokenA* is $Qb(t)/BC(t) - Qs(t)$
- The deposit limit for *TokenB* is $Qs(t)*SC(t) - Qb(t)$
- The withdrawal limit for *TokenA* is $Qs(t) - Qb(t)/SC(t)$
- The withdrawal limit for *TokenB* is $Qb(t) - Qs(t)*BC(t)$

To facilitate later participation in large sizes, Oedax can queue the amount beyond the current deposit limit in a waiting list, and when there is a counter-party beyond-limit deposit, Oedax will accept deposits from both buy and sell side to end up with at most one waiting list for either the sell or the buy side. The waiting list will automatically expire at the end of the auction.

To discourage withdrawal, a fee may apply. An Oedax auction can be configured with withdrawal disabled.

Curves

The two price curves can be defined independently, e.g, one curve can be a straight line and the other curve can be polynomial. But since these two inner auctions are happening in parallel, it may be more reasonable to design the Buy Curve and Sell Curve in such a way that the sell-price and the buy-price develop with the same velocity. We can achieve this by binding SC and BC using:

- $BC(t)*SC(t) == P*P$ (rule#7)

With this binding, only one of the two curves needs to be defined to derive the other one.

With this rule enforced, the curves for a *TokenA/TokenB* Oedax and curves for a *TokenB/TokenA* Oedax take the same shape. In other words, this rule makes Oedax *token-symmetrical*—an *ABC/XYZ* auction is the same as an *XYZ/ABC* auction. [1]

[1] *The Loopring protocol also adapts such a similiar token-symmetrical data modeling approach—a ABC/XYZ sell order is an XYZ/ABC buy order. In our precious posts, we referred it as “unidirectional order modeling”.*

Oedax Features

Oedax trading model has the following advantages:

- It does not rely on other types of trading platforms for price discovery or adjustment.
- It allows both sellers and buyers to participate in an auction after the auction starts. Oedax can even allow conditional withdrawal before the auction ends.
- It is possible to accept pre-submitted limit price orders and convert them into Oedax participation once the auction's price range satisfy the order's requirements.
- Oedax auction's final settlement volume is not restricted by the initial deposit of either token and can potentially be much larger than a Dutch auction. Oedax is suited for large trades and is market-making friendly.

Oedax inevitably inherits some shortcomings that Dutch Auctions have, including:

- The trading is not instantaneous, it takes time to end the auction;
- The settlement price is reasonable (close to market price) only when the auction is well aware of and has enough participants.



Rolling Oedax

For each trading pair, it is possible to create a serial of Oedax auctions with shared parameters and a constant inter-auction delay. Once an auction ends, a new auction is automatically started[2] with a target price equal to the last settlement price. We can call such a serial of Oedax auctions a “Rolling Oedax”. With enough participants, a Rolling Oedax will keep running and provide much larger trading capability than a standalone Oedax.

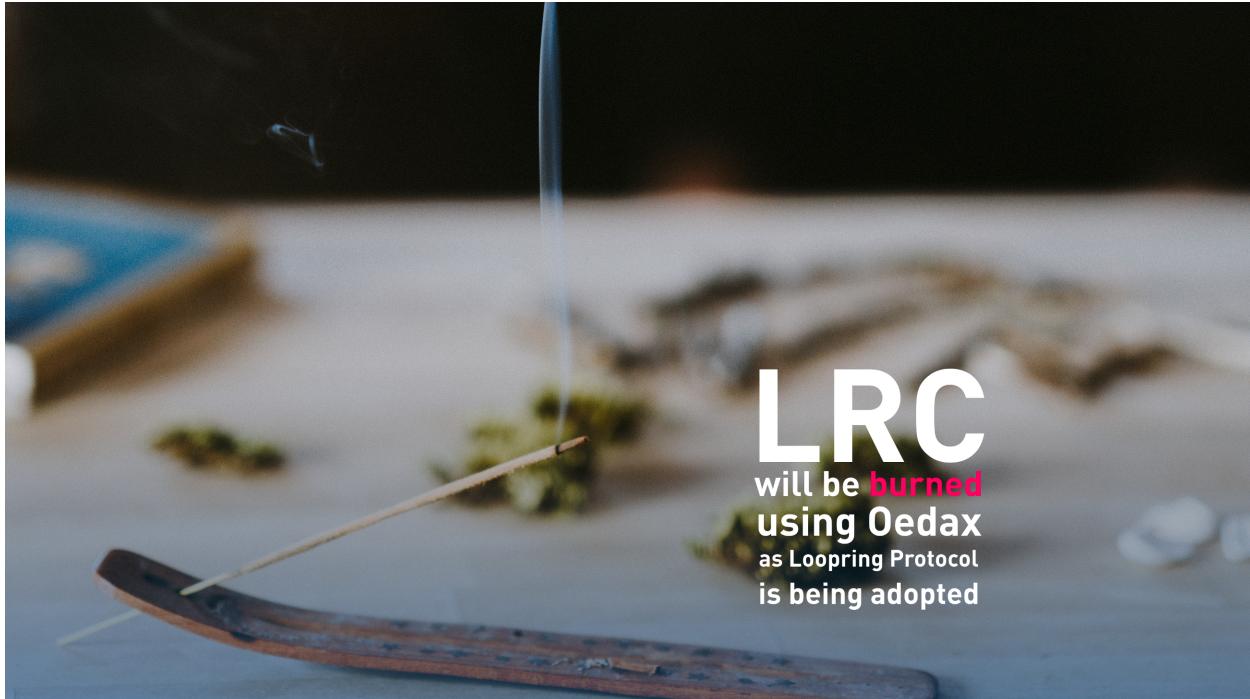
[2]: *Practically auctions must be triggered by a user action to kick off.*

