Decentralized Exchanges

CS839 – Kai Mast

Buying Cryptocurrencies

Example: Coinbase

- Users do not manage, or have access to, their own wallet keys
  - A single entity that holds about $300 million in cryptocurrency[1]
  - Obvious target for hackers[2]
  - Very convenient for beginners, but defeats the purpose of decentralization

Buying from other users directly?

- No easy way to find a seller.
- How do we actually exchange tokens safely?

[1]Source: https://www.cnbc.com/2021/02/25/coinbases-crypto-holdings-jumped-ninefold-last-year-as-bitcoin-surged.html

[2]Source: https://www.techradar.com/news/coinbase-hack-sees-thousands-of-users-accounts-drained

Decentralized Exchanges (DEXs)

Idea: Use the blockchain itself to exchange different tokens/currencies

Two major approaches:

- Direct Peer-to-Peer exchange between two parties
  - Relies on time locks, similar to off-chain mechanisms
  - Examples are **Komodo** and **0x**
- Pool Based: Smart contract holds currency and serves as intermediate
  - **Uniswap** is the most common version

Note, that a DEX cannot trade fiat currencies directly

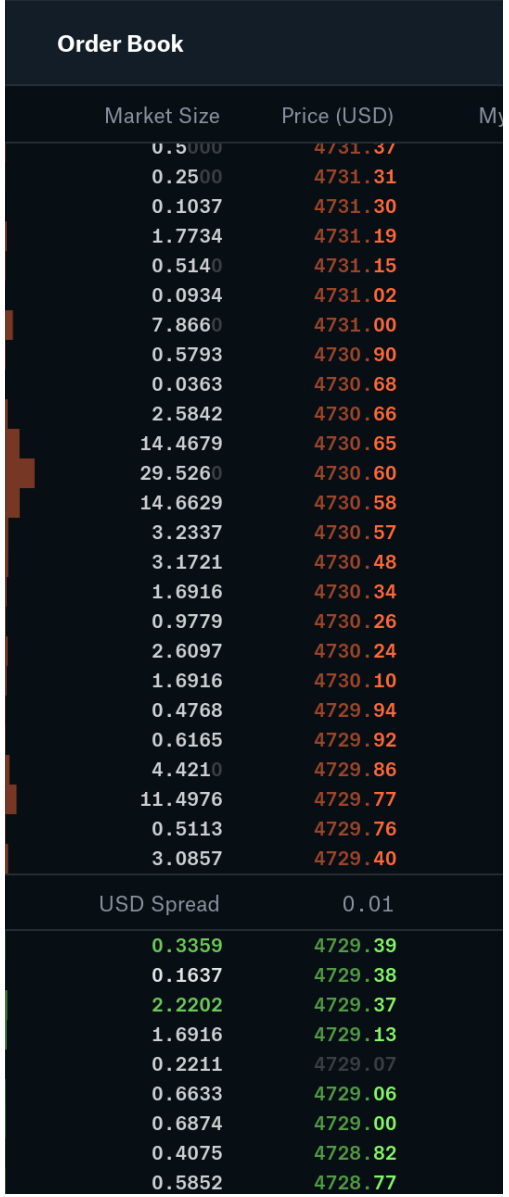
- We need to use stable coins, e.g., Tether, for that

Order Books

- Keeps track of sellers and buyers
  - Each order has a specific price and quantity associated with it
- Sell and buy orders need to be matched (or partially matched) to go through
  - Buy order presents an upper bound for the price
  - Sell order presents a lower bound for the price
  - Need to pick a price (and market size) within that range

Why can this be a problem for a decentralized exchange?

- Ethereum transactions (and Blockchain transactions) can take a long time to confirm
  - Prices may have changed at the before the transaction is confirmed
- Ethereum transaction fees are high and throughput is low
  - Representing every order as a transaction is inefficient



Coinbase order book for ETH/USD

- Top: Sell offers
- Bottom: Buy offers

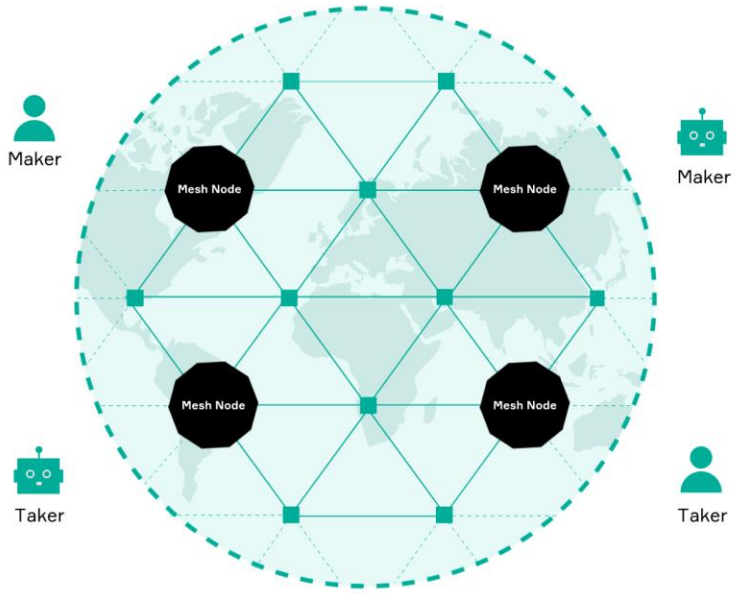
0x: An open protocol for DEX on the Ethereum blockchain