To participate in such a rolling auction, a user will need to participate in the earliest running Oedax auction of such a serial, if the size of his/her deposit is beyond the earliest auction’s deposit limit, the remaining deposit will participate in the second-earliest auction, ... If the user’s deposit is beyond the limit of all running auctions combined, the remaining deposit can be put into a waiting list which is shared by all outstanding Oedax auctions. When a new auction starts, the waiting list will be removed and all outstanding deposits in the waiting list will be converted to the *initial deposits* for the new auction.

It is also possible to treat more recent auctions as special waiting lists for older running auctions if withdrawal is enabled, which is likely to increase the trading volume of the oldest auction yet to settle.

Otherwise, settlement across multiple auctions will even out the average price of each participation (order) over a designated period of time.

Rolling Oedax enables DEXs to facilitate market-price based peer-to-peer trading and guarantees that such orders will be traded with the least risky counterparties first at reasonable prices without using any price oracle.



Use Oedax in Loopring

Oedax will be used by Loopring 2.x to achieve "Fee Burning", a unique token economic feature that enables Loopring relayers to accept any ERC20 tokens as fees, and to pay the protocol smart contract a small percentage of the fees called *the burn*.

Anonymous users will be able to trigger the burn to be auctioned off for LRC token in Oedax auctions, and the purchased LRC will be sent to address `0x0` automatically (our planned LRC ERC20 upgrade will

treat this as a special destructive operation and reduce the total LRC supply). Oedax will automate the fee burning process in a trustless and decentralized fashion, which is an essential part of Loopring's governance objective.

The fees collected by Oedax auctions will also be auctioned off for LRC, to burn.

Development & LEAF Grant

Our goal is to develop Oedax as a permissionless and fully decentralized way of trading tokens. It means everybody can create auctions for all existing and future tokens at any time in arbitrary sizes — there will be no token or participant whitelist of any kind.

We plan to open-source the Oedax smart Ccontract [repository](#) on GitHub once we get ready for the initial deployment on Ethereum mainnet. In the meanwhile, if you are interested in co-developing Oedax on Ethereum or lead the effort to make it work on other platforms, please let us know. We are setting up an LRC Grant from the Loopring Ecosystem Advancement Fund ([LEAF](#)) to incentivize quality work.

In the near future, we will add Oedax UI in [UpWallet](#).

• • •

We would love to hear from you. If you do feel like sharing your thoughts, please send us an email to foundation@loopring.org.

• • •

To stay up-to-date with Loopring, please [sign up](#) for Loopring's Bi-Weekly Update, and find us at:

- ★ Twitter: twitter.com/loopringorg
- ★ Reddit: reddit.com/r/loopringorg
- ★ Telegram: t.me/loopring_en
- ★ Discord: discord.gg/KkYccYp

- ★ StackOverflow: stackoverflow.com/c/loopring
- ★ Kakao: open.kakao.com/o/gJbSZdF (Korean)