- Proposed in 2017 by Will Warren and Amir Bandeali
- Runs a peer-to-peer exchange network
  - "Craigslist for Ethereum"
- A mesh of relay nodes build a decentralized order book
  - Helps discover other sellers/buyers
  - Only fulfilled orders are stored on the main chain

The 0x Protocol

Idea: Maintain the order book off-chain and batch

1. "Makers" post their offers using a cryptographic commitment
2. They can send their orders through the relay network or directly to a "taker"
3. Takers accept or reject orders
4. Finally, takers fullfill orders by posting transactions on the blockchains



Source: https://0x.org/docs/core-concepts

- First proposed by Vitalik Buterin (once again...)[1]
- Development started in 2017 by Hayden Adams
- The most popular DEX (right now)
- Runs as a smart contract on Ethereum
- Governed by the UNI token, which itself is one of the most traded tokens

[1] https://vitalik.ca/general/2017/06/22/marketmakers.html

Market Makers

- Market makers serves as an intermediate between sellers and buyers
  - Market makers can provide from the "ask-bid spread" (difference between sell and by offers)
- Also called *"liquidity provider"*
  - They offer a large quantity of the traded item(s) to facilitate continuous trade

Liquidity Providers in Uniswap

- Participants pool currency/tokens in a smart contract
- The contract serves as a liquidity provider
  - One contract per currency/token pair
- When people trade through with the contract, a fee is charged
  - Each pool participants gets share of the fee proportional to their pool contribution

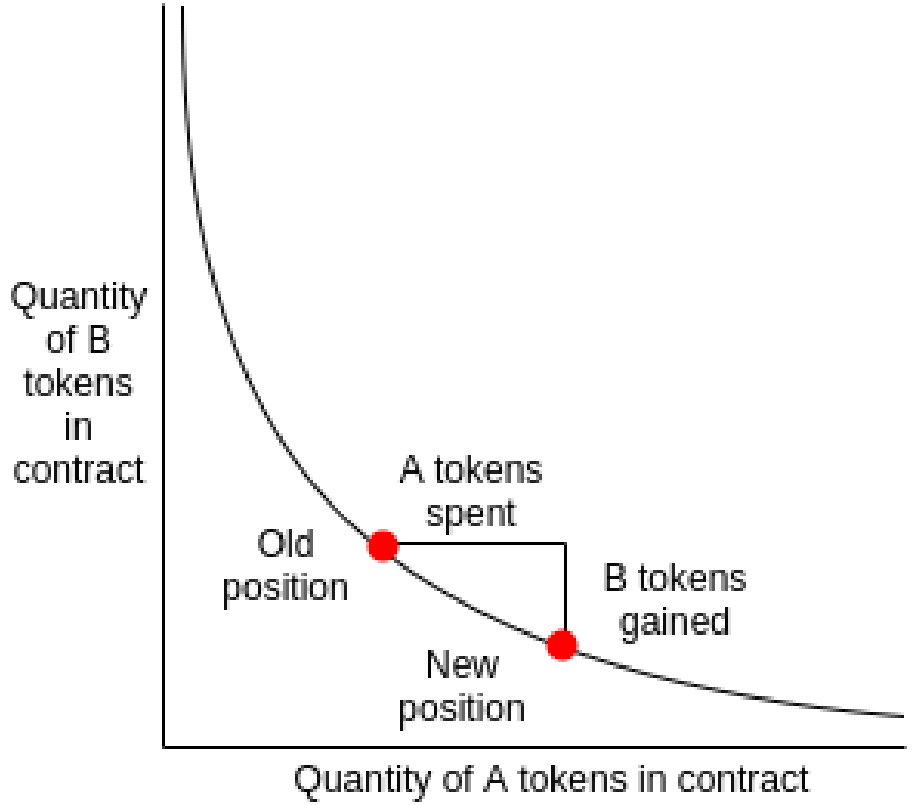
Determining the Price in Uniswap (v1)

Price is set by the smart contract so that the following equation always holds:

(Amount of Token1) * (Amount of Token2) = const.

For example

- If supply of Token1 decreases price (in Token2) increases
- If of both are equal, they can be exchanged 1-to-1
- If you want to buy all of Token1, the price is infinite



Source: https://docs.ethhub.io/guides/graphical-guide-for-understanding-uniswap/

Providing Liquidity in Uniswap (v1)

- When storing liquidity in an exchange, participants receive some number of exchange-specific tokens in return
- If participants want to exit the exchange, exchange tokens can be converted back into liquidity *at the current exchange rate*

That's all for today

Let us look at some code...

https://github.com/Uniswap/v1-contracts/blob/master/contracts/uniswap_exchange.vy